博士学位論文

意思決定のための設計空間探査と設計モード解析に関する研究

Study on Design Exploration and Design Mode Analysis for Decision Making

日 和　悟

Satoru Hiwa

同志社大学大学院生命医科学研究科

2015年3月

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Engineering design problems in real-world situations usually consist of a set of controllable parameters and quality measures of the candidate designs. Taking a vehicle design as an example, engine control parameters, body shape and topology of the vehicle and tire size can be controllable design parameters. As quality measures of the vehicle design, fuel economy, CO2 emission, accelerating performance and total mass can be readily considered. Engineers have to tackle the problem by exploring the candidates of the best design with better quality measure. Difficulties may arise in their exploration of the better designs, due to massiveness of the number of design parameters and tradeoff among plural quality measures. Moreover, once they found the candidate designs, and then they will suffer from how they should choose the best design in multiple, conflicting or competing quality measures. Such a kind of problems can be mathematically defined and solved as multiple criteria decision making (MCDM) problems.

Engineering design proceeds from the requirement definition to conceptual design stage, finally the detailed design. Figure 1.1 illustrates general example of engineering design process. Abovementioned MCDM problems usually occur in the early stage, especially in conceptual design stage. For example, in a vehicle design, how to improve the fuel efficiency under limitation of the development cost must be deliberated and determined at the conceptual design phase. Since a lot of design candidates would exist in the early state of development, unpromising designs should be discarded to efficiently advance the development [1]. Furthermore, Asiedu et al. [2] claimed that over 70% of the total life cycle cost of a product was committed at the early design stage. The conceptual design stage would be very important for such a reason. It is important to understand what kinds of design candidates exist and focus on promising one in the early state of the design process. This is the essence of conceptual design.

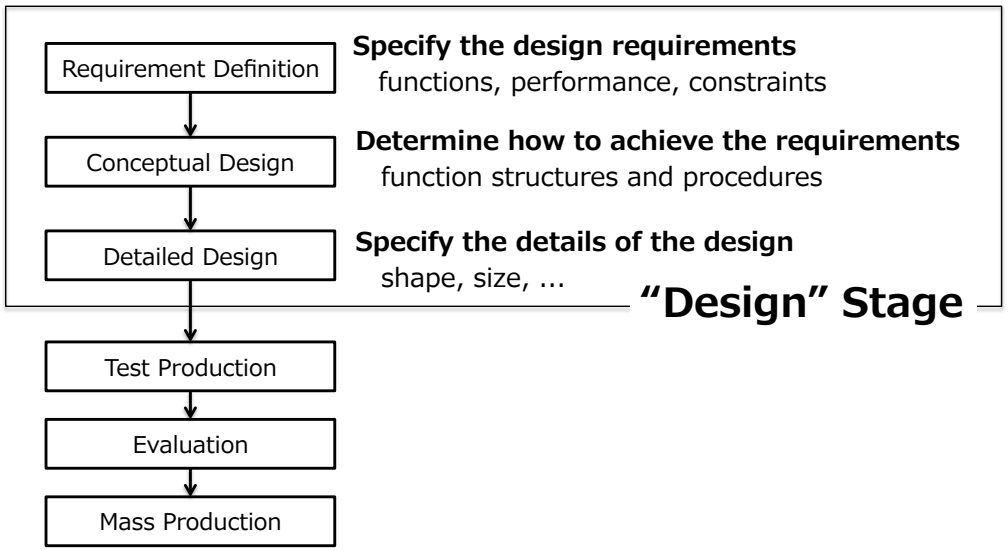


Fig. 1.1 General example of the engineering design process

1.2 Objectives

As mentioned in Section 1.1, we have following difficulties in tackling the real-world engineering design problem:

1. **Finding better candidates**

   As there are huge number of design parameters and the plural quality measures conflict each other, it is difficult to find better designs in a reasonable time.

2. **Choosing the best design from the candidates**

   As there are varieties of candidates and their feature spaces are high-dimensional, it is hard to scrutinize and compare the candidates one by one.

Objective of this thesis is solving above two issues and achieve the effective design exploration and the decision making support in engineering design. This work provides the analytical framework to solve them. Proposing framework consists of the following two parts:

1. **Multiobjective optimization framework for design exploration**

   In order to obtain better designs with many different varieties, novel framework of multiobjective optimization is proposed and developed.

2. **Design mode analysis**

   Efficient analytical framework to classify the design patterns and quantitatively extract the characteristics of representative designs is proposed and developed.

With the proposing methodologies, decision makers (engineers, product designers) are expected to obtain the characteristics of the representative and promising designs, and it will support their efficient and stable decision making in the early stage of the engineering design.

1.3 Outline of the Dissertation

The remainder of this dissertation is organized as follows. After this chapter, in Chapter 2, the concept of design exploration and optimization for MCDM is introduced. The background and motivation to develop the multiobjective optimization framework (described in Chapter 3) and decision support tool (described in Chapter 4) are described.

Chapter 3 is dedicated to multiobjective optimization framework to obtain better designs with many different varieties. Its concept, general framework and implementation examples using existing popular optimization algorithms are described. The effectiveness of the proposed framework is validated through the numerical experiments using the mathematical benchmark problems.

In Chapter 4, design mode analysis, which is analytical method for design pattern classification and feature extraction is investigated. First, the basic concept of ”design mode” is defined. And then the general framework of design mode analysis is introduced. The procedure of design mode framework is described through the popular mutiobjective optimization problem, a multiobjective knapsack problem. After that, as an example of the real-world engineering design problems, the conceptual design of hybrid rocket engine design are analyzed using design mode analysis. Finally, as an extension of application field of proposed method, fNIRS (functional Near-Infrared Spectroscopy) data, which is the time-series data of the hemodynamic responses of oxy- and deoxy-hemoglobin (Hb) in human brain, is also analyzed using design mode analysis. How to apply the time-series data to the proposed method (representation of the data and interpretation of design mode) is described and its possibility to examine brain’s activity is discussed.

Chapter 6 summarizes the overall main point of this dissertation and concludes it. Several directions for future research are also discussed in this chapter.

Chapter 2

Design Exploration and Optimization for Multiple-Criteria Decision Making

2.1 Multiobjective Optimization in Multiple-Criteria Decision Making

Multiple-criteria decision making (MCDM) [3] is a class of problems from which we make decisions among multiple and conflicting criteria. Since there is no unique (best) decision in MCDM problems, they are solved by seeking a set of available alternatives. After the alternatives are obtained, decision maker (DM) chooses their preferred decisions. In this procedure a set of alternatives are represented as a Pareto solution set. This procedure is also known as multiobjective optimization. Nondominated solutions are the ones that are not dominated in all objectives (criteria) by some other solution. Multiobjective optimization algorithms derive good approximations to the Pareto optimal solutions, which cannot be further improved.

Numerical optimization aims to maximize or minimize an objective function, subject to constraints on the possible values of the decision variables. From an engineering standpoint, the objective function is a metric that measures the performance or quality of the designed engineering system, which is influenced by the decision variables. The constraints can then be regarded as design requirements to be satisfied.

In many real-world engineering problems, decisions must satisfy several types of performance measures while meeting numerous design requirements. Situations of conflicting performance metrics, which typify real-world problems, are termed multi-objective optimization problems (MOPs). Because an MOP solution cannot simultaneously minimize or maximize all objectives, MOP processes construct tradeoff curves among objectives, which are then provided to decision makers who select the solution that satisfies their own requirements. The general formulation of

MOPs is given below:

$$\begin{aligned} \text{minimize} \quad & F(\boldsymbol{x}) = (f_1(\boldsymbol{x}), \ldots, f_m(\boldsymbol{x}))^T \\ \text{subject to} \quad & \boldsymbol{x} \in S \end{aligned} \tag{2.1}$$

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

$$\begin{aligned} S = \{\boldsymbol{x} \in \mathbb{R}^{\kappa} \mid \quad & g_j(\boldsymbol{x}) \leq 0 (j = 1, \ldots, l), \\ & h_k(\boldsymbol{x}) = 0 (k = 1, \ldots, p)\} \end{aligned} \tag{2.2}$$

Here, $g_j(\boldsymbol{x})$ and $h_k(\boldsymbol{x})$ are inequality and equality constraints, respectively. As mentioned above, due to the tradeoff among the conflicting objectives in MOPs, no solution in S will simultaneously minimize all objectives. Therefore, we solve the MOP using the concept of Pareto optimality [4, 5]. Pareto optimality that is valid for minimization problems is defined in general, as shown below.

Definition 1: Pareto Dominance

Given $\boldsymbol{x_1}, \boldsymbol{x_2} (\boldsymbol{x_1} \neq \boldsymbol{x_2}) \in S$, $\boldsymbol{x_1}$ is said to *dominate* $\boldsymbol{x_2}$ with respect to $f_i(\boldsymbol{x_1}) \leq f_i(\boldsymbol{x_2}) (\forall\, i = 1, \ldots, m)$, and $f_i(\boldsymbol{x_1}) < f_i(\boldsymbol{x_2})$ for at least one $i = 1, \ldots, m$.

Definition 2: Pareto Optimality

Let $\boldsymbol{x_0} \in S$. We say that $\boldsymbol{x_0}$ is *Pareto optimal* when there are no other solutions in S that dominate $\boldsymbol{x_0}$.

Because MOPs typically admit multiple Pareto-optimal solutions, they should present all available solutions to decision makers. However, the computation of all Pareto-optimal solutions is extremely time intensive. Therefore, reducing the time of approximating the Pareto solutions has become an important goal of multi-objective optimization algorithms. The accuracy of the approximated Pareto solutions can be evaluated from their proximity and from the diversity of their solutions. In particular, diversity is represented by both "uniformity "and" spread. "High diversity implies a wide variety of available solutions. Well-approximated solutions guide decision makers toward better solutions in the objective space. Because multi-objective evolutionary algorithms (MOEAs) exploit multiple-points searching, they can obtain highly diverse solution sets in a single run, and are widely used to solve MOPs.

2.2 Design Exploration for Multiple-Criteria Decision Making

Engineers (decision makers) can derive the varieties of designs by exploring the candidates of the best design with better quality measure, using multiobjective optimization methodologies. Figure 2.1 illustrates this procedure.

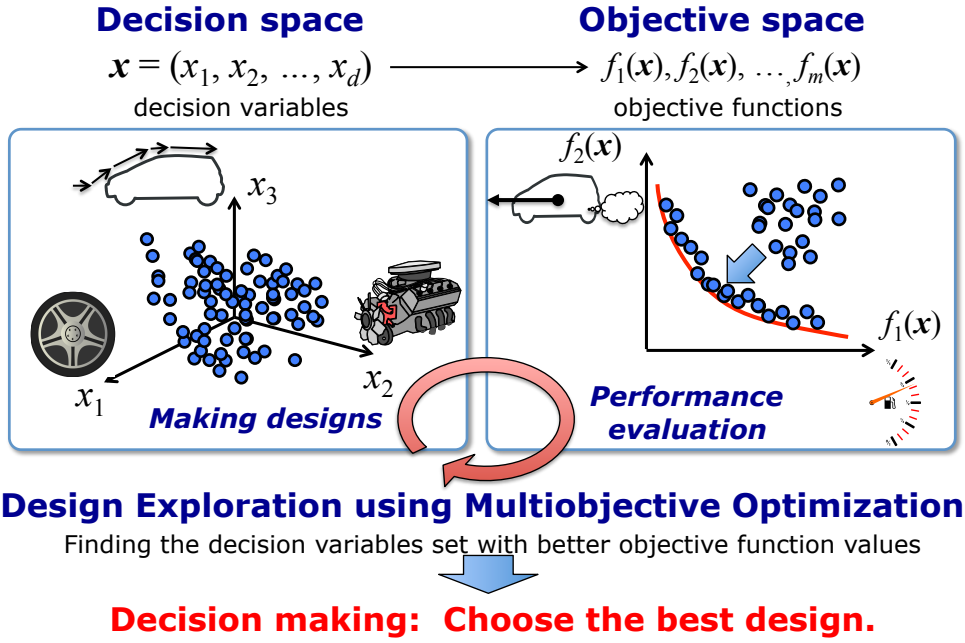


Fig. 2.1 Framework of design exploration and decision making

Once the solutions have been properly converged after the design exploration, the next step is to make decisions from them. However, it is difficult to choose a best compromise in multiple and conflicting objectives without any information about the target problems which aid our decision making. To solve this problem, many researches proposed effective use of optimization results and improved our understanding of target problems. Obayashi et al. [6, 7] proposed the framework of multiobjective design exploration (MODE). MODE consists of the Kriging Model, multiobjective genetic algorithm, analysis of variance and a self-organizing map. It can widely explore decision space and derive many Pareto solutions (alternatives) within a reasonable time using the Kriging meta-modelling of objective functions. The tradeoff information in multiple objectives and decision space characteristics are roughly grasped by visualizing the decision and objective spaces using data mining methodologies such as self-organizing maps.

On the other hand, Oyama et al. [8, 9] used the proper orthogonal decomposition (POD) to decision variables of the airfoil shape design, and revealed that any designs 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 and the parameters which are important to construct them from large and high-dimensional data set, by analyzing the fluctuation vector. Moreover, it is notable that they focused on decision space of Pareto solutions, but the objective space. In engineering design, the objective function value is just an performance metric of obtained design. Of course it is important to quantitatively evaluate the quality of optimized designs, but the engineers would rather be focused on their characteristics in decision space, for example, "which decision variables are dominant in the obtained designs," or "how many design varieties are derived in Pareto solution set," as shown in Figure 2.2. Decision space analysis would give us efficient design strategies.

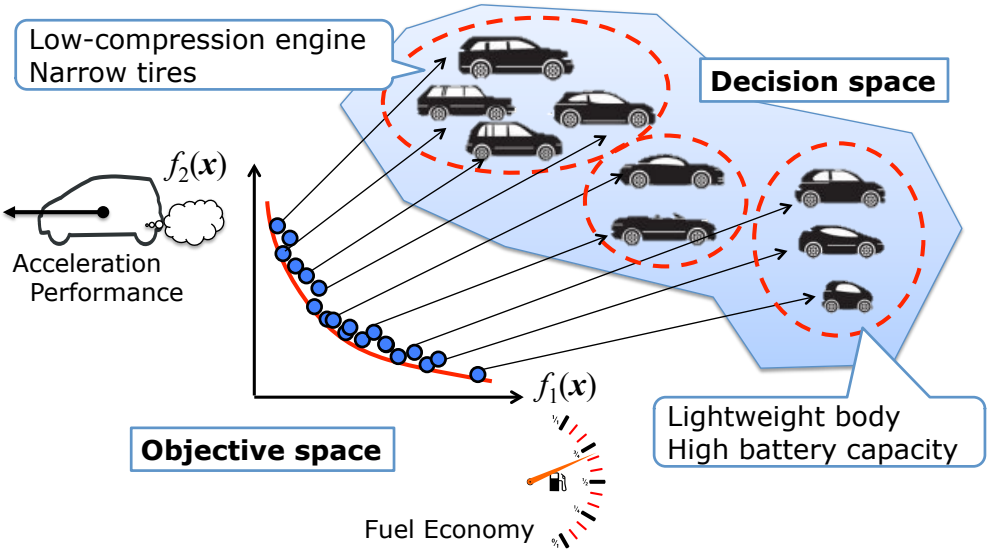


Fig. 2.2 Decision space analysis of Pareto solution set

To wrap up, this research assumes that following processes are very important in order to achieve satisfactory decision making in engineering design:

1. Multiobjective optimization to derive the wide varieties of designs
2. Decision space analysis to obtain the knowledge about the target design problem

This work assumes the entire framework of design exploration and analysis can be expressed as illustrated in Figure 2.3. First of all, decision maker (or an engineer) formulate the target design problem as an optimization problem, with the input of design specification and the constraints. Then, design exploration is performed to obtain the candidate designs. Note that traditional

experimental design can be still effective for the design exploration. It can also be used for the initial value generation of multiobjective optimization. Once the candidates are obtained, they are analyzed by design mode analysis, and finally the representative design patterns, their characteristics and design strategy which tells us how to realize them are fed back to decision maker.

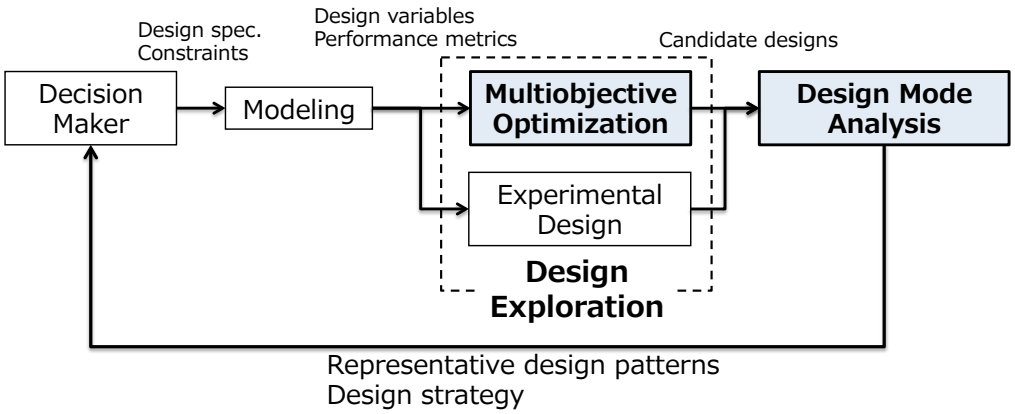


Fig. 2.3 Entire framework of design exploration and analysis

In this dissertation, taking above two important processes into consideration, (1) **novel multiobjective optimization framework to obtain the widely-spread and well-converged Pareto solutions are developed**. And then, inspired by the work of Oyama et al., (2) the concept of the fluctuation vector is extended and a new concept—design mode—is incorpolated. Based on this concept, **"design mode analysis" which is an analytical framework to find the design modes and make an effective use of them is proposed and developed.** It consists of data clustering method and principal component analysis, and expected to extract the representative design types and their characteristics. These information will aid our decision making.

Chapter 3

Development of Multiobjective Optimization Framework

3.1 Introduction

Several available MOEAs offer superior uniformity, spread, and proximity of solutions. The dominance-based framework, implemented by the nondominated sorting genetic algorithm II (NSGA-II) [10], has been widely applied and extended to many real-world problems. NSGA-II is one of the popular MOEA frameworks available [11].

The preference-based framework obtains a solution set within a certain region of the Pareto solutions. Preference information, which defines the converging area, is often represented by reference points that are specified by the decision maker before optimization. The reference-point-based NSGA-II, proposed by Deb et al. [12], is among the most popular implementations of this framework. If decision makers ' preferences are well clarified before optimization, optimal solutions are rapidly obtained. If the preferences of the decision maker are not clearly specified, the preference-based framework should be avoided because it may generate nonuniform solutions.

Many-objective problems are adequately solved by indicator-based frameworks [13]. Here, searching in high-dimensional objective spaces is advanced by an indicator such as a distance measure or hypervolume [14]. The indicator-based framework has been popularized by the indicator-based evolutionary algorithm [15], S-metric selection evolutionary multi-objective algorithm [16][17], and HypE [18]. The majority of research based on indicator-based frameworks has focused on reducing the computational cost of hypervolumes in many-objective problems.

Another efficient MOEA implementation is the decomposition-based framework. This framework, introduced by Zhang et al. [19], adopts a multi-objective evolutionary algorithm based on decomposition (MOEA/D). This algorithm uses a scalarization function and decomposes the

multi-objective problem into several single-objective problems, which are solved using information from neighboring problems. This decomposition-based framework effectively improves all performance metrics of the Pareto solutions, i.e., proximity, uniformity, and spread. MOEA/D is one of efficient MOEAs available [11].

The hybridization approach, which integrates several algorithms with different features, also effectively improves solution diversity and proximity. Martinez et al. [20] improved the solutions of MOEA/D by hybridizing the MOEA algorithm the Nelder-Mead algorithm [21]. In the hybridization scheme of Bosman [22], the existing MOEA is integrated with the multi-objective gradient-based optimization algorithm. These research efforts have improved the MOEA search by combining it with local search algorithms. Conversely, Okuda et al. [23] proposed the distributed-cooperation scheme (DC-scheme), which spreads the Pareto solutions. By combining MOEA with a single-objective evolutionary algorithm (SOEA), the DC-scheme seeks both the nondominated and optimal solutions for each objective. Okuda et al. demonstrated that the DC-scheme derived more widely spread solutions than conventional MOEAs. However, because the search is biased to spread the solutions from an early stage, proximity speed is reduced. From these results, we conclude that MOEAs must properly balance diversity and proximity. Ishibuchi et al. [24] tackled this issue by improving the mating scheme of the NSGA-II algorithm. They developed the similarity-based mating scheme, which they incorporated into the NSGA-II algorithm to solve the multi-objective 0/1 knapsack problem, and demonstrated the improved diversity performance of their scheme. As an other example, TP+PLS algorithm [25] uses two phase search idea. It sequentially hybridizes two algorithms from the dominance-based and scalarization-based algorithms. Two-phase local search (TPLS) [26] is utilized to perform a series of scalarizations and to obtain the good initial nondominated solutions. Then they are further improved by Pareto local search (PLS) using the Pareto dominance criterion in the local search [25]. The TP+PLS search scheme reveals that the two-phase search paradigm with different search objective in each phase is effective in hybridization approach.

The hybridization approach is advantageous in that, unlike other algorithms, each algorithm is functionally specialized by its search characteristic; consequently, each algorithm requires minor modifications. In this paper, we propose a new hybridization scheme that balances diversity and proximity, and obtains well-approximated and widely spread solutions. The features of our proposed scheme are listed below:

- Two search phases that separately improve proximity and diversity.
- The first phase improves the search proximity by a preference-based algorithm.

- The second phase searches for optimal solutions to each objective (referred to as " extreme points ") and spreads the solutions using the DC-scheme.

Our proposed scheme aims to improve the proximity and diversity of Pareto solutions. Whereas conventional MOEAs derive Pareto solutions by a single algorithm, we aim toward a general, versatile, and easily implementable hybrid search scheme that can incorporate newly developed algorithms for easily improved search performance.

The remainder of this paper is organized as follows: In Section 3.2, we introduce the generalized framework of our search scheme, and describe its application to conventional MOEAs. In Section 3.3, we implement our proposed scheme in two ways, and evaluate its performance on continuous MOPs. The paper concludes with Section 3.4. Note that we focus on solving continuous MOPs.

3.2 Proposed MOEA search scheme for improving proximity and spread

This section first conceptualizes our proposed scheme. We then introduce its framework, and finally its application to conventional MOEAs.

3.2.1 Concept

As mentioned above, diversity embraces both uniformity and spread, both of which can be improved by the DC-scheme [23]. Because the DC-scheme uses both MOEA and SOEA, it generates multiple subpopulations from the original population. Specifically, for k objectives, the search population is divided into $k + 1$ subpopulations—one MOEA population and k SOEA populations. In this scheme, the MOEA population searches multiple-objective Pareto solutions and SOEA populations explore the extreme points. Moreover, to spread the MOEA population, some solutions from the SOEA and MOEA populations are exchanged at specified intervals. The SOEA improves the spread of solutions, whereas the MOEA and the solution exchange between SOEA and MOEA improve the uniformity of solutions. However, because searching in the DC-scheme is biased toward improving the diversity of the population from an early stage, it delivers poorer proximity than conventional MOEAs. To improve diversity while maintaining good proximity, our proposed scheme divides the search into two phases. The first phase accelerates proximity and the second improves diversity. Figure 3.1 illustrates our proposed scheme.

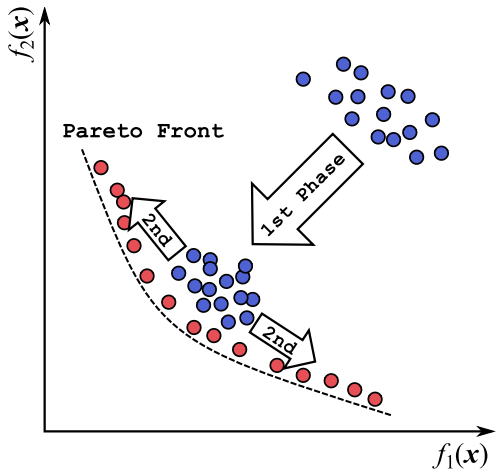


Fig. 3.1 Concept of Proposed Search Scheme

In this work the author determine the generalized framework of the algorithm using proposed search scheme and give representative implementation examples. The performance of two typical implementations using NSGA-II and MOEA/D are tested on recent test suites, and the parameter effects of the search-phase switching criterion are studied.

3.2.2 General Framework

Here, the general framework of our proposed search scheme based on the abovementioned concepts is defined. The two search phases are implemented in three algorithms with different search characteristics, each assigned a different search objective.

First Phase

The first phase of the proposed scheme accelerates the proximity of Pareto solutions. For this purpose, we can guide the MOEA search by the measured distance to the Pareto solutions. However, in many real-world MOPs, the distance between the optimum front and the search population is not readily determinable because the Pareto solutions are unknown. Our proposed scheme, instead, uses a reference point, which is specified by the decision maker within the objective function space, and which can be located in either the feasible or infeasible regions. Because reference-point-based MOEAs prioritize solutions nearby the reference point, solutions tend to converge in that locality. The first phase can adopt any type of reference-point-based or preference-based MOEAs [12][27][28][29] can be applied in the first phase.

Second Phase

The second phase adopts the DC-scheme [23]. Subsequent to the first phase in a k-objective problem, the population is divided into one MOEA subpopulation and k SOEA subpopulations. Because the DC-scheme requires no specific base algorithm, we can apply any SOEA and MOEA. The DC-scheme for a bi-objective problem is illustrated in Figure 3.2. In the DC-scheme, the MOEA and SOEA populations search for Pareto solutions in a parallel manner, and exchange their elite individuals at predefined intervals. Here, an elite individual has the best objective value (f_i) in the SOEA subpopulation. The elite individuals in the MOEA subpopulation are the nondominated solutions with the best objective values (f_i). Hence, in a k-objective problem, the MOEA population contains at least k elite solutions.

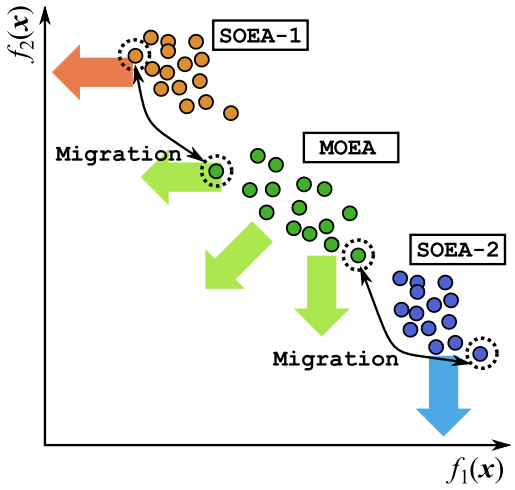


Fig. 3.2 DC-scheme

Criteria for Search Phase Switching

In our framework, one of the most critical design decisions is when to switch the search phase. Because the search phase should automatically switch when the solutions have converged, we require a proximity indicator. For this purpose, Lopez Jaimes et al. [30] developed the multiple resolution multi-objective genetic algorithm (MRMOGA), wherein the indicator is the average ratio of the number of solutions in the parent set that are dominated by an individual. However, if an individual in the parent set dominates many solutions, this indicator increases and the search continues, even when few " effective " individuals that can accelerate the search are generated.

To overcome this problem, the proximity indicator in our scheme is the average number of parent individuals that are dominated by each newly generated individual. This indicator represents

the effectiveness of newly generated individuals in enhancing the search. If the indicator equals 1.0, each new individual dominates one parent individual. A lower value indicates that the search is converging. The average value of this indicator is then computed over g_{check} generations. If the following equation is satisfied, we conclude that the search has converged:

$$\sum_{i=1}^{g_{check}} \frac{\mu_i}{g_{check}} \leq \varepsilon \tag{3.1}$$

Here, μ_i is the average number of parents dominated by each newly generated individual during the i-th generation, and ε is the indicator threshold. Because this threshold may depend on the target problem, it is investigated in a later section. The search phase of the proposed scheme is switched once Equation (3.1) is satisfied.

Proposed Search Scheme

Abovementioned reference-point-based search is performed to improve the proximity, and the search phase switching criterion is periodically checked. If the population converged, then the population obtained by the first phase would be further improved by DC-scheme regarding the spread as the second phase search. Figure 3.3 illustrates the search procedure of our proposed framework.

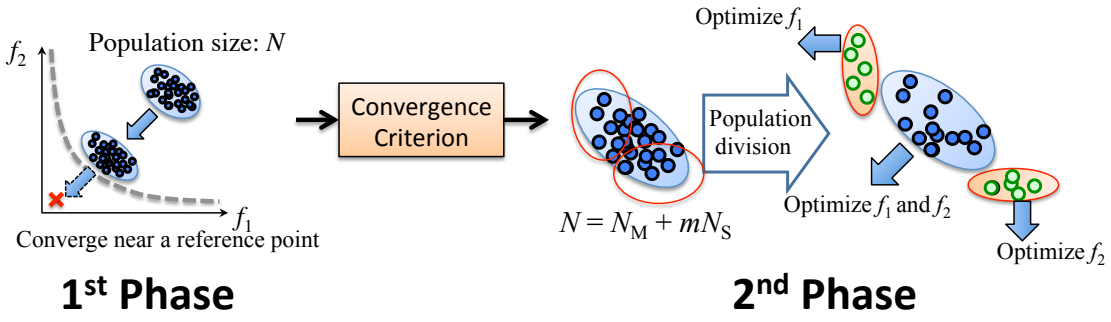


Fig. 3.3 Search procedure of the proposed search scheme

The search process of our proposed scheme is outlined in Algorithm 3.1. Algorithm 3.1 assumes that $N = N_M + \sum_{k=1}^{m} N_{Sk}$. Line 21 preserves the nondominated solutions explored thus far in the external population of fixed size. The effects of the external population on search performance are discussed later (see numerical experimental section). Moreover, the external population update scheme depends on the MOEA implementation chosen for the proposed scheme and the individual selection method (Line 11 in Algorithm 3.1). Both methods are described below. The following two subsections present the implementations of our proposed scheme.

Algorithm 3.1 Proposed search scheme

1: Initialize a population P with N individuals
2: Initialize a reference point
3: **for** $i = 0$ to g
4: Perform a single generation of reference-point-based MOEA
5: **if** $i \bmod g_{check} = 0 \bigwedge$ Equation (3.1) = true
6: $r := i + 1$
7: **break**
8: **end if**
9: **end for**
10: Initialize the external population $EP := P$
11: Select N_M individuals from P and add them to P_M
12: **for** $k = 1$ to m
13: Select the top N_{Sk} individuals in f_k from P and add them to P_{Sk}.
14: **end for**
15: **for** $i = r$ to g
16: Perform a single generation of MOEA with the population P_M
17: Perform a single generation of SOEA with the population P_{Sk} ($k = 1, \ldots, m$)
18: **if** $i \bmod g_{check} = 0$
19: Exchange the elite individuals between P_M and P_{Sk} ($k = 1, \ldots, m$)
20: **end if**
21: Update EP
22: **end for**

3.2.3 Application to the Dominance-Based Algorithm

In this subsection, we apply our proposed scheme to conventional algorithms. We adopt a dominance-based MOEA called the NSGA-II algorithm. Traditional dominance-based MOEAs, such as the strength Pareto evolutionary algorithm 2 (SPEA2) [31], the generalized differential evolution 3 (GDE3) [32], and simulated-annealing-based algorithms [33], are also applicable. The SOEA is the distributed genetic algorithm (DGA) [34].

Modifications to NSGA-II

The first phase of our scheme requires a reference-point-based MOEA, here adapted from an ordinary dominance-based algorithm. Specifically, we specify the distance from the reference point as the mating selection criterion, and modify the mating scheme of the original NSGA-II. The method is detailed below.

Algorithm 3.2 Modified mating selection procedure (population size $= N$)

1: Sort the N individuals in the parent population P in the ascending order of their Euclidean distance from the reference point.
2: Select the top $\frac{N}{2}$ individuals in P and add them to the search population M.
3: Select $\frac{N}{2}$ individuals by tournament selection based on their rank, and add them to M.
4: Perform the evolutionary operations (selection, crossover, and mutation) of NSGA-II using M as the mating pool.

Line 2 in Algorithm 3.2 copies the $\frac{N}{2}$ individuals closest to the reference point to M. Note that probabilistic methods, such as tournament selection, do not guarantee selection of these individuals. By copying them to M, we guide the search population toward convergence near the reference point. In addition, because the tournament selection step at Line 3 is based on fitness rank, the search is directed toward the reference point while preserving the diversity of the search population. If two individuals have the same rank, the algorithm selects the individual with the smallest Euclidean distance. Moreover, if two individuals share both rank and Euclidean distance, one of them is randomly chosen.

Here, the location of the reference point becomes important. If the reference point is located in a feasible region, the proximity may decrease when the population overtakes the reference point. To counteract this decline, the distance to the reference point is set negative when the feasible reference point is overtaken by the population. Consequently, the search guides the population away from the reference point. This check is performed at each generation. We refer to this algorithm as the reference-point-based mating NSGA-II (RM-NSGA-II).

Distributed Genetic Algorithm

DGA, first proposed by Tanase et al [34], is a parallel implementation of the genetic algorithm (GA) [35]. DGA reduces the number of iterations required for optimum solution searching by splitting the population into subpopulations, called islands. The original GA is implemented on each subpopulation. At predefined intervals, islands exchange some of their individuals (this operation is referred to as migration). The number of migrating individuals is determined by the migration rate. DGA adopts a ring migration topology with random destinations, where each subpopulation is assigned a random destination during each migration interval. Premature convergence is prevented by an elite-preservation strategy. Specifically, at each generation, the elite individuals (the best-fit individuals in each objective) are preserved in an elite archive, and are merged with the search population during the selection.

Implementation of the Proposed Search Scheme

In our proposed scheme, the abovementioned search algorithms are incorporated into a new hybrid algorithm. The search flow proceeds via Algorithm 3.1, presented earlier. In the first phase, the search scheme is implemented by the RM-NSGA-II algorithm. In the second phase, the SOEA and MOEA populations are managed by DGA and NSGA-II, respectively. Line 11 of Algorithm 3.1 uses the nondominated sorting scheme of the original NSGA-II to extract the NSGA-II population P_M from population P. Hence, NM nondominated solutions are moved from P to P_M. The selection method of the DGA population is described in subsection 3.2.2. The effectiveness of NSGA-II implemented in our proposed scheme is discussed in Section 3.3.

3.2.4 Application to the Decomposition-based Algorithm

Decomposition-based MOEAs have recently attracted much interest due to their excellent performance on many-objective, high-dimensional, multimodal, and other complex problems with lower computational complexity than conventional MOEAs [36]. Decomposition-based MOEAs originated with Zhang et al. 's MOEA/D algorithm [19]. This framework has become widely incorporated into other EAs, such as particle swarm optimization (PSO) [37] and differential evolution (DE) [38], and the original MOEA/D has been improved [39][40]. In this section, we design an MOEA/D-based hybrid algorithm for implementing our proposed search scheme.

Framework of MOEA/D

To realize a reference point in our MOEA/D, we adopt the Tchebycheff aggregation approach. The multi-objective problems are decomposed into the following scalar single-objective problem [19]:

$$\begin{aligned} &\text{minimize } s^{te}(\boldsymbol{x} \mid \boldsymbol{\lambda}^j, \boldsymbol{z}^*) = \max_{1 \le i \le m} \{\lambda_i^j | f_i(\boldsymbol{x}) - z_i^* |\} \\ &\text{subject to } \boldsymbol{x} \in S \end{aligned} \tag{3.2}$$

where $\boldsymbol{z}^* = (z_1^*, \ldots, z_m^*)^T$ is the reference point. The reference point of the original MOEA/D is a set of extreme points for each objective, i.e., $z_i^* = \min\{f_i(\boldsymbol{x}) \mid \boldsymbol{x} \in S\}$. In Equation (3.2), $\boldsymbol{\lambda}^j$ is an m-dimensional weight vector, where $j = 1 \ldots N$. When the weight vectors are uniformly distributed, each vector component receives a value from $\{\frac{0}{H}, \frac{1}{H}, \ldots, \frac{H}{H}\}$ subject to $|\boldsymbol{\lambda}| = 1$, where H is the number of problem divisions. N is then computed as $N = C_{H+m-1}^{m-1}$. $s^{te}(\boldsymbol{x} \mid \boldsymbol{\lambda}^j, \boldsymbol{z}^*)$ is the Tchebycheff scalarizing function with the weight vector $\boldsymbol{\lambda}^j$ and the reference point $\boldsymbol{z}^*$.

Randomly chosen individuals are recombined from neighboring subproblems. Neighbors are determined by the similarity of the weight vector of each subproblem. The reference point is updated by the generated individuals, and the population is updated from the generated and neighboring individuals. If a newly generated individual defeats its neighbors, it replaces all of them.

Implementation of the Proposed Search Scheme

The first phase uses a MOEA/D with a user-specified reference point. The original MOEA/D automatically sets and updates the reference point. In this paper, the reference point is fixed to ensure consistency with the NSGA-II-based algorithm designed in the previous section. In the second phase, the SOEA and MOEA are DGA and the original MOEA/D, respectively. Hence, our proposed scheme was designed with a MOEA/D-based hybrid algorithm. When initializing the MOEA population in the second phase, the MOEA/D population P_M is randomly extracted from population P, which transfers N_M solutions from P to P_M. The selection method of the DGA population is described in subsection 3.2.2. Moreover, similar to RM-NSGA-II, the weight vectors are rendered negative when the population overtakes a feasible reference point, allowing the population to escape from the reference locality. This check is performed at each generation. The external population is updated according to the generation alternation scheme of the original MOEA/D, as shown below:

Algorithm 3.3 Update scheme of the external population

1: **for all** $\boldsymbol{x}_{new}$ generated in a single generation
2: Select an individual $\boldsymbol{x} \in EP$.
3: Compare $\boldsymbol{x}_{new}$ with $\boldsymbol{x}$ using the Tchebycheff scalar function.
4: **if** $\boldsymbol{x}_{new}$ is better than $\boldsymbol{x}$
5: Update $\boldsymbol{x}$ and the neighboring individuals of $\boldsymbol{x}$ in EP.
6: **end if**
7: **end for**

The effectiveness of MOEA/D implemented in our proposed search scheme is evaluated in Section 3.3.

3.3 Numerical Experiments

In this section, we verify the effectiveness of our proposed scheme by applying the two above-mentioned implementations to multi-objective continuous test problems.

Table 3.1 Properties of problems WFG1–WFG9

Problem	Obj.	Separability	Modality	Bias	Geometry
WFG1	$f_{1:M}$	separable	unimodal	polynomial, flat	convex, mixed
WFG2	$f_{1:M-1}$	non-separable	unimodal	—	convex, disconnected
	f_M	non-separable	multimodal		
WFG3	$f_{1:M}$	non-separable	unimodal	—	linear, degenerate
WFG4	$f_{1:M}$	separable	multimodal	—	concave
WFG5	$f_{1:M}$	separable	deceptive	—	concave
WFG6	$f_{1:M}$	non-separable	unimodal	—	concave
WFG7	$f_{1:M}$	separable	unimodal	parameter dependent	concave
WFG8	$f_{1:M}$	non-separable	unimodal	parameter dependent	concave
WFG9	$f_{1:M}$	non-separable	multimodal, deceptive	parameter dependent	concave

3.3.1 WFG Toolkit

Our search scheme is tested using the WFG toolkit, a set of continuous and scalable test problems defined in terms of a simple underlying problem. The toolkit also specifies a fitness space, and a series of composable and configurable transformations, enabling the addition of arbitrary levels of complexity to the test problem [41][42]. The toolkit contains nine typical problems, WFG1–WFG9 [41], whose properties are listed in Table 3.1. We formulate WFG1–WFG9 as both bi- and tri-objective problems. In both formulations, we set 20 distance-related parameters and 4 position-related parameters.

3.3.2 Performance Metric

The performance of MOEAs is quantified by the inverted generational distance (IGD) [43], hypervolume [14] [44] spread [41], and coverage [45]. Here, we adopt the IGD, defined as the average distance between each Pareto optimal front solution and the closest obtained solution, as a performance index. This metric is suitable for assessing both proximity and diversity. Let P^* be a set of uniform distributed points in the objective space along the Pareto optimal front, and P be a nondominated front obtained by a MOEA. The IGD value is computed as

$$\mathrm{IGD}(P^*, P) = \frac{\sum_{v \in P^*} d(v, P)}{|P^*|} \tag{3.3}$$

where $d(v, P)$ is the minimum Euclidean distance between the solutions of the Pareto optimal front and the solution in P, v. As the IGD approaches 0, the solutions in P approach the Pareto optimal front.

3.3.3 Dominance-Based Algorithm: NSGA-II

We first tested the NSGA-II-based hybrid version of our search scheme on the WFG toolkit functions WFG1–WFG9. The aims of this experiment are listed below:

- To study effects of the parameters for switching the search phase on search performance.
- To compare our algorithm with the original NSGA-II.
- To determine how search performance is affected by the reference point location (for two separate reference point scenarios—within feasible and infeasible regions).
- To study the effect of an external population.
- To compare the computational cost of the proposed search scheme with the original algorithm.

We ran three algorithms (the original NSGA-II, the hybrid algorithm with an external population, and the hybrid algorithm with no external population) on the bi- and tri-objective WFG test problems.

Experimental Setup

The population sizes of the three algorithms were set to 240 and 300 in the bi-objective cases and tri-objective problems, respectively. The same sizes were selected for the external populations. The maximum number of generations was set to 750 and 1000 in the bi- and tri-objective problems, respectively. We also simulated binary crossover (SBX) and polynomial mutation [46].

In the first phase of both hybrid algorithms, the proximity was checked at intervals of 25 generations ($g_{check} = 25$). g_{check} may also depend on the problem and affect the convergence time, but it behaves similarly with the threshold of the search-phase-switching criterion ε. Therefore we fix the g_{check} setting and study the threshold value ε in this paper. Here ε was varied as 0.50, 0.10. 0.05, 0.01. The reference point was set to $(f_1, f_2) = (0, 0)$ and $(f_1, f_2, f_3) = (0, 0, 0)$ in infeasible locations, and to $(f_1, f_2) = (2, 4)$ and $(f_1, f_2, f_3) = (2, 4, 6)$ in feasible locations.

For all tests, the NSGA-II population size in the second phase was set to 120; thus the DGA population size was 60 in every objective. Each DGA population was divided into islands of 10 individuals (giving six islands per DGA population). Every five generations, the DGA exchanged five individuals among six islands. The best-fit individual on each island was preserved as the elite individual. A tournament size of 4 was assigned in the survival selection scheme of DGA.

Moreover, every 25 generations, the NSGA-II population exchanged the best-fit individual in each objective with the DGA population.

To compare the computational cost of each algorithm, all the numerical simulations were performed on a PC with Intel Core 2 Duo 1.6 GHz and 4 GB of DDR3 memory.

Results

Tables 3.2 and 3.3, present the IGD means and standard deviations at various thresholds of the switching criterion, ε, in the bi- and tri-objective WFG problems, respectively. The columns titled " suspended " list the IGD mean and standard deviation of the original algorithm computed at the mean number of generations when the first search phase of the hybrid algorithm terminated. Note that the hybrid algorithm switches its search phase according to the proposed stopping criterion. In order to compare the accuracy of the obtained solutions of the hybrid algorithm with the original one, the original algorithm was suspended at the mean number of generations in the first phase of the hybrid one.

The number of generations at first phase termination is listed for different thresholds of the switching criterion in Tables 3.4 and 3.5. The column titled " pct " shows the ratio of generations in the first-phase search to the maximum number of generations in the entire search. We observe that the optimum threshold depends on the problem. In addition, adjusting the threshold controls the balance between the first- and second-phase generations.

The IGD of our proposed search scheme is improved by an external population, especially in tri-objective problems. Our hybrid algorithm with an external population reduces the mean IGD relative to the original NSGA-II in problems WFG1, WFG3, WFG4, and WFG6. The three algorithms yield approximately the same performance in WFG2. However, our proposed scheme performs worse than the original NSGA-II in the tri-objective WFG5, WFG7, and WFG9 problems, and in both the bi- and tri-objective WFG8 problems. The standard deviations of the IGDs reveal that the proposed scheme is more robust than the original.

According to the results, the performance of our proposed scheme is insensitive to the location of the reference point. Although the reference points $(f_1, f_2) = (2, 4)$ and $(f_1, f_2, f_3) = (2, 4, 6)$ are easily overtaken by the search population early in the first-phase search, assigning a negative sign to the reference point distance prevents the population from becoming trapped in the reference region. For each algorithm applied to the tri-objective case of WFG1, the nondominated solutions yielded by the lowest IGD run are presented in Fig. 3.4. From this figure, we observe that after the first phase, the converged area differs for each reference point. However, the overall performance of our proposed scheme is not significantly altered. Successful and unsuccessful

Table 3.2 IGD values of the solutions found by the original NSGA-II and NSGA-II with the proposed search scheme, in the bi-objective case. Standard deviations are given in parentheses.

Problem	External Population	With Proposed Search Scheme																Original suspended	Original final
		Reference Point $(f_1, f_2) = (0, 0)$								Reference Point $(f_1, f_2) = (2, 4)$									
		$\varepsilon = 0.5$		$\varepsilon = 0.1$		$\varepsilon = 0.05$		$\varepsilon = 0.01$		$\varepsilon = 0.5$		$\varepsilon = 0.1$		$\varepsilon = 0.05$		$\varepsilon = 0.01$			
		1st	2nd	1st	2nd	1st	2nd	1st	2nd	1st	2nd	1st	2nd	1st	2nd	1st	2nd		
WFG1	False	1.83	0.17	1.82	0.17	1.82	0.17	1.80	0.17	2.45	0.17	2.47	0.17	2.47	0.17	2.47	0.17	2.14	1.73
		(0.02)	(0.00)	(0.02)	(0.00)	(0.02)	(0.00)	(0.09)	(0.00)	(0.14)	(0.01)	(0.11)	(0.01)	(0.11)	(0.01)	(0.11)	(0.01)	(0.06)	(0.04)
	True		0.17		0.17		0.17		0.17		0.17		0.17		0.17		0.17		
			(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		
WFG2	False	0.19	0.12	0.18	0.12	0.18	0.12	0.19	0.13	0.19	0.12	0.19	0.12	0.19	0.12	0.19	0.12	0.12	0.12
		(0.01)	(0.00)	(0.01)	(0.00)	(0.01)	(0.00)	(0.01)	(0.03)	(0.02)	(0.00)	(0.02)	(0.00)	(0.02)	(0.00)	(0.02)	(0.02)	(0.00)	(0.00)
	True		0.12		0.12		0.12		0.13		0.12		0.12		0.12		0.12		
			(0.00)		(0.00)		(0.00)		(0.03)		(0.00)		(0.00)		(0.00)		(0.02)		
WFG3	False	0.73	0.20	0.71	0.20	0.70	0.20	0.54	0.27	0.34	0.20	0.32	0.20	0.30	0.20	0.24	0.22	0.30	0.21
		(0.05)	(0.00)	(0.06)	(0.00)	(0.06)	(0.00)	(0.07)	(0.13)	(0.02)	(0.00)	(0.03)	(0.00)	(0.03)	(0.00)	(0.04)	(0.04)	(0.02)	(0.00)
	True		0.19		0.19		0.19		0.27		0.19		0.19		0.19		0.22		
			(0.00)		(0.00)		(0.00)		(0.13)		(0.00)		(0.00)		(0.00)		(0.04)		
WFG4	False	0.07	0.01	0.04	0.01	0.02	0.01	0.01	0.01	0.06	0.01	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.01
		(0.01)	(0.00)	(0.01)	(0.00)	(0.01)	(0.00)	(0.00)	(0.00)	(0.03)	(0.00)	(0.01)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)
	True		0.01		0.01		0.01		0.01		0.01		0.01		0.01		0.01		
			(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		
WFG5	False	0.10	0.07	0.08	0.07	0.08	0.07	0.07	0.07	0.13	0.07	0.10	0.07	0.08	0.07	0.07	0.07	0.07	0.07
		(0.01)	(0.00)	(0.01)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.05)	(0.00)	(0.04)	(0.00)	(0.01)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)
	True		0.07		0.07		0.07		0.07		0.07		0.07		0.07		0.07		
			(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		
WFG6	False	0.07	0.02	0.05	0.02	0.03	0.02	0.02	0.02	0.31	0.03	0.30	0.03	0.28	0.03	0.12	0.03	0.03	0.02
		(0.02)	(0.01)	(0.02)	(0.01)	(0.02)	(0.02)	(0.02)	(0.02)	(0.08)	(0.02)	(0.09)	(0.02)	(0.11)	(0.02)	(0.17)	(0.02)	(0.02)	(0.02)
	True		0.02		0.02		0.02		0.02		0.02		0.02		0.02		0.03		
			(0.01)		(0.01)		(0.02)		(0.02)		(0.02)		(0.02)		(0.02)		(0.02)		
WFG7	False	0.09	0.01	0.08	0.01	0.07	0.01	0.01	0.01	0.12	0.01	0.10	0.01	0.09	0.01	0.03	0.01	0.03	0.03
		(0.02)	(0.00)	(0.02)	(0.00)	(0.02)	(0.00)	(0.01)	(0.00)	(0.03)	(0.00)	(0.03)	(0.00)	(0.02)	(0.00)	(0.01)	(0.00)	(0.01)	(0.01)
	True		0.01		0.01		0.01		0.01		0.01		0.01		0.01		0.01		
			(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		(0.01)		
WFG8	False	0.21	0.07	0.17	0.17	0.17	0.17	0.17	0.17	0.18	0.04	0.17	0.17	0.17	0.17	0.17	0.17	0.05	0.04
		(0.10)	(0.06)	(0.07)	(0.07)	(0.07)	(0.07)	(0.07)	(0.07)	(0.03)	(0.01)	(0.03)	(0.03)	(0.03)	(0.03)	(0.03)	(0.03)	(0.02)	(0.02)
	True		0.05		0.17		0.17		0.17		0.04		0.17		0.17		0.17		
			(0.05)		(0.07)		(0.07)		(0.07)		(0.01)		(0.03)		(0.03)		(0.03)		
WFG9	False	0.04	0.01	0.02	0.01	0.02	0.01	0.01	0.01	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
		(0.01)	(0.00)	(0.01)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.01)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)
	True		0.01		0.01		0.01		0.01		0.01		0.01		0.01		0.01		
			(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		(0.00)		

runs of our proposed scheme are further discussed in subsection 3.3.5.

3.3.4 Decomposition-Based Algorithm: MOEA/D

We then tested our proposed scheme using the MOEA/D-based hybrid algorithm on WFG1–WFG9. The aims of this experiment are listed below:

- To study the effects of the parameters for switching the search phase on search performance.
- To compare our algorithm with the original MOEA/D.
- To determine the effect of an external population.
- To compare the computational cost of the proposed search scheme with the original algorithm.

As before, we ran three algorithms (the original MOEA/D, the hybrid algorithm with an external population, and the hybrid algorithm with no external population) on the bi- and tri-objective WFG test problems.

Table 3.3 IGD values of the solutions found by the original NSGA-II and NSGA-II with the proposed search scheme, in the tri-objective case. Standard deviations are given in parentheses.

		With Proposed Search Scheme																	
		Reference Point $(f_1, f_2, f_3) = (0, 0, 0)$								Reference Point $(f_1, f_2, f_3) = (2, 4, 6)$									
Problem	External	$\varepsilon = 0.5$		$\varepsilon = 0.1$		$\varepsilon = 0.05$		$\varepsilon = 0.01$		$\varepsilon = 0.5$		$\varepsilon = 0.1$		$\varepsilon = 0.05$		$\varepsilon = 0.01$		Original	
	Population	1st	2nd	1st	2nd	1st	2nd	1st	2nd	1st	2nd	1st	2nd	1st	2nd	1st	2nd	suspended	final
WFG1	False	1.71	0.58	1.69	0.58	1.67	0.59	1.64	0.58	2.45	0.61	2.45	0.61	2.45	0.61	2.45	0.58	1.89	1.46
		(0.04)	(0.05)	(0.04)	(0.05)	(0.04)	(0.06)	(0.03)	(0.06)	(0.02)	(0.13)	(0.02)	(0.13)	(0.02)	(0.13)	(0.02)	(0.04)	(0.06)	(0.03)
	True		0.55		0.55		0.55		0.55		0.55		0.55		0.55		0.55		
			(0.06)		(0.06)		(0.06)		(0.06)		(0.04)		(0.04)		(0.04)		(0.05)		
WFG2	False	0.38	0.22	0.41	0.22	0.43	0.21	0.42	0.38	0.39	0.29	0.39	0.27	0.39	0.28	0.37	0.36	0.27	0.20
		(0.08)	(0.05)	(0.12)	(0.05)	(0.10)	(0.05)	(0.11)	(0.11)	(0.05)	(0.09)	(0.04)	(0.08)	(0.04)	(0.09)	(0.07)	(0.09)	(0.09)	(0.09)
	True		0.20		0.20		0.20		0.38		0.27		0.27		0.27		0.35		
			(0.06)		(0.06)		(0.05)		(0.11)		(0.09)		(0.09)		(0.09)		(0.09)		
WFG3	False	0.29	0.02	0.26	0.02	0.24	0.02	0.11	0.02	0.27	0.02	0.27	0.02	0.27	0.02	0.20	0.02	0.02	0.01
		(0.07)	(0.00)	(0.05)	(0.00)	(0.05)	(0.00)	(0.05)	(0.02)	(0.04)	(0.00)	(0.04)	(0.00)	(0.04)	(0.00)	(0.07)	(0.00)	(0.01)	(0.01)
	True		0.01		0.01		0.01		0.01		0.01		0.01		0.01		0.01		
			(0.00)		(0.00)		(0.00)		(0.02)		(0.00)		(0.00)		(0.00)		(0.00)		
WFG4	False	0.35	0.21	0.31	0.21	0.34	0.21	0.46	0.45	0.40	0.22	0.29	0.20	0.29	0.21	0.44	0.44	0.24	0.16
		(0.03)	(0.03)	(0.02)	(0.03)	(0.03)	(0.04)	(0.01)	(0.03)	(0.04)	(0.04)	(0.03)	(0.03)	(0.04)	(0.02)	(0.05)	(0.05)	(0.02)	(0.03)
	True		0.14		0.14		0.15		0.45		0.15		0.16		0.14		0.44		
			(0.02)		(0.02)		(0.03)		(0.04)		(0.03)		(0.03)		(0.02)		(0.05)		
WFG5	False	0.46	0.27	0.39	0.27	0.39	0.27	0.39	0.28	0.43	0.27	0.42	0.28	0.43	0.27	0.48	0.27	0.26	0.18
		(0.04)	(0.01)	(0.05)	(0.02)	(0.05)	(0.01)	(0.08)	(0.02)	(0.04)	(0.01)	(0.05)	(0.02)	(0.07)	(0.01)	(0.20)	(0.03)	(0.02)	(0.01)
	True		0.20		0.20		0.20		0.21		0.20		0.20		0.21		0.20		
			(0.01)		(0.01)		(0.01)		(0.03)		(0.01)		(0.01)		(0.01)		(0.01)		
WFG6	False	0.45	0.24	0.39	0.24	0.38	0.25	0.39	0.38	0.43	0.24	0.31	0.24	0.27	0.25	0.21	0.19	0.38	0.19
		(0.05)	(0.01)	(0.07)	(0.01)	(0.07)	(0.03)	(0.48)	(0.48)	(0.07)	(0.03)	(0.10)	(0.03)	(0.10)	(0.02)	(0.13)	(0.05)	(0.05)	(0.02)
	True		0.18		0.18		0.18		0.34		0.18		0.18		0.17		0.18		
			(0.02)		(0.02)		(0.02)		(0.50)		(0.03)		(0.03)		(0.03)		(0.04)		
WFG7	False	0.48	0.35	0.50	0.35	0.50	0.35	0.48	0.47	0.51	0.35	0.48	0.34	0.47	0.34	0.36	0.36	0.36	0.26
		(0.02)	(0.02)	(0.03)	(0.02)	(0.03)	(0.02)	(0.03)	(0.04)	(0.04)	(0.02)	(0.04)	(0.02)	(0.05)	(0.02)	(0.04)	(0.03)	(0.02)	(0.01)
	True		0.29		0.29		0.29		0.46		0.28		0.29		0.28		0.35		
			(0.02)		(0.01)		(0.01)		(0.06)		(0.02)		(0.02)		(0.02)		(0.04)		
WFG8	False	0.92	0.64	0.89	0.71	0.85	0.72	0.79	0.79	0.73	0.44	0.52	0.50	0.51	0.51	0.51	0.51	0.47	0.26
		(0.04)	(0.15)	(0.04)	(0.07)	(0.03)	(0.04)	(0.02)	(0.02)	(0.07)	(0.13)	(0.18)	(0.17)	(0.19)	(0.19)	(0.19)	(0.19)	(0.03)	(0.02)
	True		0.59		0.66		0.67		0.79		0.36		0.46		0.50		0.51		
			(0.17)		(0.11)		(0.09)		(0.02)		(0.16)		(0.20)		(0.18)		(0.19)		
WFG9	False	0.27	0.25	0.26	0.24	0.26	0.24	0.41	0.43	0.24	0.23	0.21	0.23	0.20	0.23	0.15	0.15	0.20	0.15
		(0.03)	(0.03)	(0.05)	(0.02)	(0.05)	(0.02)	(0.27)	(0.25)	(0.03)	(0.02)	(0.04)	(0.02)	(0.04)	(0.02)	(0.01)	(0.01)	(0.02)	(0.01)
	True		0.18		0.18		0.18		0.41		0.17		0.17		0.17		0.15		
			(0.02)		(0.02)		(0.02)		(0.27)		(0.01)		(0.02)		(0.02)		(0.01)		

Table 3.4 Number of generations when the first phase of NSGA-II was terminated in the proposed scheme, in the bi-objective case. " pct " denotes the ratio of generations in the first-phase search to the maximum number of generations.

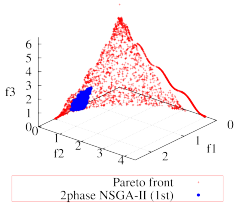
	With Proposed Search Scheme															
	Reference Point $(f_1, f_2) = (0, 0)$								Reference Point $(f_1, f_2) = (2, 4)$							
Problem	$\varepsilon = 0.5$		$\varepsilon = 0.1$		$\varepsilon = 0.05$		$\varepsilon = 0.01$		$\varepsilon = 0.5$		$\varepsilon = 0.1$		$\varepsilon = 0.05$		$\varepsilon = 0.01$	
	#gen	pct	#gen	pct	#gen	pct	#gen	pct	#gen	pct	#gen	pct	#gen	pct	#gen	pct
WFG1	129	17.2	132	17.6	136	18.1	148	19.8	148	19.8	153	20.3	154	20.6	164	21.9
WFG2	80	10.7	118	15.7	176	23.4	524	69.8	78	10.3	110	14.7	143	19.0	439	58.6
WFG3	51	6.8	78	10.3	103	13.7	508	67.7	52	6.9	88	11.7	132	17.6	608	81.1
WFG4	50	6.7	79	10.6	127	16.9	488	65.0	51	6.8	97	12.9	166	22.1	602	80.3
WFG5	49	6.6	70	9.3	82	10.9	332	44.2	50	6.7	77	10.2	99	13.2	399	53.3
WFG6	51	6.8	83	11.0	138	18.3	640	85.4	76	10.1	105	14.0	130	17.3	363	48.3
WFG7	50	6.7	63	8.3	81	10.8	563	75.0	50	6.7	75	10.0	94	12.6	565	75.4
WFG8	319	42.6	750	100.0	750	100.0	750	100.0	404	53.9	750	100.0	750	100.0	750	100.0
WFG9	48	6.4	87	11.6	133	17.7	584	77.8	50	6.7	88	11.7	123	16.3	507	67.6

Experimental Setup

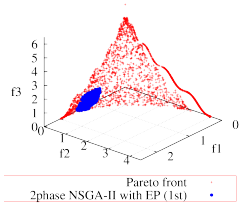
The parameter settings were identical to those in the NSGA-II experiment (subsection 3.3). However, since the effect of reference point location had been clarified in the previous experiment, it was not assessed here. Therefore, the reference point was fixed to $(f_1, f_2) = (0, 0)$ and $(f_1, f_2, f_3) = (0, 0, 0)$. Notably, the code of MOEA/D was written by ourselves, and its

Table 3.5 Number of the generation when the first phase of NSGA-II with the proposed search scheme was terminated, in tri-objective case. 'pct' represents the ratio of 1st-phase search to the maximum number of generations.

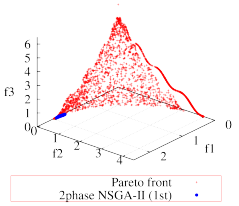
Problem	With Proposed Search Scheme															
	Reference Point $(f_1, f_2) = (0, 0, 0)$								Reference Point $(f_1, f_2) = (2, 4, 6)$							
	$\varepsilon = 0.5$		$\varepsilon = 0.1$		$\varepsilon = 0.05$		$\varepsilon = 0.01$		$\varepsilon = 0.5$		$\varepsilon = 0.1$		$\varepsilon = 0.05$		$\varepsilon = 0.01$	
	#gen	pct	#gen	pct	#gen	pct	#gen	pct	#gen	pct	#gen	pct	#gen	pct	#gen	pct
WFG1	131	13.1	135	13.5	142	14.2	156	15.6	153	15	153	15.3	156	15.6	163	16.3
WFG2	50	5.0	86	8.6	131	13.1	958	95.8	50	5.0	78	7.8	104	10.4	918	91.8
WFG3	49	4.9	73	7.3	98	9.8	606	60.6	50	5.0	81	8.1	113	11.3	486	48.6
WFG4	25	2.5	50	5.0	100	10.0	997	99.7	25	2.5	55	5.5	100	10.0	1000	100.0
WFG5	25	2.5	50	5.0	58	5.8	268	26.8	25	2.5	50	5.0	79	7.9	278	27.8
WFG6	25	2.5	51	5.1	69	6.9	646	64.6	28	2.8	58	5.8	86	8.6	902	90.2
WFG7	25	2.5	50	5.0	55	5.5	962	96.2	25	2.5	50	5.0	73	7.3	960	96.0
WFG8	25	2.5	53	5.3	167	16.7	1000	100.0	25	2.5	278	27.8	949	94.9	1000	100.0
WFG9	25	2.5	50	5.0	50	5.0	725	72.5	25	2.5	50	5.0	53	5.3	1000	100.0



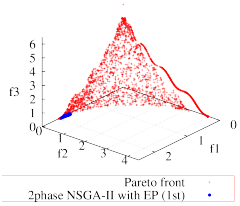
(a) 2-phase NSGA-II without EP (1st phase), Reference: (0, 0, 0)



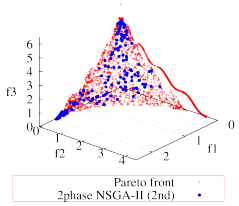
(b) 2-phase NSGA-II with EP (1st phase), Reference: (0, 0, 0)



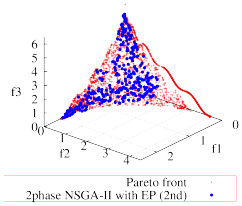
(c) 2-phase NSGA-II without EP (1st phase), Reference: (2, 4, 6)



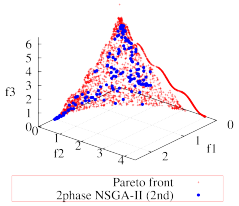
(d) 2-phase NSGA-II with EP (1st phase), Reference: (2, 4, 6)



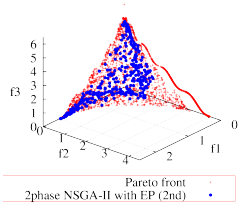
(e) 2-phase NSGA-II without EP (2nd phase), Reference: (0, 0, 0)



(f) 2-phase NSGA-II with EP (2nd phase), Reference: (0, 0, 0)



(g) 2-phase NSGA-II without EP (2nd phase), Reference: (2, 4, 6)



(h) 2-phase NSGA-II with EP (2nd phase), Reference: (2, 4, 6)

Fig. 3.4 Plots of the nondominated solutions with the lowest IGD-metric in 30 runs, with two different reference point settings, for the tri-objective WFG1.

performance was verified by comparison with jMetal [47] implementation of MOEA/D.

Results

The IGD means and standard deviations at various thresholds ε of the switching criterion are displayed in Tables 3.6 and 3.7 for the bi- and tri-objective WFG problems, respectively. Table 3.8 presents the number of generations when the first phase of the search terminates at different thresholds of search-phase switching.

Table 3.6 IGD values of the solutions found by the original MOEA/D and MOEA/D in the proposed search scheme, for the bi-objective case. Standard deviations are given in parentheses.

Problem	With Proposed Search Scheme										
	External Population	$\varepsilon = 0.5$		$\varepsilon = 0.1$		$\varepsilon = 0.05$		$\varepsilon = 0.01$		Original	
		1st	2nd	1st	2nd	1st	2nd	1st	2nd	suspended	final
WFG1	False	0.38	0.34	0.28	0.27	0.25	0.25	0.24	0.22	1.94	1.51
		(0.13)	(0.19)	(0.07)	(0.11)	(0.06)	(0.11)	(0.05)	(0.09)	(0.08)	(0.17)
	True		0.37		0.28		0.25		0.24		
			(0.12)		(0.07)		(0.06)		(0.05)		
WFG2	False	0.18	0.13	0.15	0.13	0.14	0.13	0.14	0.13	0.30	0.24
		(0.05)	(0.01)	(0.03)	(0.01)	(0.02)	(0.01)	(0.02)	(0.01)	(0.15)	(0.16)
	True		0.18		0.15		0.14		0.14		
			(0.05)		(0.03)		(0.02)		(0.02)		
WFG3	False	0.22	0.20	0.22	0.20	0.21	0.20	0.21	0.20	0.57	0.20
		(0.02)	(0.01)	(0.01)	(0.00)	(0.01)	(0.00)	(0.01)	(0.00)	(0.07)	(0.00)
	True		0.20		0.20		0.20		0.20		
			(0.01)		(0.01)		(0.01)		(0.00)		
WFG4	False	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.03	0.01
		(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.01)	(0.00)
	True		0.01		0.01		0.01		0.01		
			(0.00)		(0.00)		(0.00)		(0.00)		
WFG5	False	0.07	0.07	0.07	0.07	0.07	0.07	0.07	0.07	0.13	0.07
		(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.04)	(0.00)
	True		0.07		0.07		0.07		0.07		
			(0.00)		(0.00)		(0.00)		(0.00)		
WFG6	False	0.11	0.11	0.11	0.11	0.11	0.11	0.11	0.10	0.35	0.12
		(0.09)	(0.09)	(0.09)	(0.09)	(0.09)	(0.09)	(0.09)	(0.09)	(0.17)	(0.09)
	True		0.11		0.11		0.11		0.11		
			(0.09)		(0.09)		(0.09)		(0.09)		
WFG7	False	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.17	0.01
		(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.00)	(0.04)	(0.01)
	True		0.01		0.01		0.01		0.01		
			(0.00)		(0.00)		(0.00)		(0.00)		
WFG8	False	0.18	0.22	0.17	0.21	0.16	0.16	0.16	0.16	0.53	0.46
		(0.11)	(0.23)	(0.10)	(0.23)	(0.10)	(0.12)	(0.10)	(0.12)	(0.16)	(0.20)
	True		0.17		0.17		0.18		0.17		
			(0.10)		(0.11)		(0.11)		(0.11)		
WFG9	False	0.03	0.02	0.03	0.02	0.03	0.02	0.03	0.02	0.04	0.01
		(0.02)	(0.01)	(0.02)	(0.01)	(0.02)	(0.01)	(0.02)	(0.01)	(0.03)	(0.01)
	True		0.03		0.03		0.03		0.02		
			(0.02)		(0.02)		(0.02)		(0.02)		

Again, the external population improves the mean IGD of our proposed search scheme, espe-

Table 3.7 IGD values of the solutions found by the original MOEA/D and MOEA/D in the proposed search scheme, for the tri-objective case. Standard deviations are given in parentheses.

Problem	With Proposed Search Scheme									Original	
	External Population	$\varepsilon = 0.5$		$\varepsilon = 0.1$		$\varepsilon = 0.05$		$\varepsilon = 0.01$			
		1st	2nd	1st	2nd	1st	2nd	1st	2nd	suspended	final
WFG1	False	0.74	0.45	0.69	0.43	0.67	0.42	0.68	0.41	1.52	0.81
		(0.11)	(0.08)	(0.15)	(0.07)	(0.15)	(0.05)	(0.15)	(0.04)	(0.09)	(0.12)
	True		0.49		0.46		0.44		0.44		
			(0.07)		(0.06)		(0.06)		(0.06)		
WFG2	False	0.59	0.45	0.64	0.50	0.64	0.51	0.65	0.53	0.56	0.57
		(0.12)	(0.11)	(0.10)	(0.12)	(0.10)	(0.12)	(0.10)	(0.13)	(0.11)	(0.10)
	True		0.55		0.63		0.63		0.64		
			(0.13)		(0.10)		(0.10)		(0.10)		
WFG3	False	0.10	0.07	0.05	0.07	0.05	0.07	0.05	0.07	0.23	0.03
		(0.04)	(0.00)	(0.02)	(0.00)	(0.02)	(0.00)	(0.02)	(0.00)	(0.07)	(0.00)
	True		0.04		0.03		0.03		0.03		
			(0.02)		(0.01)		(0.01)		(0.01)		
WFG4	False	0.19	0.35	0.19	0.35	0.19	0.35	0.19	0.35	0.23	0.21
		(0.02)	(0.01)	(0.01)	(0.01)	(0.01)	(0.01)	(0.01)	(0.01)	(0.03)	(0.00)
	True		0.19		0.19		0.19		0.19		
			(0.01)		(0.01)		(0.01)		(0.01)		
WFG5	False	0.23	0.29	0.20	0.29	0.20	0.29	0.20	0.29	0.28	0.19
		(0.02)	(0.01)	(0.01)	(0.01)	(0.01)	(0.01)	(0.01)	(0.01)	(0.05)	(0.00)
	True		0.22		0.20		0.20		0.20		
			(0.01)		(0.00)		(0.00)		(0.00)		
WFG6	False	0.34	0.32	0.28	0.32	0.28	0.32	0.28	0.32	0.46	0.27
		(0.09)	(0.06)	(0.08)	(0.06)	(0.08)	(0.06)	(0.08)	(0.06)	(0.21)	(0.08)
	True		0.30		0.27		0.27		0.27		
			(0.08)		(0.08)		(0.08)		(0.08)		
WFG7	False	0.22	0.27	0.18	0.27	0.18	0.27	0.18	0.27	0.20	0.16
		(0.04)	(0.00)	(0.01)	(0.00)	(0.01)	(0.00)	(0.01)	(0.00)	(0.03)	(0.00)
	True		0.20		0.19		0.18		0.18		
			(0.02)		(0.01)		(0.01)		(0.01)		
WFG8	False	0.40	0.41	0.40	0.40	0.39	0.41	0.39	0.41	0.68	0.50
		(0.16)	(0.14)	(0.16)	(0.14)	(0.16)	(0.14)	(0.17)	(0.14)	(0.23)	(0.24)
	True		0.38		0.38		0.38		0.37		
			(0.15)		(0.15)		(0.15)		(0.15)		
WFG9	False	0.24	0.27	0.21	0.28	0.21	0.28	0.21	0.27	0.24	0.20
		(0.05)	(0.03)	(0.03)	(0.02)	(0.03)	(0.02)	(0.03)	(0.02)	(0.09)	(0.08)
	True		0.23		0.21		0.21		0.21		
			(0.03)		(0.02)		(0.02)		(0.02)		

cially in the tri-objective cases. In the WFG1, WFG2, WFG4, WFG6, and WFG8 problems, our proposed search scheme with the hybrid algorithm and an external population achieves a lower mean IGD than the original MOEA/D. The performances of the three algorithms are approximately equal in the WFG3 problem. The proposed scheme performs worse than the original algorithm in the tri-objective WFG5 and WGF7 problems, and in the bi- and tri-objective WFG9 problems. As found for the NSGA-II-based algorithm, the proposed scheme is more robust (ex-

Table 3.8 Number of generations at termination of the first phase of the proposed scheme in the MOEA/D-based implementations. " pct " denotes the ratio of generation number in the first phase search to the maximum number of generations.

	Bi-objective								Tri-objective							
	With Proposed Search Scheme								With Proposed Search Scheme							
Problem	$\varepsilon = 0.5$		$\varepsilon = 0.1$		$\varepsilon = 0.05$		$\varepsilon = 0.01$		$\varepsilon = 0.5$		$\varepsilon = 0.1$		$\varepsilon = 0.05$		$\varepsilon = 0.01$	
	#gen	pct	#gen	pct	#gen	pct	#gen	pct	#gen	pct	#gen	pct	#gen	pct	#gen	pct
WFG1	229	30.6	272	36.2	278	37.1	288	38.3	178	17.8	212	21.2	222	22.2	223	22.3
WFG2	72	9.6	134	17.9	149	19.9	163	21.7	457	45.7	729	72.9	756	75.6	790	79.0
WFG3	50	6.7	58	7.8	64	8.6	68	9.0	27	2.7	50	5.0	50	5.0	50	5.0
WFG4	50	6.7	58	7.7	62	8.2	66	8.8	39	3.9	50	5.0	50	5.0	50	5.0
WFG5	50	6.7	50	6.7	50	6.7	51	6.8	27	2.7	50	5.0	50	5.0	50	5.0
WFG6	50	6.7	52	6.9	54	7.2	57	7.6	27	2.7	50	5.0	50	5.0	50	5.0
WFG7	87	11.6	106	14.1	108	14.3	114	15.2	38	3.8	53	5.3	53	5.3	63	6.3
WFG8	80	10.7	118	15.7	125	16.7	143	19.0	49	4.9	53	5.3	55	5.5	59	5.9
WFG9	48	6.4	57	7.6	59	7.9	62	8.2	28	2.8	50	5.0	50	5.0	52	5.2

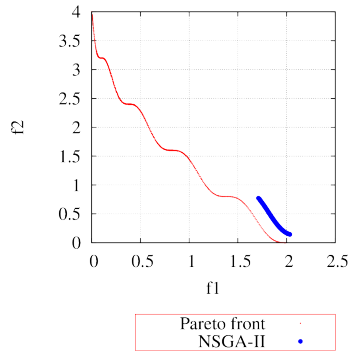
hibits lower IGD standard deviation) than the original algorithm. The search behavior of our proposed scheme is discussed in subsection 3.3.5.

3.3.5 Discussions

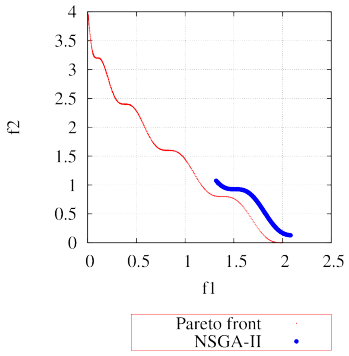
NSGA-II-based implementation

First, we discuss the search behavior of the NSGA-II-based implementation of the proposed search scheme. Figures 3.5, 3.7, and 3.6 depict the nondominated fronts, for which our search scheme improves the IGD values when applied to the bi- and tri-objective WFG1 problems and the bi-objective WFG7 problem, respectively. In these figures, the original NSGA-II, NSGA-II implemented in our search scheme, and NSGA-II implemented in our proposed search scheme with an external population, are respectively labeled " NSGA-II, "" 2-phase NSGA-II, "and " 2-phase NSGA-II with EP. "Figure 3.8 depicts the run for which the IGD value is worsened by our scheme, when applied to the tri-objective WFG7 problem. As shown in Figures 3.5, 3.6 and 3.7, the populations of our proposed scheme converge near the Pareto solutions during the first phase. Moreover, in the second phase, the populations spread more widely than in the original algorithm. Spread is limited in the original algorithm because proximity and diversity are simultaneously implemented from an early stage of the search. Our search strategy, proximity searching followed by population spreading, works very well in the WFG1 and bi-objective WFG7 problems.

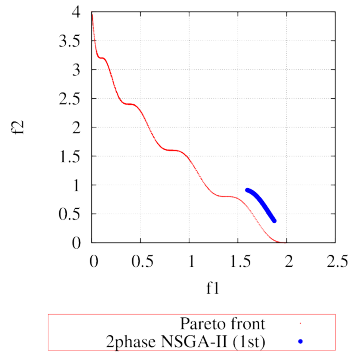
In Figure 3.8, the original NSGA-II algorithm yields a better population spread than our search scheme using the hybrid algorithm. Therefore, the performance of our proposed scheme is degraded by inadequate searching by the SOEA population in the second phase. We infer that the spread of the population affects the searching ability of SOEA, because the ability to find the



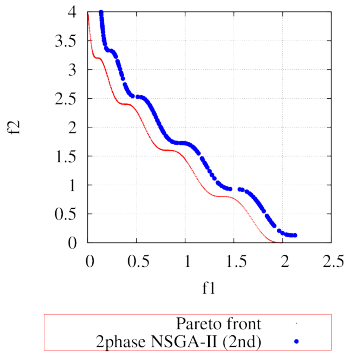
(a) NSGA-II (stopped at 132-th gen.)



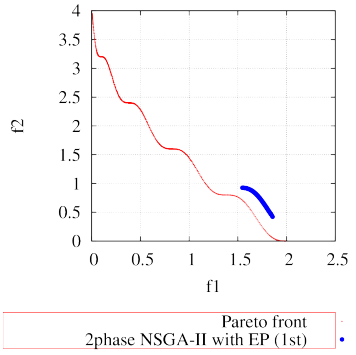
(b) NSGA-II



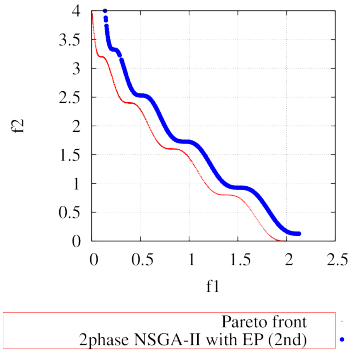
(c) 2-phase NSGA-II without EP (1st phase)



(d) 2-phase NSGA-II without EP (2nd phase)



(e) 2-phase NSGA-II with EP (1st phase)

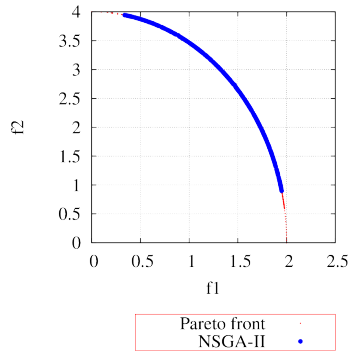


(f) 2-phase NSGA-II with EP (2nd phase)

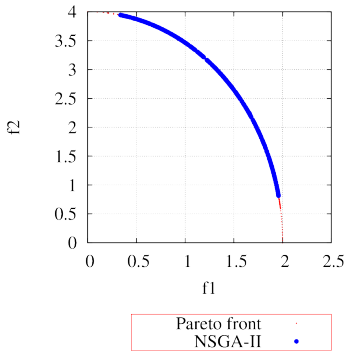
Fig. 3.5 Plots of the nondominated solutions with the lowest IGD-metric in 30 runs of NSGA-II and NSGA-II in the proposed search scheme (with/without external population), for the bi-objective WFG1.

extreme point depends on the performance of SOEA.

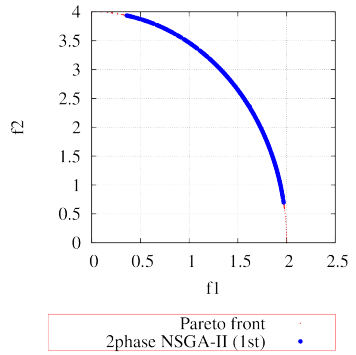
Figure 3.9 shows the final solutions obtained by the best runs of the proposed search scheme



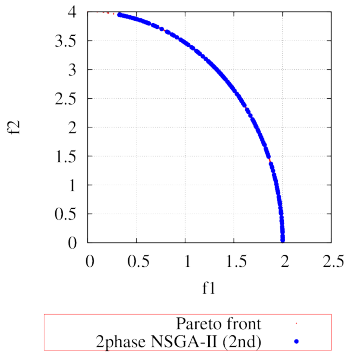
(a) NSGA-II (stopped at 563-th gen.)



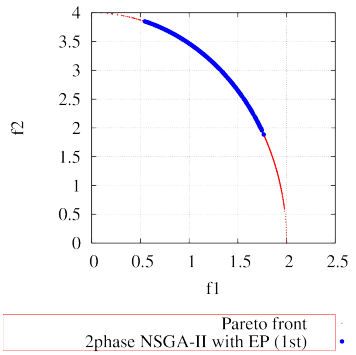
(b) NSGA-II



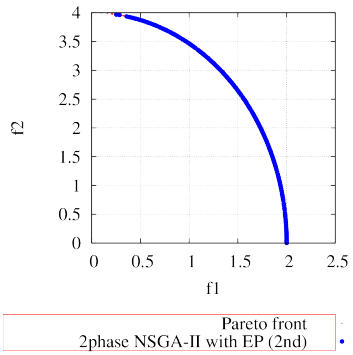
(c) 2-phase NSGA-II without EP (1st phase)



(d) 2-phase NSGA-II without EP (2nd phase)



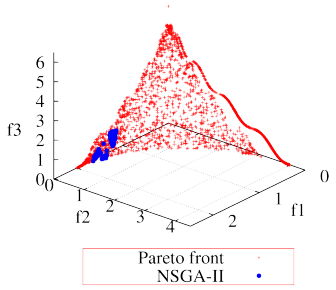
(e) 2-phase NSGA-II with EP (1st phase)



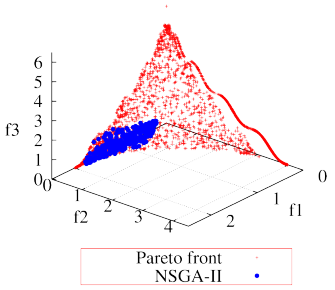
(f) 2-phase NSGA-II with EP (2nd phase)

Fig. 3.6 Plots of the nondominated solutions with the lowest IGD-metric in 30 runs of NSGA-II and NSGA-II in the proposed search scheme (with/without external population), for the bi-objective WFG7.

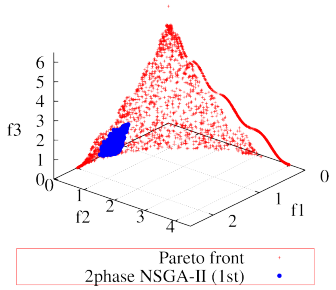
(with the external population) and the original algorithm in tri-objective WFG8. In the figures, the areas near the f_1-extreme point (indicated as 'B') and the f_2-extreme point (indicated as 'A') are



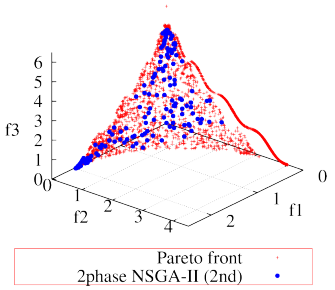
(a) NSGA-II (stopped at 50-th gen.)



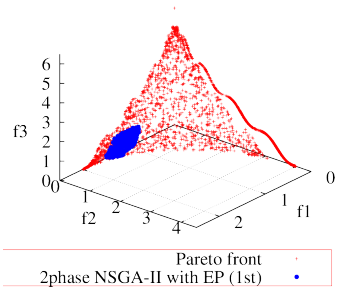
(b) NSGA-II



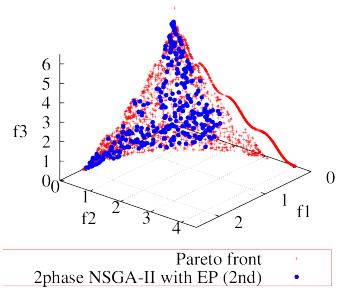
(c) 2-phase NSGA-II without EP (1st phase)



(d) 2-phase NSGA-II without EP (2nd phase)



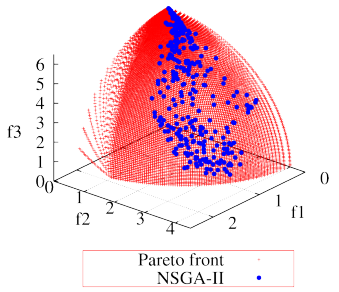
(e) 2-phase NSGA-II with EP (1st phase)



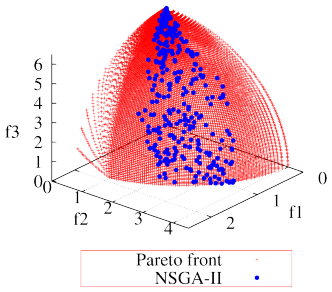
(f) 2-phase NSGA-II with EP (2nd phase)

Fig. 3.7 Plots of the nondominated solutions with the lowest IGD-metric in 30 runs of NSGA-II and NSGA-II with the proposed search scheme (with/without external population), for tri-objective WFG1.

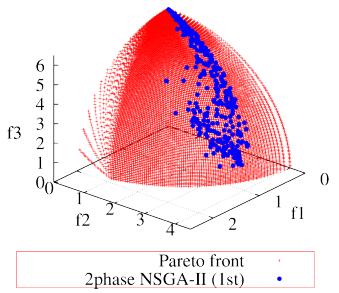
enlarged. From Figure 3.9, it can be revealed that in the best run our search scheme successfully obtains the extreme point in a single objective function, while the original one can not find it.



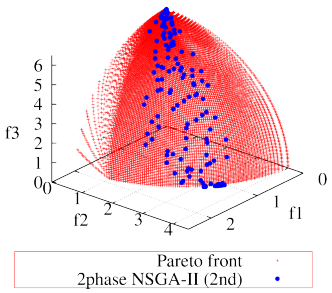
(a) NSGA-II (stopped at 50-th gen.)



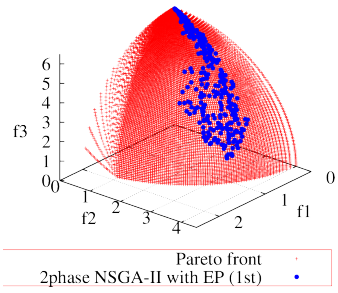
(b) NSGA-II



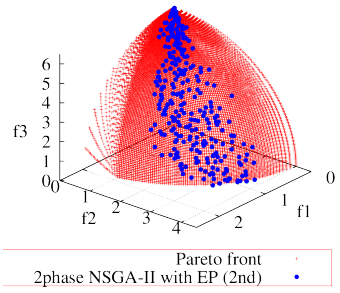
(c) 2-phase NSGA-II without EP (1st phase)



(d) 2-phase NSGA-II without EP (2nd phase)



(e) 2-phase NSGA-II with EP (1st phase)

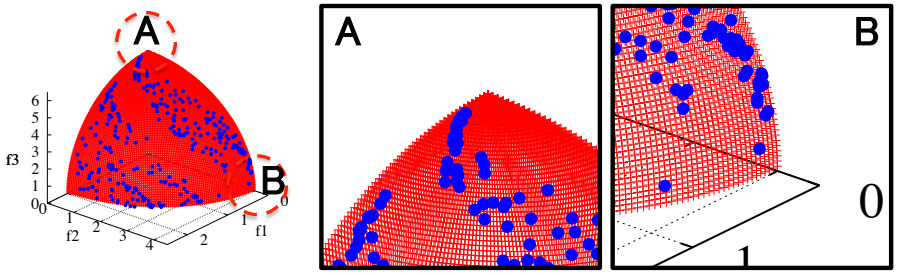


(f) 2-phase NSGA-II with EP (2nd phase)

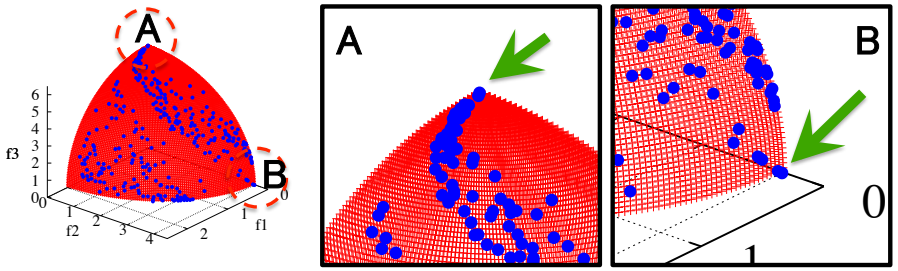
Fig. 3.8 Plots of the nondominated solutions with the lowest IGD-metric in 30 runs of NSGA-II and NSGA-II with the proposed search scheme (with/without external population), for tri-objective WFG7.

This could be more effective in many objective problems, because the search ability of MOEA was degraded by the increase in the number of objectives [48]. This finding is illustrated in the

behavior of the tri-objective WFG7, WFG8, and WFG9 solutions. To improve the performance of the second search phase, we must apply a stronger single-objective algorithm to the SOEA population. For the WFG5 problem, the IGD values are lower than in other problems regardless of algorithm used, and our proposed scheme performs worse than the original algorithm. This behavior occurs because WFG5 is an easy problem and should be solved by the original NSGA-II, rather than by our proposed search scheme. In fact, this problem is deceptive but not complex (see Table 3.1).



(a) NSGA-II



(b) 2-phase NSGA-II with EP

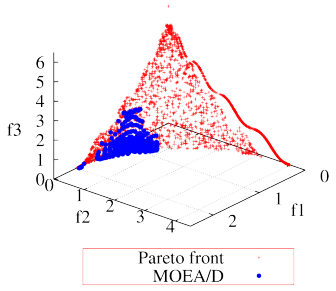
Fig. 3.9 Plots of the nondominated solutions with the lowest IGD-metric in 30 runs of NSGA-II and NSGA-II with the proposed search scheme (with external population), for tri-objective WFG8.

Regardless of the success of our proposed scheme, the external population improved the coverage. Furthermore, in the absence of an external population, the coverage may be poorer than in the original NSGA-II, because the SOEA subpopulation in the second phase of our proposed scheme does not explore the Pareto solutions, unlike the single MOEA in the original algorithm. Thus, we recommend using an external population with our proposed scheme.

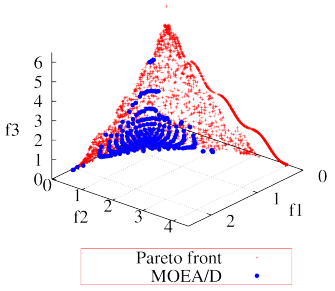
MOEA/D-based implementation

We now discuss the search behavior of the MOEA/D-based implementations of our proposed scheme.

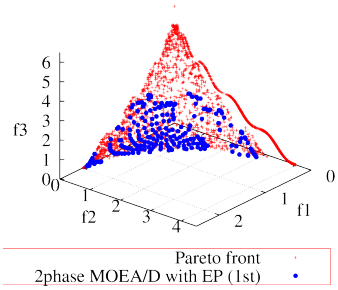
The distributions of the final solutions obtained in the run scoring the lowest IGD are plotted in figures 3.10 and 3.11. Results are shown for each algorithm (i.e., original MOEA/D, hybrid algorithm with no external population, and the hybrid algorithm with an external population). In these figures, the original MOEA/D, MOEA/D implemented in our proposed search scheme, and MOEA/D implemented in our proposed scheme with an external population are labeled “ MOEA/D, ” “ 2-phase MOEA/D, ” and ”2-phase MOEA/D with EP. ”



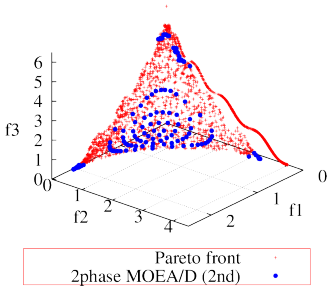
(a) MOEA/D (stopped at 223-th gen.)



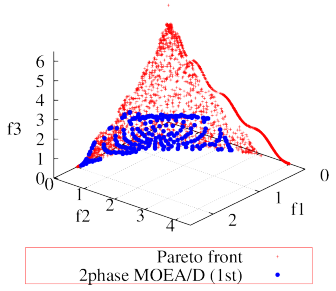
(b) MOEA/D



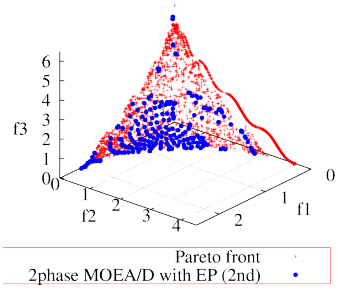
(c) 2-phase MOEA/D with EP (1st phase)



(d) 2-phase MOEA/D without EP (2nd phase)

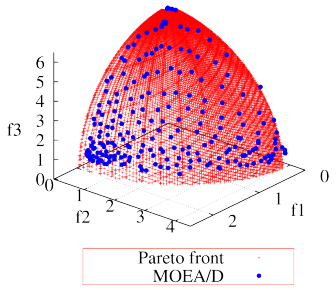


(e) 2-phase MOEA/D without EP (1st phase)

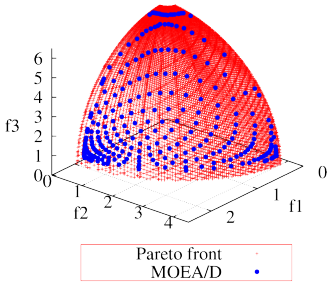


(f) 2-phase MOEA/D with EP (2nd phase)

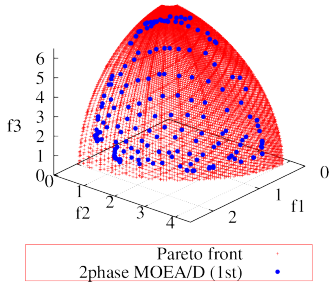
Fig. 3.10 Plots of the nondominated solutions with the lowest IGD-metric in 30 runs of MOEA/D and MOEA/D in the proposed search scheme (with/without external population), for the tri-objective WFG1.



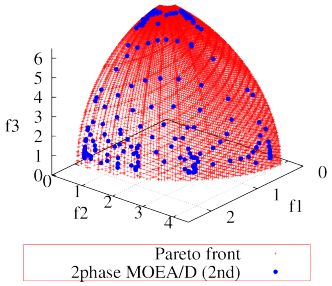
(a) MOEA/D (stopped at 50-th gen.)



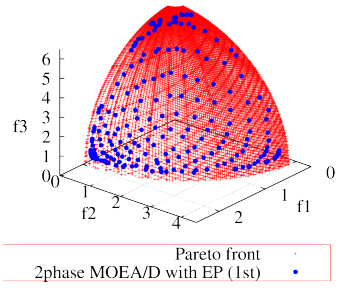
(b) MOEA/D



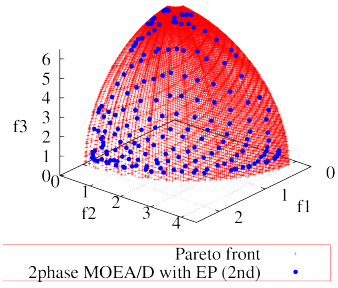
(c) 2-phase MOEA/D without EP (1st phase)



(d) 2-phase MOEA/D without EP (2nd phase)



(e) 2-phase MOEA/D with EP (1st phase)



(f) 2-phase MOEA/D with EP (2nd phase)

Fig. 3.11 Plots of the nondominated solutions with the lowest IGD-metric in 30 runs of MOEA/D and MOEA/D in the proposed search scheme (with/without external population), for the tri-objective WFG9.

As demonstrated in Figure 3.10, our proposed search scheme improves the IGD values from those of the original algorithm in the tri-objective WFG1 problem. In the tri-objective WFG9

problem, where the IGD value is degraded from that of the original MOEA/D, our proposed scheme fails (Figure 3.11). Figure 3.10 reveals that the population of our proposed search scheme converges near the Pareto solutions during the first phase, and spreads more widely during the second phase than in the original algorithm. In contract to the NSGA-II case, the first phase spreads the population more widely than in the original MOEA/D. This occurs because the MOEA/D framework is unmodified in the first phase, whereas the mating scheme of NSGA-II is modified in that phase. In the WFG1 problem, the performance of our proposed scheme is improved over the original MOEA/D because the DC-scheme enables a wider population spread.

In Figure 3.11, although our proposed search method yields a poorer IGD than the original algorithm, we observe no significant difference between the population spreads among the three algorithms. However, the distribution of the nondominated solutions was more uniform in the original MOEA/D. This behavior is expected because the original MOEA/D decomposes the multi-objective problem into uniformly distributed subproblems, whereas population division by our proposed scheme at the beginning of the second phase is nonuniform. To overcome this problem, a more efficient updating scheme could be applied to the external population. Alternatively, uniformity could be considered in the SOEA implementation. Similar to the NSGA-II case, the easy WFG5 problem should be solved by the original MOEA/D rather than by our proposed scheme.

Comparison of the computational cost

CPU times of the proposed search scheme (with external population) and the original algorithm on NSGA-II are shown in Table 3.9 and 3.10.

Table 3.9 CPU time [sec] (mean value of 30 runs) of the original NSGA-II and NSGA-II with the proposed search scheme, in the bi-objective case. ('NA' indicates overall evaluations were consumed in the first phase, so that the second phase search was not performed.)

Problem	Original	With Proposed Search Scheme											
		$\varepsilon = 0.5$			$\varepsilon = 0.1$			$\varepsilon = 0.05$			$\varepsilon = 0.01$		
		1st	2nd	Total	1st	2nd	Total	1st	2nd	Total	1st	2nd	Total
WFG1	10.50	1.63	8.53	10.16	1.68	8.51	10.19	1.74	8.48	10.22	1.83	8.43	10.26
WFG2	7.26	1.10	7.24	8.34	1.75	6.85	8.60	2.71	6.24	8.95	8.41	3.10	11.51
WFG3	10.45	0.82	8.36	9.19	1.37	8.08	9.45	1.89	7.79	9.68	10.64	4.07	14.71
WFG4	10.82	0.81	9.06	9.87	1.40	8.72	10.12	2.38	8.12	10.51	10.17	4.16	14.33
WFG5	10.93	0.85	9.19	10.04	1.30	8.92	10.21	1.54	8.78	10.32	7.07	6.07	13.14
WFG6	10.99	0.81	8.82	9.63	1.44	8.46	9.91	2.57	7.82	10.39	13.34	3.06	16.40
WFG7	14.97	1.14	14.49	15.64	1.47	14.27	15.74	1.96	13.97	15.93	14.86	5.67	20.53
WFG8	6.73	3.86	5.03	8.89	9.34	NA	9.34	9.34	NA	9.34	9.34	NA	9.34
WFG9	16.26	1.13	16.78	17.91	2.16	15.94	18.10	3.44	14.90	18.34	16.46	7.16	23.61

Table 3.10 CPU time [sec] (mean value of 30 runs) of the original NSGA-II and NSGA-II with the proposed search scheme, in the tri-objective case. ('NA' indicates overall evaluations were consumed in the first phase, so that the second phase search was not performed.)

Problem	Original	With Proposed Search Scheme											
		$\varepsilon = 0.5$			$\varepsilon = 0.1$			$\varepsilon = 0.05$			$\varepsilon = 0.01$		
		1st	2nd	Total	1st	2nd	Total	1st	2nd	Total	1st	2nd	Total
WFG1	27.14	3.42	21.55	24.97	3.74	21.39	25.13	3.78	21.35	25.13	4.44	21.07	25.51
WFG2	20.75	1.24	18.59	19.83	2.42	18.04	20.47	3.99	17.23	21.22	32.89	6.69	39.58
WFG3	25.84	1.53	18.11	19.64	2.42	17.71	20.13	3.41	17.25	20.66	24.24	8.86	33.10
WFG4	31.02	0.93	24.11	25.04	1.94	23.31	25.25	3.96	22.04	25.99	41.55	2.92	44.47
WFG5	32.36	0.97	23.90	24.87	2.07	23.33	25.41	2.40	23.17	25.57	12.20	19.32	31.52
WFG6	30.61	0.92	21.95	22.87	2.04	21.47	23.50	2.83	21.13	23.95	29.03	10.68	39.71
WFG7	36.91	1.14	35.84	36.98	2.37	34.83	37.20	2.61	34.68	37.29	47.73	17.07	64.80
WFG8	20.30	0.82	21.78	22.60	1.88	21.20	23.07	6.20	18.49	24.69	38.28	NA	38.28
WFG9	39.85	1.22	41.85	43.07	2.60	40.65	43.25	2.60	40.64	43.24	39.36	29.79	69.15

It can be seen that the proposed search scheme based on NSGA-II requires more CPU time than the original one when the threshold value ϵ get lower. It indicates that the CPU time of the proposed search scheme depends on the proportion of the first phase execution on the overall search. The first phase algorithm RM-NSGA-II needs extra CPU time to calculate the distance and use it in the mating operation to the reference point compared to the original one. Moreover, since the computational complexity of DGA is lower than NSGA-II because it does not calculate the crowding distance, the second phase search is preferred to be increased in terms of the time complexity.

On the other hand, in the case of MOEA/D (the CPU times are summarized in Table 3.11 and 3.12), the time complexity of the proposed search scheme is determined purely by the second phase search, DC-scheme, because the original algorithm is used in the first phase search (except for checking the convergence in the first phase). As MOEA/D solves single optimization sub-problems simultaneously at each iteration, there are no significant difference between DGA and MOEA/D. In this case, DC-scheme, which applies the individual migration between the SOEA populations and MOEA population, is computationally heavier than the original algorithm, so that the proposed search scheme is computationally more expensive than the original algorithm. But we expect that this difference in time complexity is not so serious if the proposed method is applied to the problem with computationally expensive objective functions, which we often encounter in real-world applications.

As the summary, the CPU time improvement of the proposed algorithm against the original one depends on the time complexity of the base algorithm used in each search phase. Especially, if the MOEA population has lower computational complexity, the CPU time of our search scheme

Table 3.11 CPU time [sec] (mean value of 30 runs) of the original MOEA/D and MOEA/D with the proposed search scheme, in the bi-objective case. ('NA' indicates overall evaluations were consumed in the first phase, so that the second phase search was not performed.)

Problem	Original	With Proposed Search Scheme											
		$\varepsilon = 0.5$			$\varepsilon = 0.1$			$\varepsilon = 0.05$			$\varepsilon = 0.01$		
		1st	2nd	Total	1st	2nd	Total	1st	2nd	Total	1st	2nd	Total
WFG1	4.15	1.15	4.04	5.19	1.28	3.83	5.11	1.31	3.79	5.09	1.33	3.75	5.08
WFG2	3.13	0.28	4.12	4.40	0.44	3.83	4.27	0.53	3.67	4.20	0.58	3.58	4.16
WFG3	3.43	0.24	4.80	5.04	0.28	4.74	5.02	0.29	4.73	5.02	0.31	4.70	5.01
WFG4	4.08	0.28	5.76	6.04	0.32	5.71	6.03	0.35	5.66	6.01	0.37	5.63	6.00
WFG5	3.69	0.26	5.15	5.41	0.26	5.15	5.41	0.26	5.16	5.41	0.26	5.15	5.41
WFG6	3.90	0.27	5.44	5.71	0.29	5.41	5.70	0.30	5.41	5.70	0.32	5.38	5.70
WFG7	7.89	0.89	10.63	11.52	1.09	10.33	11.42	1.12	10.25	11.37	1.15	10.21	11.36
WFG8	4.06	0.44	5.58	6.02	0.63	5.34	5.97	0.66	5.28	5.94	0.73	5.19	5.92
WFG9	9.49	0.63	13.32	13.95	0.66	13.28	13.94	0.68	13.21	13.90	0.75	13.10	13.85

Table 3.12 CPU time [sec] (mean value of 30 runs) of the original MOEA/D and MOEA/D with the proposed search scheme, in the tri-objective case. ('NA' indicates overall evaluations were consumed in the first phase, so that the second phase search was not performed.)

Problem	Original	With Proposed Search Scheme											
		$\varepsilon = 0.5$			$\varepsilon = 0.1$			$\varepsilon = 0.05$			$\varepsilon = 0.01$		
		1st	2nd	Total	1st	2nd	Total	1st	2nd	Total	1st	2nd	Total
WFG1	7.47	1.32	11.31	12.63	1.52	11.10	12.62	1.53	11.09	12.63	1.60	10.99	12.59
WFG2	6.69	2.63	6.92	9.55	4.01	4.63	8.64	4.13	4.49	8.62	4.23	4.60	8.83
WFG3	6.57	0.21	10.96	11.18	0.37	10.88	11.25	0.37	10.89	11.26	0.37	10.89	11.26
WFG4	7.80	0.36	13.78	14.15	0.43	13.66	14.09	0.43	13.67	14.10	0.43	13.67	14.10
WFG5	7.12	0.24	12.48	12.71	0.40	12.23	12.63	0.40	12.24	12.64	0.40	12.24	12.64
WFG6	7.05	0.23	11.76	12.00	0.38	11.33	11.71	0.38	11.35	11.72	0.38	11.33	11.71
WFG7	15.57	0.59	27.10	27.69	0.78	26.73	27.51	0.82	26.61	27.44	0.92	26.45	27.37
WFG8	8.97	0.46	14.40	14.87	0.50	14.34	14.84	0.52	14.31	14.82	0.54	14.28	14.82
WFG9	15.95	0.49	31.15	31.64	0.83	30.47	31.31	0.83	30.44	31.27	0.85	30.42	31.27

is worse than the original one.

Summary of the experiments

The results of the numerical experiments are summarized below:

- Our proposed search scheme performs at least as well as the original algorithm in the WFG1, WFG2, WFG3, WFG4, and WFG6 problems.
- The original algorithm outperforms our proposed scheme in the WFG5, WFG7, WFG8, and WFG9 problems.
- Because WFG5 is an easy problem, it should be solved by the original algorithm rather than by our proposed scheme.

- To correct the poor performance of our proposed scheme in certain problems, we should improve the search performance in the SOEA implementation. Specifically, we need to consider novel and stronger algorithms for searching the SOEA population.
- An external population improves the coverage of our proposed search scheme.

In summary, our proposed search scheme demonstrates higher diversity and better proximity than conventional MOEAs.

3.4 Summary of this chapter

The popularity of MOPs stems from their wide variety of solutions presented to decision makers. Such varying solutions are more useful when they are well approximated and widely spread. For this purpose, the author proposed a novel search scheme for MOEAs that improves the diversity and proximity of Pareto solutions. The proposed scheme comprises two phases, each using a different type of EA. The first phase enables rapid convergence to the Pareto solutions, and the second phase spreads the search population. In the first phase, rapid proximity is achieved by a user-specified reference point; in the second phase, solutions are spread by a DC-scheme proposed by Okuda et al.[23].

The proposed scheme was applied to a dominance-based algorithm (NSGA-II), and a decomposition-based algorithm (MOEA/D). To verify the effectiveness of our proposed search scheme, both implementations were tested on the WFG test suite. The results showed that for WFG1, WFG2, WFG3, WFG4, and WFG6, the proposed search scheme was superior to the original algorithm. Conversely, the original algorithm outperformed our scheme in WFG5, WFG7, WFG8, and WFG9. In cases of inferior performance by our proposed scheme, it was observed that the SOEA population for each objective did not search the extreme point during the second phase. Hence, it could be concluded that the overall performance of our scheme crucially depended on the algorithm applied to the SOEA population. Moreover, it was verified that an external population improved the coverage of our proposed search scheme. Different SOEA implementations should be investigated in additional studies.

A major contribution of our research is the development of a versatile search scheme, which is applicable to any type of MOEA. Finally, it has been showed that the performance of conventional MOEA could be easily improved by hybridization with an appropriate SOEA.

Chapter 4

Design Mode Analysis

4.1 Introduction: What is ”design mode” ?

Decision variables are defined as a set of parameters that determine the solutions to optimization problems. In product design problems, for example, they frequently represent size, weight, and shape of the product. Objective functions indicate the goal of the product design, such as performance and cost of the target product. Decision variables and objective functions form the decision space and the objective space, respectively, as shown in Figure 4.1. 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 nondominated solutions, especially in the objective space [19, 49]. How-

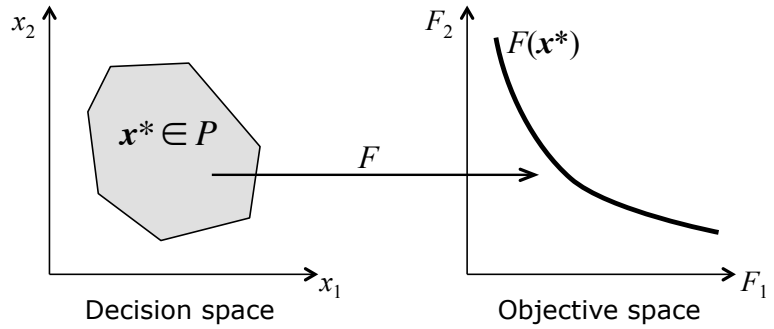


Fig. 4.1 Representation of decision space and objective space

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

In this study, we incorporate the“ design mode ”concept as an essential perspective in decision

space analysis. This concept is derived from principal component analysis (PCA) in statistics, proper orthogonal decomposition (POD). These methods are mathematically same (the word "PCA" is used in this work) and 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 maximal variance among the variables. Oyama et al. [8, 9] applied PCA in their analysis of airfoil shape design, and reported its effectiveness. They focused on designing the shape of the product, seeking the principal airfoil shapes that form the Pareto solutions.

In this study, we generalize and extend the concept to deal with all types of decision parameters in the product design. Directions indicated by principal component vectors also tell us how to change the decision variables to construct the designs in the given data set. Besides, contribution ratio of each decision variable to the principal component vectors shows how important that variable is in creating the designs. We assume that the principal component vector gives us important information in the engineering design, and define it as "design mode." The analytical framework based on the design mode is also proposed.

Let $X^{(i)} = (x_1, \ldots, x_m) \in \mathbb{R}^{\mathrm{>}}$ to be a decision variable of the i-th Pareto solution, and $X = (X^{(1)}, \ldots, X^{(N)})$ be a Pareto solution set of size N. In PCA, the following optimization problem is solved, and the vector $\boldsymbol{w}$ that maximizes the variation in the decision variables is selected:

$$\max_{\|\boldsymbol{w}\|=1} \mathrm{Var}[\boldsymbol{w}^T X] = \max_{\|\boldsymbol{w}\|=1} \boldsymbol{w}^T \mathrm{Var}(X) \boldsymbol{w} \tag{4.1}$$

where the matrix $\mathrm{Var}(X) = C$ is the covariance matrix of the dataset X. Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m$ be eigenvalues of C, and v_j be their corresponding eigenvectors. The k-th principal component $Y^{(i,k)}$ of the data $X^{(i)}$ is then represented by

$$Y^{(i,k)} = \boldsymbol{v}_k X^{(i)} \tag{4.2}$$

The original dataset can be decompoed 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:

$$X^{(i)} \simeq \boldsymbol{u} + \sum_{k=1}^{p} \alpha_k \boldsymbol{v}_k \tag{4.3}$$

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.

$$P = \frac{\sum_{i=1}^{p} \lambda_i}{\sum_{i=1}^{m} \lambda_i} \tag{4.4}$$

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 axis of the decision variable x_j in the original decision space and the new axis indicated by the k-th design mode, we calculate the component loading:

$$r_{kj} = \sqrt{\frac{\lambda_k}{s_j}} v_{kj} \tag{4.5}$$

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

This section has revealed some important facts of design mode analysis.

- Applying PCA to a Pareto set enables the extraction of dominant designs and decision variables.
- Pareto solutions can be approximated by the mean vector of the dataset and a linear combination of a certain number of eigenvectors.
- 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.

4.2 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 procedures of the design mode analysis are (1) *generate the dataset*, (2) *cluster the dataset*, (3) *perform PCA*, and (4-1) *perform correlation analysis* or (4-2) *construct new design*. The proposed framework is illustrated in Figure 4.2.

The differences of our proposed approach from the conventional method by Oyama et. al are the following points:

1. Data clustering process is incorporated into the analysis.

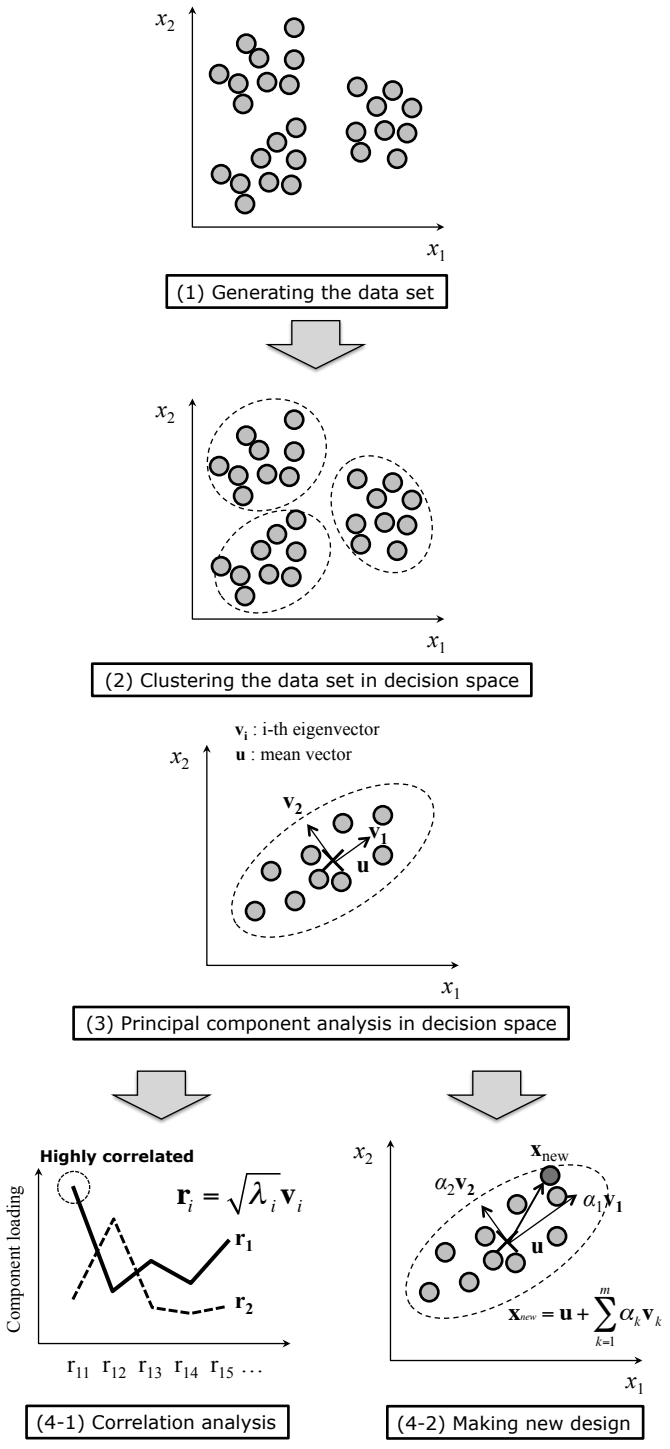


Fig. 4.2 Framework of design mode analysis

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.

4.2.1 Generating the dataset

As a first step in our proposed analysis, we consider how to generate the dataset (design examples). One of the easiest ways to generate them is to use statistical sampling methods such as Latin hypercube sampling. Representative designs can also be supplied by an expert engineer. How the dataset should be generated depends 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. Otherwise, if the aim is to elucidate the design methodology of an expert, many design examples constructed by the experts must be collected. In this study, we investigate the characteristics of the decision space around the Pareto solution set, and then obtain the solutions by EMO. Note that our proposed framework does not determine the data generation method. The multiobjective optimization is nothing more than one of the best tools to generate characteristic designs.

4.2.2 Clustering the dataset

By incorporating data clustering as data preprocessing, we can obtain reliably-distinguished design modes. Data clustering for the Pareto solution set can be applied to either 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 in the conventional method proposed by Oyama et al. [8, 9]. 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.

4.2.3 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.

4.2.4 Correlation analysis

Having obtained the design modes in 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 in creating new designs in each cluster. Decision variables that make low contributions to the design mode can be eliminated from the variables, and instead set as constants. The component loadings characterize the design modes and give us their features. This process was also not considered in the conventional method, and it will give us useful information about the target problem.

4.2.5 Constructing new design

Our proposed method, called design mode analysis, is employed as an analytical tool as well as a design support tool. Since most designs in each cluster can be approximated in the decision space formed by design modes, we can easily generate new designs with the same features in this space, using Equation (4.3). Note that designs sharing characteristics with a specific design are not easily created by random sampling in the original decision space, especially if that decision space is high dimensional. We summarize the proposed framework in the pseudo code shown in Algorithm 4.1.

Algorithm 4.1 Design mode analysis

1: Generate a design data set $X = (X^{(1)}, \ldots, X^{(N)})$ of size N.
2: Divide the data set into H clusters by using data clustering.
3: **for** $i = 1$ to H
4: Extract the design mode $\boldsymbol{v}_k = (v_{k1}, v_{k2}, \ldots, v_{km})$ by applying the PCA ($k = 1, \ldots, N$) on i-th
5: Calculate and study the component loading $\boldsymbol{r}_k = (r_{k1}, r_{k2}, \ldots, r_{km})$ for each $\boldsymbol{v}_k$ using $r_{kj} = \sqrt{\frac{\lambda_k}{s_j}} v_{kj}$.
6: Choose a base design $\boldsymbol{b}$ 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 $X' = \boldsymbol{b} + \sum_{k=1}^{p} \alpha_k \boldsymbol{v}_k$. Coefficient α_k is an arbitrary constant.
9: **end for**

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

4.3 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.

- Each design mode characterizes the dataset.
- Data clustering process effectively distinguishes the design modes.
- Any design in each cluster is approximated by the mean design and a linear combination of the design modes in the cluster.
- Our proposed method can be applied to binary-valued problems.

4.3.1 Experimental setup

The target design problem, MOKP, is a multiobjective extension of the classic 0/1 knapsack problem (KP), a kind of NP-complete combinatorial problem. When we try to pack the items, which have own values and weights, into the knapsacks with the capacity constraints, it is important to choose the items so as to maximize the total profit of the items put into knapsacks, because we can not pick all the items. This is popularly known as the classic KP in combinatorial optimization, and often appears in real-world decision-making problems in various fields, such as production scheduling, portfolio management, and so on. Figure 4.3 illustrates the classic KP. In KP, the 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 a objective function. The objective is to find a subset of the items with total weight not exceeding the knapsack capacity, while maximizing the total profit. MOKP is a extended form of KP whose number of the knapsack is simply increased.

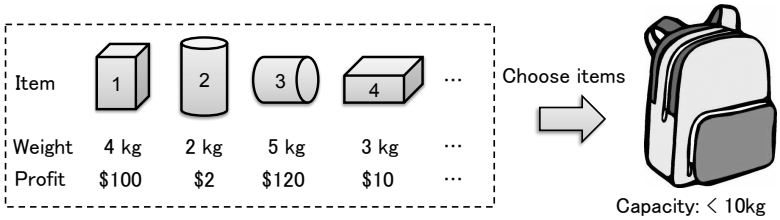


Fig. 4.3 Knapsack problem

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

$$\begin{aligned} \text{maximize} \qquad & f_i(\boldsymbol{x}) = \sum_{j=1}^{N} x_j p_{(i,j)}, \quad x_j \in \{0,1\} \\ \text{subject to} \qquad & g_i(\boldsymbol{x}) = \sum_{j=1}^{N} x_j w_{(i,j)} \leq W_i, \quad 1 \leq i \leq k \end{aligned} \tag{4.6}$$

where $p_{(i,j)}$ indicates the profit of 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 [50], a collection of datasets available for testing various multiobjective optimization problems. The study constitutes a bi-objective (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) [49], 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. Setting of the crossover and mutation rate used here follows the practice in [19, 49]. 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 out of focus of our study to adjust and study the parameter setting. The multiobjective optimization is nothing more than the tool to generate the data set to be analyzed.

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

4.3.2 Results and discussion

Figure 4.4 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 executed on the decision variables of the nondominated solutions set. The results of running PCA on MOKP nondominated solutions set are summarized in Table 4.1. The original dataset can be explained if the cumulative proportion of explained variance is ≥ 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 4.5. The component loadings of ele-

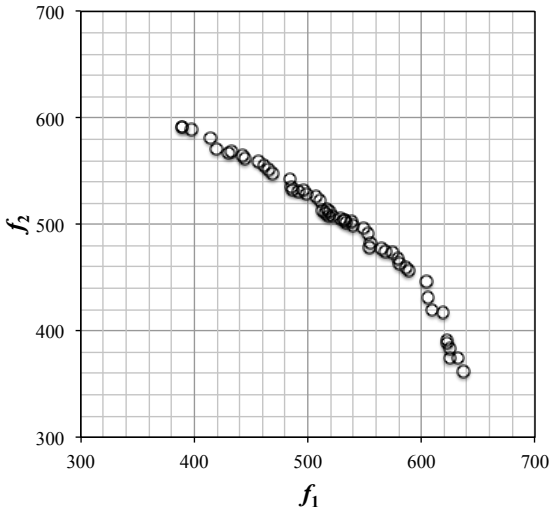


Fig. 4.4 Nondominated solutions obtained by NSGA-II

ments with zero variance are plotted as zero, because Equation (4.5) is noncalculable when $s_j = 0$. 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 strategies can pack the items into two knapsacks while maximizing the profits.

Table 4.1 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

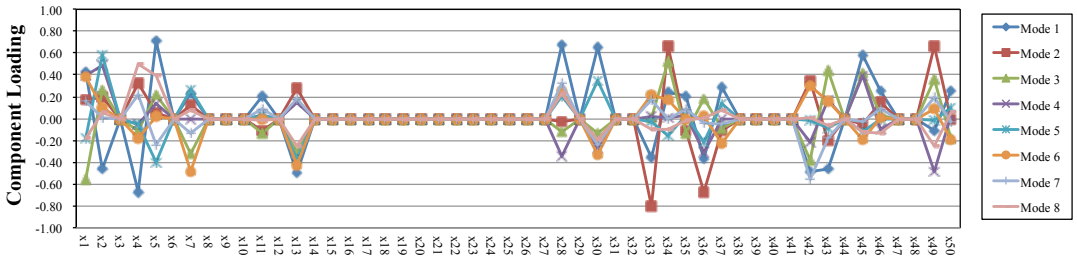


Fig. 4.5 Distribution of the component loadings (Modes 1 to 8)

Next, to evaluate the effectiveness of the data clustering, k-means clustering was performed on the dataset, yielding three clusters ($k = 3$). Figure 4.7 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 4.6. Thus, it appears that data clustering distinguishes the design modes.

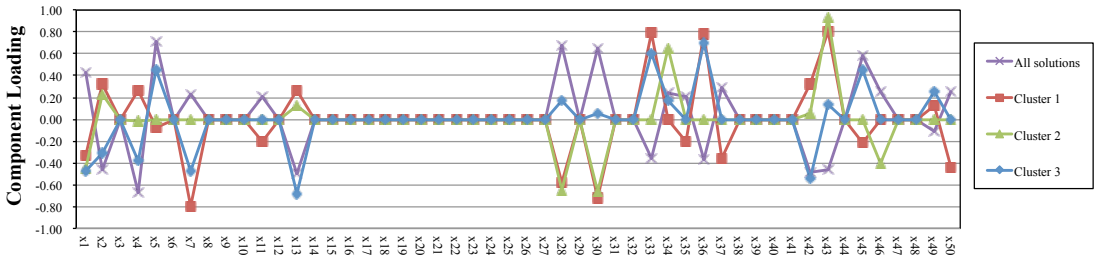


Fig. 4.6 Distribution of the component loadings of the first design mode of each cluster

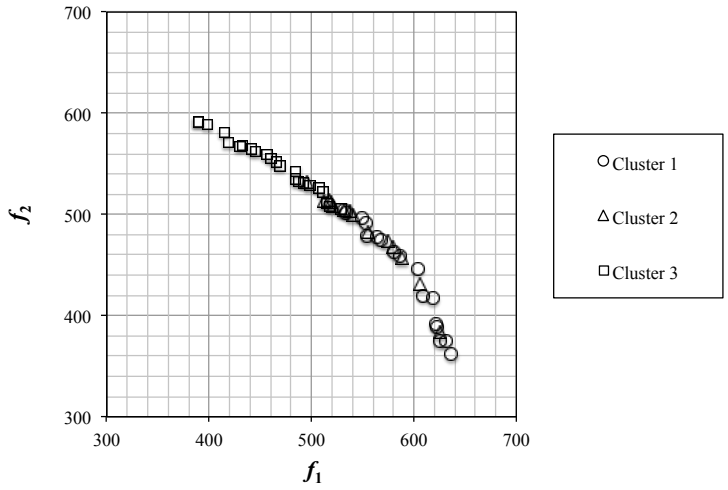


Fig. 4.7 Three clusters obtained by k-means clustering

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 (sufficiently many to achieve cumulative proportion of the variance ≥ 0.80). The procedure for approximating the solution set is shown below.

Design Approximation Method:

1. Choose a target design $X^{(i)} = (x_{i1}, \ldots, x_{im})$ from the dataset $X = (X^{(1)}, \ldots, X^{(N)})$.

2. Choose p eigenvectors so as to satisfy cumulative proportion of the variance $P \geq 0.80$.

3. Find optimum coefficients α_j $(j = 1 \ldots p)$ of the approximated design calculated by Equation (4.3) so as to minimize the sum of squared error (SSE) between the target and approximated design in decision space:

$$\sum_{k=1}^{m}(x'_{ik} - x_{ik})^2 \tag{4.7}$$

where x_{ik} is k-th decision variable of i-th design in the dataset, and x'_{ik} is approximated design of x_{ik}. Note that if there are the constraints in decision variables of the original dataset, they should be add to Equation (4.7).

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. However, the solution obtained by the abovementioned 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 round off to the closest whole number. If the integer lies outside of $[0, 1]$, it is assumed as 0 or 1. If it is smaller than 0, it is assumed as 0. In the same way, if it is larger than 1, it is assumed as 1.

Results of the design approximation are summarized in Table 4.2, where g is the constraint value in Equation (4.6). SSE is the approximated error in the decision space defined by Equation (4.7). 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 trialed designs approximated in two different ways. First, PCA was applied to all solutions, and the design was approximated by the mean vector and the design modes of all solutions. Second, PCA and design approximation were executed on the dataset of Cluster 1. The results of both trials are listed for comparison in Table 4.2. Moreover, the approximated designs in the objective space are plotted in Fig. 4.8.

Table 4.2 Summary of design approximation using design modes

	f_1	f_2	g	SSE	D_h	D_f	p
Target design	**637**	**362**	**187**	-	-	-	-
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)	**637**	**362**	**187**	**3.40E−01**	**0**	**0.00**	**7**

Table 4.2 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,

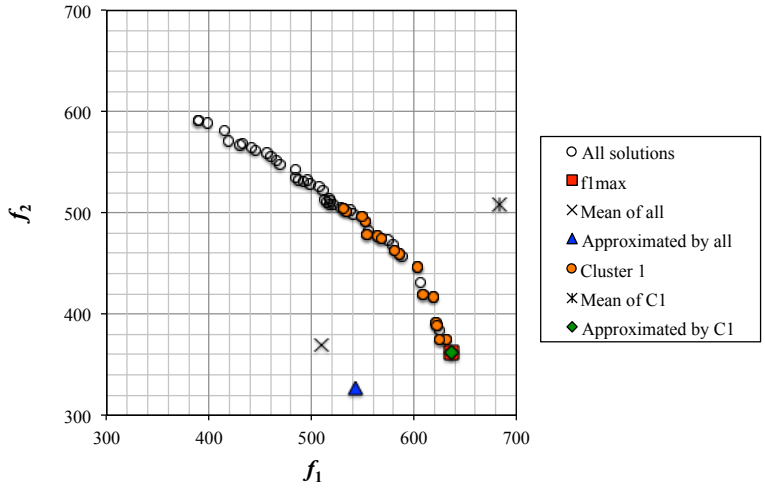


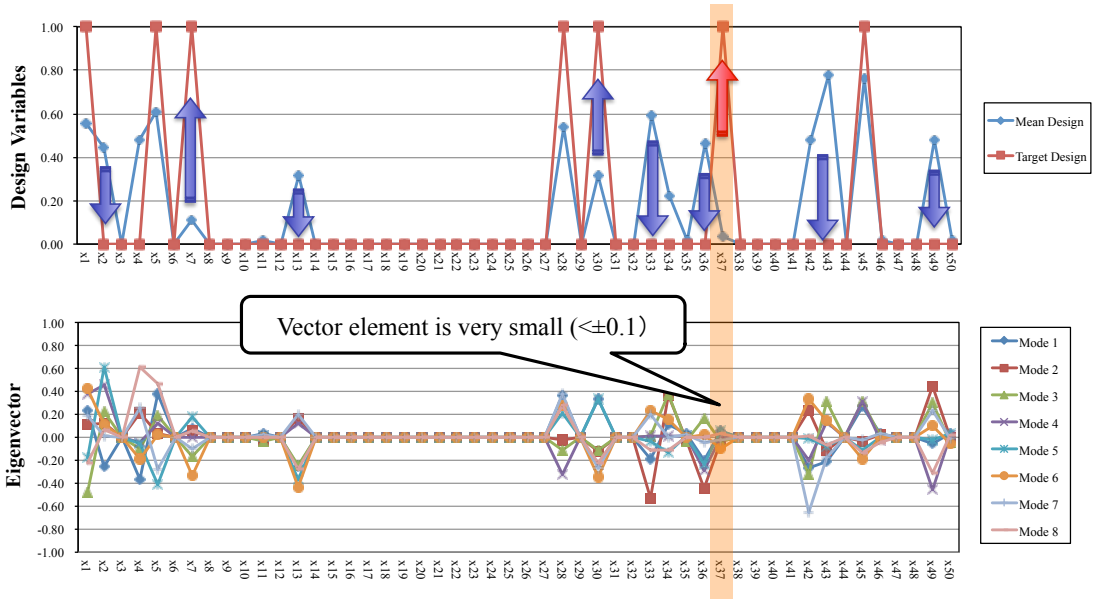
Fig. 4.8 Plots of approximated designs in objective space

the SSE was an order of magnitude greater. These results are also evident in the plots of Figure 4.8.

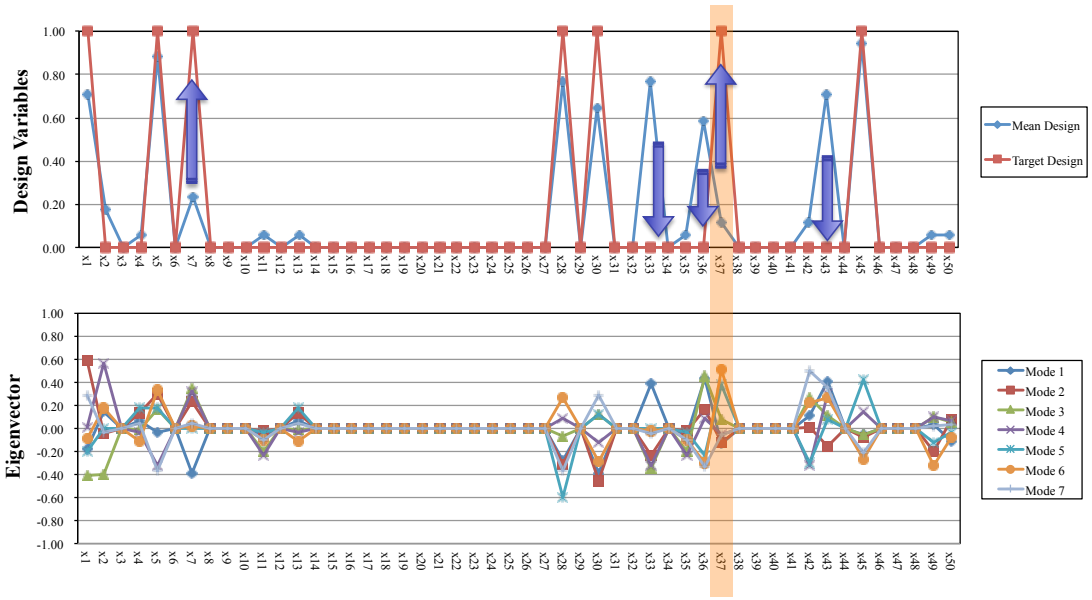
To ensure that data clustering effectively distinguishes the design modes, we analyzed the performance of the approximated design. In this analysis, we adjusted the size of each eigenvector before adding it to the mean vector. The upper charts in Figure 4.9 show the distributions of the elements of the mean and target designs, 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 whose values differ from the target values should be altered when the eigenvectors are added. The directions of the altered elements are indicated by arrows on the charts. For instance, observe the 37th variable emphasized by the hatched pattern in Figure 4.9(a) and 4.9(b). In the design approximation without data clustering (Figure 4.9(a)), the magnitude of the 37th variable is very much smaller (± 0.1) than that obtained after data clustering (Figure 4.9(b).). In this case, a larger coefficient α_j $(j = 1 \ldots p)$ 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 the clustered data, the lacking element of the mean design compared to the target design is compensated by the corresponding element of the eigenvectors. These results show that data clustering is effective for extracting the design modes precisely.

This case study also highlights the importance of the granularity of analysis in our proposed design mode analysis. In the absence of clustering, we assume that some decision variables



(a) Without clustering (PCA is performed on the entire solutions)



(b) Cluster 1 (PCA is performed on each cluster)

Fig. 4.9 Distribution of eigenvectors in design approximation of f_1-max solution

contribute negligibly to the design. Thus, a granularity exists in the design mode extraction. Data clustering increases the granularity of the design modes. Watanabe et al. [51]. proposed an interactive granularity control method, which is applicable to our proposed design mode analysis.

4.4 Granularity in the design mode analysis

In the above case study, we introduced the concept of “ granularity ” in the design mode analysis. The granularity of the design mode extraction required by DM depends on the situation. For example, if the 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 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. Focusing on granularity, this subsection proposes a more general framework for our design mode analysis.

Following Watanabe et al. [51], we adopted a hierarchical approach. The proposed framework iterates binary clustering and PCA, and approximates a design for each cluster. Differently 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 4.10.

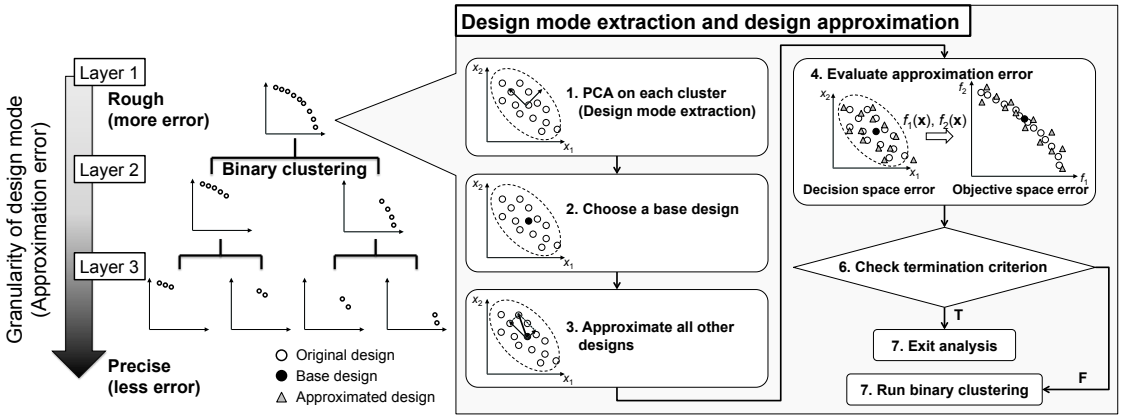


Fig. 4.10 Proposed framework of design mode analysis focusing on granularity

The analysis began with a single cluster comprising the given design dataset. For each cluster, design modes were extracted by PCA, and an approximate design was constructed. Thus, we obtain the design modes and approximation error in each cluster. Although the approximation was executed on the decision space, the objective function values of the approximated decision variables were evaluated, and the approximation error in the objective space was also calculated.

The design approximation process is discussed in 4.3.2. If the design modes are successfully derived, any design can be approximated by a linear combination of the design modes, as shown in Equation (4.3). 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 in its own cluster. The easiest way to choose the representative design is to calculate

the mean vector and set it as a 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 new two clusters, and progressing to the next granularity layer. Note that PCA and clustering are not performed on clusters of a single data size. In this case, the cluster is inherited by the next layer.

The abovementioned procedures yield design modes at any level of granularity.

4.5 Case study—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, one of the most useful real-world optimization problems for testing the performance of optimization algorithms [52], was first proposed by Oyama et al. [53]. The executable software for objective function evaluation is available from the website [54].

4.5.1 Problem definition

Hybrid rocket engine is the rocket engine in which propellant is stored in two different kinds of phases. It is becoming increasingly popular with its advantage of low environmental impact, flexible thrust control by throttling, and reduced chemical explosion hazard. In the hybrid rocket engine, the thrust and the engine design are strongly correlated because it obtains the thrust by combustion in the boundary layer diffusion flame. Thus, it is important and difficult to design the solid fuel geometry and the oxidizer supply system.

The investigated hybrid rocket comprises four parts: a payload, an oxidizer tank, a thrust chamber, and a nozzle. The thrust of the hybrid rocket is developed by combustion in a turbulent boundary layer in the thrust chamber. The thrust is also affected by the oxidizer and mass/fuel ratio. 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 in order to maximize the altitude gained while minimizing the gross weight. A schematic of the hybrid rocket is shown in Figure 4.11.

Here, the six-dimensional decision space comprises the initial mass flow of the oxidizer $\dot{m}_{\mathrm{oxi}}(0)$ [kg/s], fuel length L_{fuel} [m], initial port radius $r_{\mathrm{port}}(0)$ [m], combustion time t_{burn} [s], initial

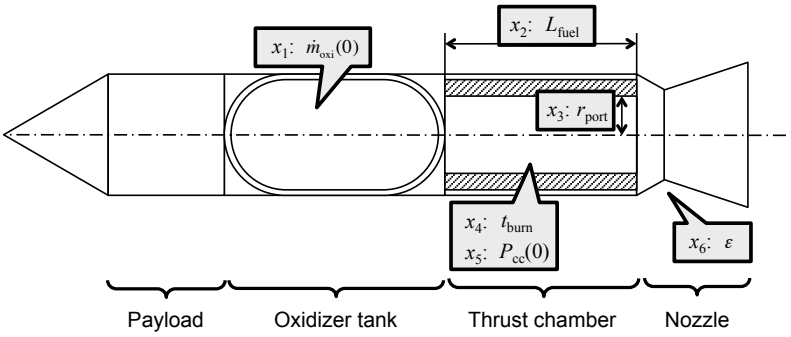


Fig. 4.11 Schematic of the hybrid rocket

pressure in the combustion chamber $P_{\mathrm{cc}}(0)$ [MPa], and aperture ratio of nozzle ϵ. The two objective functions aim to simultaneously maximize the altitude $H_{\max}$ [km] and minimize the gross vehicle weight $M_{\mathrm{tot}}(0)$ [kg]. The following equation of motion is assumed in the flight analysis:

$$a(t) = \frac{T(t) - D(t)}{M_{\mathrm{tot}}(t)} - g \tag{4.8}$$

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^2] is gravitational acceleration. The following equation relates the thrust $T(t)$ to the aperture ratio of the nozzle ϵ, and the pressure in the combustion chamber $P_{\mathrm{cc}}(0)$ [MPa]:

$$T(t) = \eta_T \left[\lambda \dot{m}_{\mathrm{prop}}(t) u_e + (P_e - P_a) A_e\right] \tag{4.9}$$

where η is the total thrust loss coefficient, and λ is the momentum loss coefficient, embodying the effect of friction (¡1) at the nozzle exit. $\dot{m}_{\mathrm{prop}}(t)$ 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 nozzle exit.

The drag $D(t)$ is decomposed into the pressure drag $D_p(t)$ and the friction drag $D_f(t)$. The parameters $\dot{m}_{\mathrm{prop}}(t)$, $D_p(t)$, and $D_f(t)$ are not described because of space limitations. For details on these parameters, the reader is referred to [53, 54].

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

$$M_{\mathrm{tot}}(t) = M_{\mathrm{en}}(t) + M_{\mathrm{pay}}(t) + M_{\mathrm{ex}}(t) \tag{4.10}$$

$$M_{\mathrm{en}}(t) = M_{\mathrm{oxi}}(t) + M_{\mathrm{fuel}}(t) + M_{\mathrm{res}}(t) + M_{\mathrm{ch}}(t) \tag{4.11}$$

$$M_{\mathrm{ex}}(t) = \frac{3}{2} M_{\mathrm{en}}(t) \tag{4.12}$$

$$M_{\mathrm{oxi}}(t) = \int_0^{t_{\mathrm{burn}}} \dot{m}_{\mathrm{oxi}}(t) dt \tag{4.13}$$

$$M_{\mathrm{fuel}}(t) = \int_0^{t_{\mathrm{burn}}} \dot{m}_{\mathrm{fuel}}(t) dt \tag{4.14}$$

$$M_{\mathrm{res}} = \rho V_{\mathrm{res}} \tag{4.15}$$

$$M_{\mathrm{ch}} = \rho V_{\mathrm{ch}} \tag{4.16}$$

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

Given Equations (4.10 – 4.16), we define the multiobjective optimization problem of the hybrid rocket engine design as

$$\begin{aligned}
\text{maximize} \quad & f_1 = H_{\max} \\
\text{minimize} \quad & f_2 = M_{\mathrm{tot}}(0) \\
\text{subject to :} \quad & \\
& 1.0 \leq \dot{m}_{\mathrm{oxi}}(0) \leq 30.0 \\
& 1.0 \leq L_{\mathrm{fuel}} \leq 10.0 \\
& 10.0 \leq r_{\mathrm{port}}(0) \leq 200.0 \\
& 15.0 \leq t_{\mathrm{burn}} \leq 35.0 \\
& 3.0 \leq P_{\mathrm{cc}}(0) \leq 4.0 \\
& 5.0 \leq \epsilon \leq 7.0
\end{aligned} \tag{4.17}$$

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

4.5.2 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 at 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 $1/$(chromosome length). The decision variable vector of a single solution is represented as $\mathbf{X} = [\dot{m}_{\mathrm{oxi}}(0), L_{\mathrm{fuel}}, r_{\mathrm{port}}(0), t_{\mathrm{burn}}, P_{\mathrm{cc}}(0), \epsilon]$ 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. Setting of the crossover and mutation rate used here follows the practice in [19, 49]. Population size, the number of generations, and other genetic parameters are empirically chosen here. The parameter study to obtain appropriate parameter setting is dismissed because it is out of focus of our study to improve the accuracy of optimization. The obtained nondominated solutions are plotted in Figure 4.12.

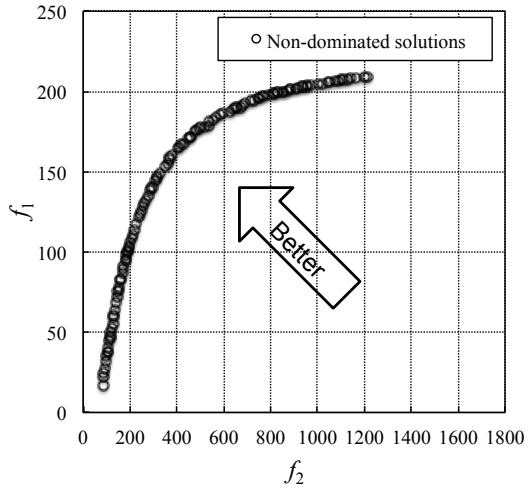


Fig. 4.12 Nondominated solution set of hybrid rocket engine design problem

Intuitively, more fuel will achieve higher altitude; however, fuel increases the weight of the rocket. The weight-altitude tradeoff is evident in Figure 4.12, 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.

4.5.3 Design mode analysis

In this subsection, we extract the decision space characteristics of the nondominated solutions of the hybrid rocket engine design, and derive an appropriate design strategy using our proposed design mode analysis (described in subsection 4.4). Prior to running the PCA, we first pre-process the decision variables of the hybrid rocket design to normalize its mean and variance. Thus, PCA is performed based on the correlations matrix. The data set should be normalized when the range

and scale of variables is different each other. Otherwise, as in the case of MOKP, the covariance matrix is used for PCA to preserve variance without normalization of the data set.

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):

$$\sum_{k=1}^{m} \frac{(x'_k - x_{ik})^2}{x_{ik}^2} \tag{4.18}$$

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}. Moreover, the PCA is also performed on the correlation matrix. The SSRE is minimized by the optimization algrithm (i.e., sequential least squares programming) [55].

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

The pseudo code of the extended framework of design mode analysis is shown in Algorithm 4.2.

Algorithm 4.2 Design mode analysis focusing on granularity

1: Generate a design data set $C_1^{(1)} = (X^{(1)}, \ldots, X^{(N)})$ of size N.
2: Scale the data set such that all decision variables have zero mean and unit variance.
3: Initialize total approximation error $E = \infty$.
4: Set a threshold η for E.
5: Initialize layer counter $i = 1$.
6: **while** $E > \eta$
7: Initialize the number of clusters in current layer $H = 2^i - 1$.
8: Initialize the counter of the clusters in new layer $k = 1$.is an arbitrary constant.
9: Initialize $E = 0$.
10: **for** $j = 1$ to H
11: Extract the design mode by applying the PCA on $C_j^{(i)}$.
12: Calculate the component loading for each design mode.
13: Choose a base design from $C_j^{(i)}$, 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**

4.5.4 Results and discussion

Figure 4.13 plots the history of the error in the decision space at each layer of the design approximation. Although the accuracy worsens in the second and third layer relative to the first layer, 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. Figure 4.14 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 4.15 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. In-

stead, 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 4.16 indicates the original and approximated designs of each cluster at each layer. In these plots, " Cx-y " denotes the y-th cluster at the x-th layer. Notably, the designs around C2-2 in the 2nd layer are almost exactly retained in passing through the 3rd to the 4th layer, possibly because these designs have distinguishable characteristics.

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 4.17. The component loading, denoted 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.

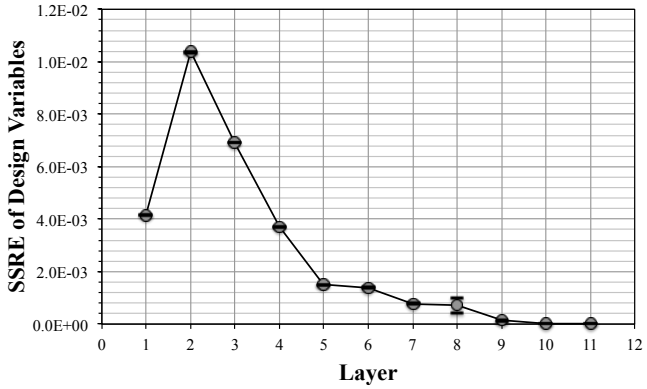


Fig. 4.13 History of the accuracy of the approximated decision variables

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 $\dot{m}_{\mathrm{oxi}}(0)$ and the fuel length L_{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. It should be noted that 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 the new design, at first we 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

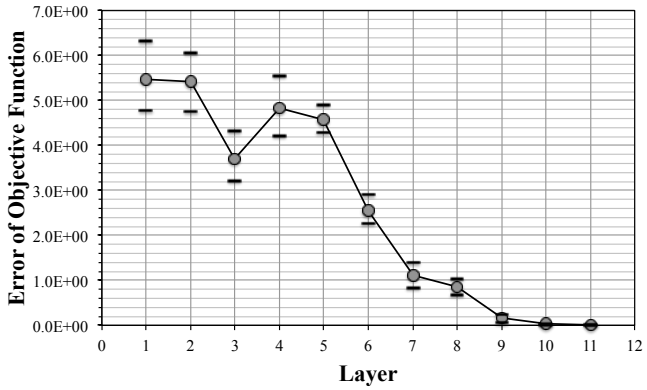


Fig. 4.14 History of the accuracy of the approximated objective function values

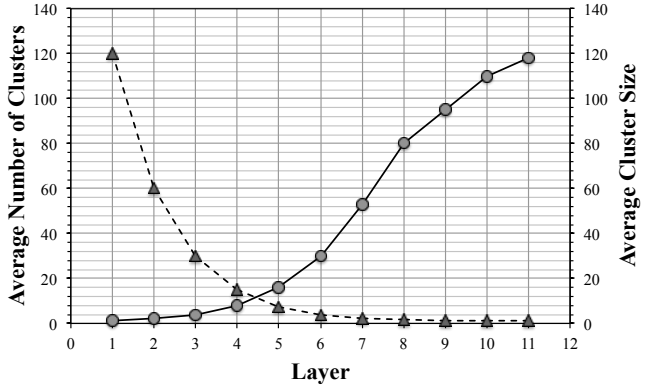


Fig. 4.15 History of the number of clusters in each layer

sign of the component loading means the direction of each decision variable on the axis indicated by 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 why the accuracy of design approximation deteriorates 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

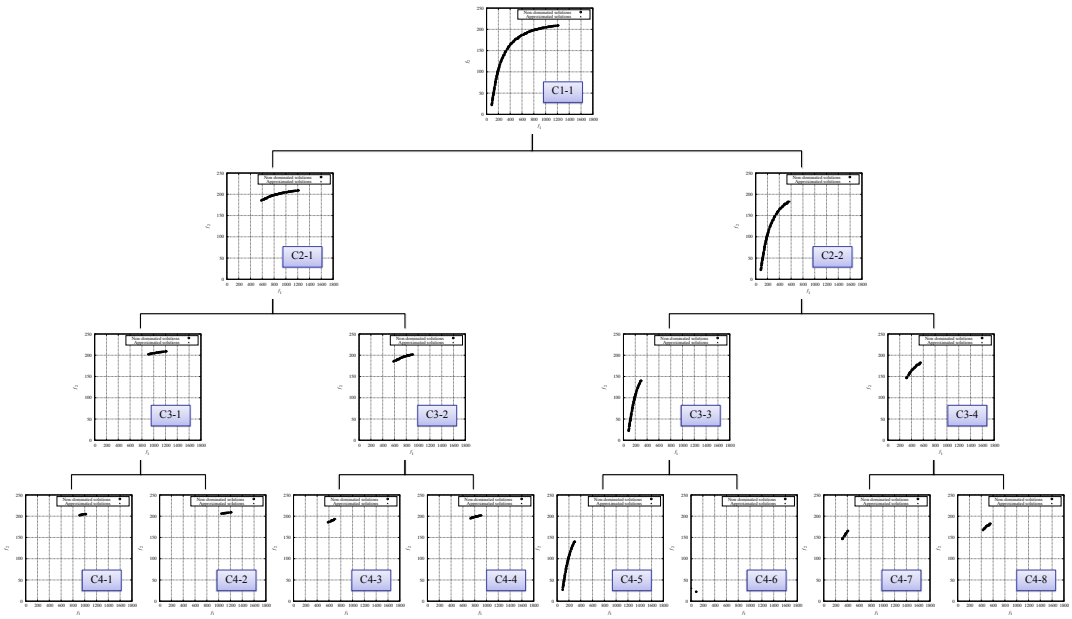
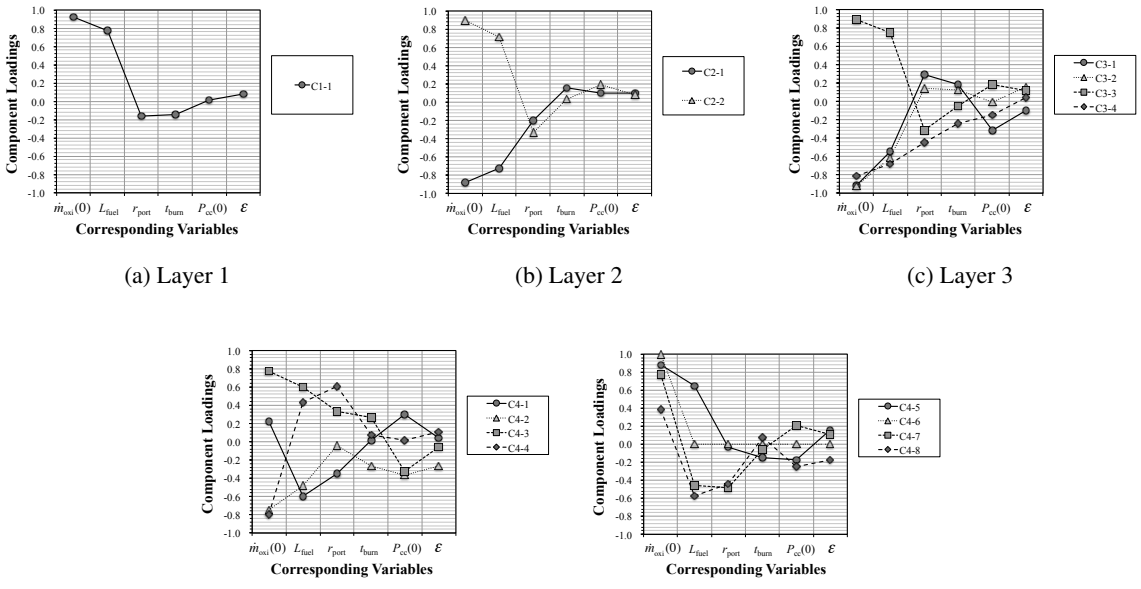


Fig. 4.16 Approximated solutions of each cluster in each layer



(a) Layer 1 (b) Layer 2 (c) Layer 3

(d) Layer 4 (from C4-1 to C4-4) (e) Layer 4 (from C4-5 to C4-8)

Fig. 4.17 Mode-1 component loadings of the each cluster at each layer

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 revealed only 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 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 become opposite in sign (x_3 is an initial port radius $r_{\mathrm{port}}(0)$). Thus, C4-4 may provide a design strategy that the decision parameter x_1 should be changed to the negative direction, and at the same time x_2 and x_3 should be modified in the positive direction. We conclude that unique design modes are obtained at the 4th layer.

From the experiments in this section, we infer that

- The deeper the layer, the better the approximation accuracy (observed in Figure 4.13 and 4.14).
- The design mode is characterized by the component loading distributions of each mode.
- In the hybrid rocket design problem, the initial mass flow of oxidizer and the fuel length dominate the Pareto
- 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 of 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 among the dataset, may improve the data classification.

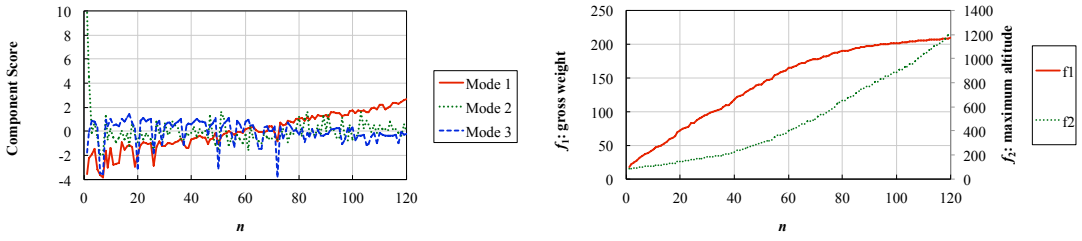
4.6 Feature analysis of the design modes obtained

Here we confirm what types of design modes are derived on the hybrid rocket engine design through our proposed design mode analysis. We focus on the C1-1 (equals the entire dataset). Table 4.3 shows the summary of design mode obtained in C1-1. We think the effective design modes are from mode 1 to mode 3 because the cumulative proportion of the variance attains 0.80 when the mode-3 vector is used.

Table 4.3 Summary of design mode analysis on C1-1

Design mode (i)	1	2	3	4	5	6
Standard deviation ($\sqrt{\lambda_i}$)	1.52	1.22	0.99	0.84	0.70	0.18
Proportion of variance	0.38	0.25	0.16	0.12	0.08	0.01
Cumulative proportion	0.38	0.63	0.80	0.91	0.99	1.00

Fig. 4.18 shows the distribution of the component score of each design mode and the objective function values. The horizontal axis shows each solution numbered in ascending order of objective function values. The component score is the projection of the data along the principal component direction (each design mode). We can easily understand that the mode 1 significantly affects the objective function values. Moreover, it is interesting that the solution with the minimum objective function values is very distinctive from other solutions in mode 2. It seems to be characteristic design in this problem and may be difficult to represent it without using the mode-2 vector.



(a) Component score of mode 1 - 4

(b) Layer 2

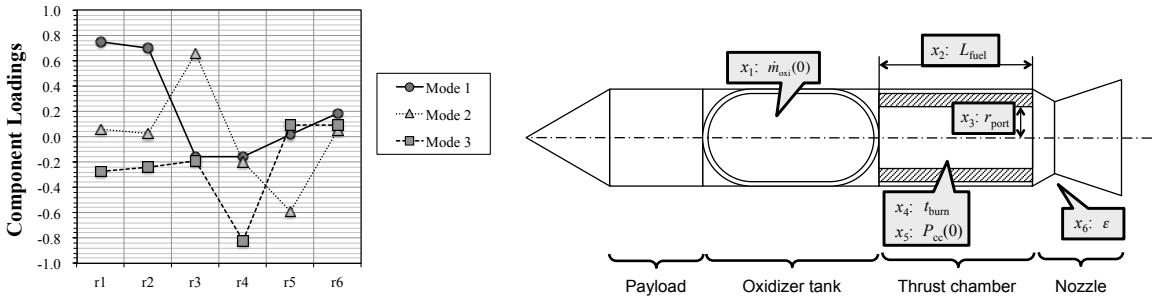
Fig. 4.18 Distribution of component score (mode 1 - 3) and the objective function values in C1-1

In order to know which decision variables are dominant in this design problem, the component loading of each design mode is analyzed, as shown in Table 4.4 and Fig. 4.19. Here, r_i is the i-th element of component loading of each mode, and shows correlation with decision variable x_i. The results show that each design mode is correlated with different decision variables from each

other. As mentioned above, in this problem, mode 1 is dominant. From the component loading of mode 1, we can understand that the decision variables x_1 and x_2 is highly correlated. It can be expected that we can easily make any designs on the nondominated solution set by changing the x_1 and x_2 along the direction indicated by the mode-1 vector.

Table 4.4 Component loadings

Mode (i)	1	2	3	Correlated x_i
r_1	**0.75**	0.06	-0.28	initial mass flow of an oxidizer $\dot{m}_{\mathrm{oxi}}(0)$ [kg/sec]
r_2	**0.70**	0.03	-0.24	fuel length L_{fuel} [m]
r_3	-0.16	**0.66**	-0.19	initial radius of a port $r_{\mathrm{port}}(0)$ [m]
r_4	-0.15	-0.20	**-0.82**	combustion time t_{burn} [sec]
r_5	0.02	**-0.59**	0.09	initial pressure in combustion chamber $P_{\mathrm{cc}}(0)$ [MPa]
r_6	0.18	0.05	0.09	aperture ratio of nozzle ϵ [-]



(a) Distributions of component loadings

(b) Illustration of decision variables

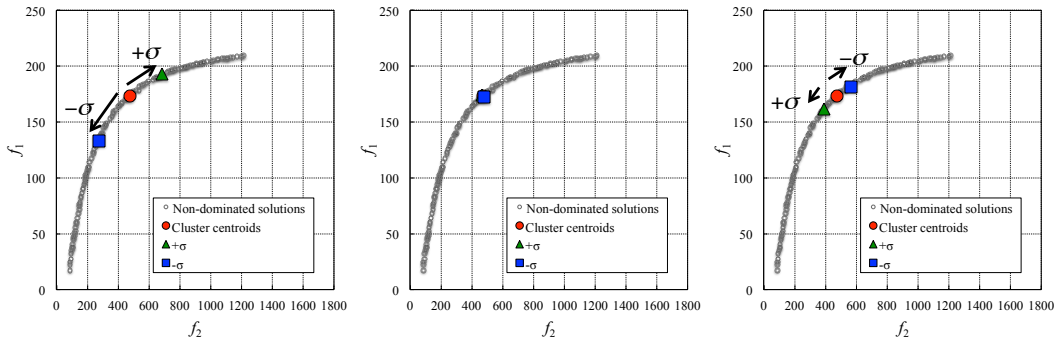
Fig. 4.19 Distributions of component loadings and correlated decision variables

Next, to see the response on the objective space when the decision variables are changed along each design mode, new solutions are generated and evaluated based on the design modes. The mean vector $\boldsymbol{u}$ is used as a base design, and its decision variable is changed by adding the each design mode vector in positive and negative direction. Each solution $X^k_{\pm\sigma}$ along the design mode k is generated by:

$$X^k_{\pm\sigma} = \boldsymbol{u} \pm \begin{bmatrix} s_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & s_m \end{bmatrix} \boldsymbol{v}_k \tag{4.19}$$

where $\boldsymbol{v}_k$ is the k-th design mode vector (eigenvector), and s_i is standard deviation of the decision variable x_i. The solutions generated here are evaluated on the objective space ($f_i(X^k_{\pm\sigma})$ is calculated) and plotted in Fig. 4.20.

The response in the objective space along the mode-1 direction reveals that we can generate the higher altitude designs by changing the decision variables on the positive direction of mode-1



(a) Mode-1 direction (b) Mode-2 direction (c) Mode-3 direction

Fig. 4.20 Response in objective space when the decision variables are changed in each design mode direction

vector. In this case, mode-1 vector suggests us to increase the initial mass flow of an oxidizer and fuel length. On the other hand, if the decision variables are changed on mode-1 negative direction, lightweight designs are obtained.

Parameter change in mode-2 direction does not affect the objective function values, although it can be sensitive in other objective functions which was not considered here.

Mode-3 direction, which is correlated with combustion time, can affect the objective function values. The oxidizer weight is found by integration of the mass flow of an oxidizer with integral range of [0, t_{burn}]. The fuel weight is also found by integration of the mass flow of fuel, with the same integral range. If the combustion time is decreased, the oxidizer weight and the fuel weight are decreased. Mode-3 vector is negatively correlated with the combustion time as shown in Table 4.4. The combustion time can theoretically affect the total mass of the rocket. It has been verified that the design mode analysis could successfully extract the dominant decision variables, and different types of design mode are obtained by our proposed analysis method.

Here, **three typical design modes are obtained in this design problem**:

[I] **Averaging the nondominated solution set**

Well-balanced design in the altitude and the total gross vehicle weight is obtained.

[II] **Increasing or decreasing the initial mass flow of an oxidizer and fuel length**

Higher altitude designs or lightweight designs are obtained.

[III] **Increasing or decreasing the combustion time**

The total gross vehicle weight can be controlled.

We can easily make decisions by choosing preferable design modes depending on the situation.

4.7 Summary of this chapter

A design mode analysis of Pareto solution sets that supports human decision making was proposed and developed. The design mode of the Pareto solution set was extracted by PCA. It was demonstrated that any design in the Pareto set could be represented by a linear combination of the eigenvectors of the base design. From this finding, the author 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. It was 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.

Chapter 5

Application to Time-series Analysis of fNIRS Data

5.1 Introduction

The design mode analysis has its origin in engineering design problem, but its framework is general and versatile. Especially, the case studies described above dealt with non-time series data. In this section the proposed method is applied to time-series data analysis, taking functional near-infrared spectroscopy (fNIRS) data as an example. How to adopt the proposed method to time-series data (data representation, setup for data clustering and interpretation of design mode) is discussed.

5.2 fNIRS measurement and analysis

fNIRS allows us to monitor and visualize the blood flow in human brain, by measuring the hemodynamic responses of oxy- and deoxy-hemoglobin (Hb). Multiple probes to detect oxy- and deoxy-Hb change [mM mm] are placed on the head as shown in Figure 5.1. fNIRS measurement is becoming popular with its advantage of non-invasive, easy to use and low cost. Many researches focus on the development of analytical methods to statistically reveal the functional activities of the brain and human brain networks by finding brain activation patterns in the time-series oxy- and deoxy-Hb change [56].

In the fNIRS measurement, we observe the Hb change while human subject is performing some tasks as an stimuli for the brain activities. It should be noted that we have to insert some rest period prior to the task period in order to detect the baseline of the signal and derive the brain response to the task, because fNIRS can measure only the relative change of Hb but the

Fig. 5.1 fNIRS measurement

absolute value of Hb concentrations. Thus, in the measuring period, the rest and task periods are alternated several times. This experimental design is called a block design [57], and illustrated in Figure 5.2. This research also uses it for the experiment.

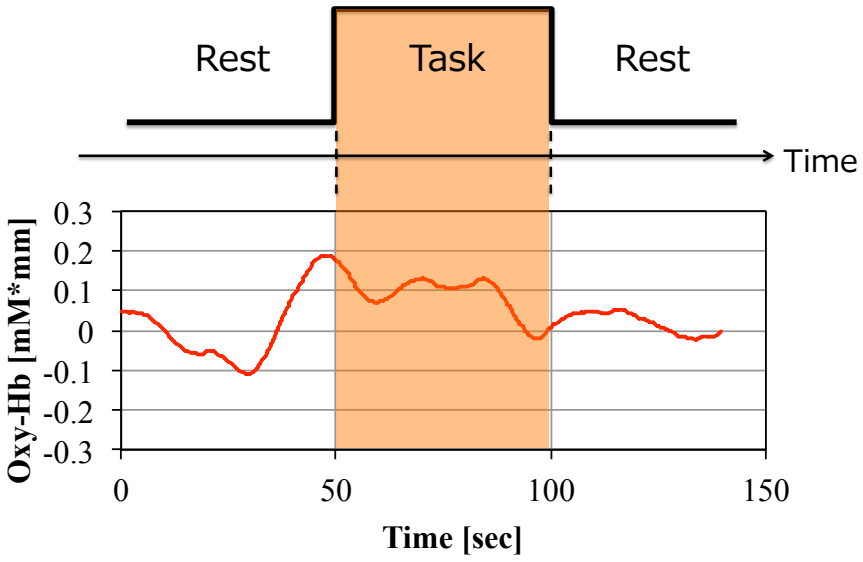


Fig. 5.2 A block design used for the experiment in this research

5.3 Analysis of fNIRS data, as a ROI determination problem

Since the multichannel fNIRS system enables us the plural real-time data acquisitions within a single measurement, another difficulty arises when we deal with these multichannel fNIRS signals; which channels should we focus on? At the beginning of the analysis, we have to determine the region-of-interest (ROI) channels that are considered to be worth to analyze. Generally, there are three reasons to determine ROI channels; (1) workload reduction for statistical tests, (2) reducing Type I error by limiting the number of the statistical tests to a few ROIs and (3) limiting the region to be analyzed, where is functionally defined on the basis of some other information [58]. ROIs should be defined with a structural or functional perspectives. The simplest way to define the structural ROIs is to relate manually the fNIRS channel map with anatomically defined brain regions. It is considered to be a theoretical approach. Functional ROIs can be identified by finding meaningful activation patterns within the fNIRS signals obtained from the same individual, that is, a data-driven approach. As mentioned above, fNIRS has poor spatial resolution, it is difficult to correctly make the correlation between the channel maps and the brain regions. That is why, a data-driven approach is effective to its advantage. ROIs should be discussed with both approaches that complement each other.

One of the most important things to take the data-driven approach is to roughly grasp response patterns of blood flow change (fNIRS signals). Once we found them, each pattern could be related with any brain function or region. However, it is very difficult to extract the representative patterns from the multiple (many subjects') and high-dimensional (multichannel) dataset such as fNIRS signals, through one by one consideration of each data.

With this background, we try to apply our proposed design mode analysis to fNIRS dataset, in order to analyze the multiple-channel blood flow changes and determine the ROIs.

5.4 How can the design mode interpreted in fNIRS data analysis?

The essence of the proposed design mode analysis is that it decomposes the dataset into the mean vector and the linear combination of several orthogonal basis vectors (eigenvector, referred to as design mode in this research). The mean vector indicates the representative base structure of dataset (or each cluster of it), and each design mode characterizes the perturbation to the base structure in order to represent any data in the given dataset.

Here, the following interpretations are assumed in application of design mode analysis to the fNIRS data analysis:

- The mean vector (signal) of each cluster obtained by the design mode analysis can be associated with the functional activity of the brain. (Each cluster may be related to any brain function.)
- The eigenvector (design mode) of each cluster can be the fluctuation affected by the mean vector of other cluster.

These assumptions are based on the idea of brain functional localization and functional connectivity among them. Figure 5.3 illustrates them.

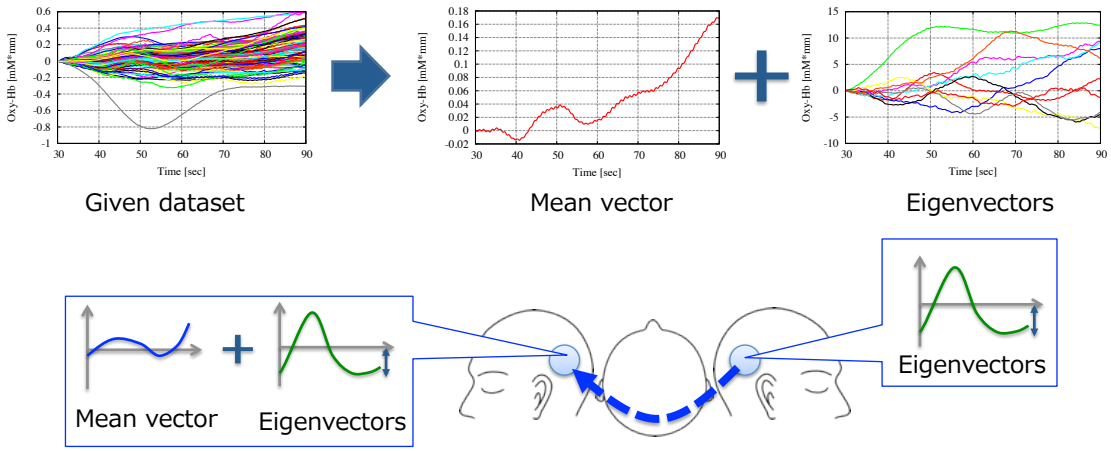


Fig. 5.3 Interpretation of design mode analysis of fNIRS dataset

Experiments and discussions in this chapter examine the possibility of the design mode analysis as an analytical tool for exploring the brain function network.

5.5 Design mode analysis on fNIRS data

The following subsections describe the application example of design mode analysis to the fNIRS data.

5.5.1 Data representation of fNIRS signal

Design mode analysis requires the dataset to be a set of design variables. Let $y(t)$ is the oxy- or deoxy-Hb change [mM mm] at time t, and then a fNIRS signal Y with sample time Δt is represented as:

$$Y = (y(t_0), y(t_1), \ldots, y(t_{n+1})) \quad (t_i = t_0 + i\Delta t) \tag{5.1}$$

Here it should be noted that the baseline of $y(t)$ depends on the situation (i.e., person, experimental environment, and the type of tasks), because fNIRS detect anything more than the relative

change of oxy- or deoxy Hb. For this reason, the difference $y(t + \Delta t) - y(t)$ is calculated and defined as a design variable in design mode analysis. A set of design variables X is represented as follows:

$$X = (x_1, x_2, \ldots, x_n) = (y(t_1) - y(t_0), y(t_2) - y(t_1), \ldots, y(t_{n+1}) - y(t_n)) \tag{5.2}$$

By this representation, the baseline dependency could be canceled.

5.5.2 Distance metric for data clustering

As the design mode analysis includes the data clustering process in its framework, the distance metric to evaluate the similarity between two fNIRS signals is required. In this research, dynamic time warping (DTW) [59] is employed as a distance measure, because it can handle time shifting and scaling even if the signal may vary in time or speed. DTW finds an optimal match of two signals by dynamic programming, allowing time-domain stretching and shrinking. The algorithm of DTW is shown below:

Algorithm 5.1 Dynamic time warping [60]

1: Let $A = (a_1, a_2, \ldots, a_m)$ and $B = (b_1, b_2, \ldots, b_m)$ are the time-series signals.

2: $\mathrm{DTW}(\phi, \phi) = 0$

3: $\mathrm{DTW}(A, \phi) = \mathrm{DTW}(\phi, B) = \infty$

4: $\mathrm{DTW}(A, B) = \mathrm{dist}(a_1, b_1)$

$$+min \begin{cases} \mathrm{DTW}(A, (b_2, b_3, \ldots, b_m)) \\ \mathrm{DTW}((a_2, a_3, \ldots, a_n), B) \\ \mathrm{DTW}((a_2, a_3, \ldots, a_n), (b_2, b_3, \ldots, b_m)) \end{cases}$$

where $\mathrm{dist}(a_i, b_j)$ is the distance between two elements a_i and b_j, and any types of distance measure can be used. This research employed L_1 distance $|a_i - b_j|$.

5.5.3 Experiments to obtain the fNIRS dataset

To obtain the fNIRS dataset, the experiment is performed using fNIRS measurement system ETG-7100 (Hitachi Medical Corporation). Five probes (24 channels $\times$ 3 probes and 22 channels $\times$ 2 probes, totally 116 channels) are placed according to international 10/20 system as shown in Figure 5.4.

The cerebral blood flow change are measured during stereopsis. The experimental design is shown below. Figure 5.5 also illustrates the procedure of the experiment.

1. Start of the experiment: Rest for 30 sec.

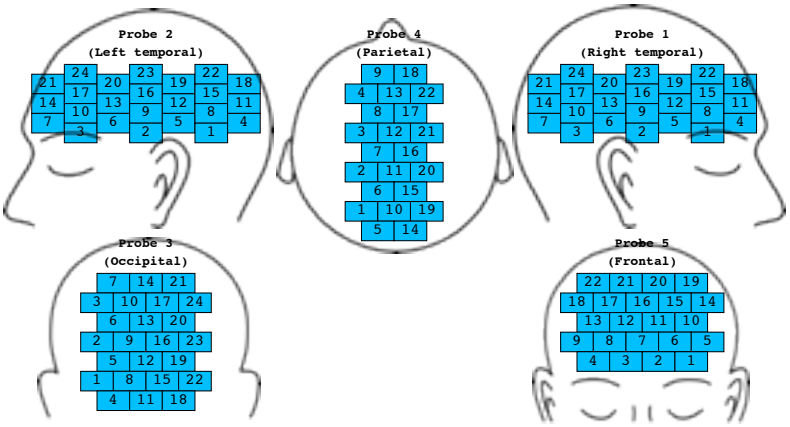


Fig. 5.4 Channel assignment

2. Task period: Repeat the following two steps for 60 sec.

   (a) Rest for 2 sec.

   (b) Gaze at stereogram sheet. If it is recognized as stereoscopic image, back to (a). The number of the sheet recognized is counted. 10 sec has passed with no recognition, back to (a).

3. End of the experiment: Rest for 50 sec.

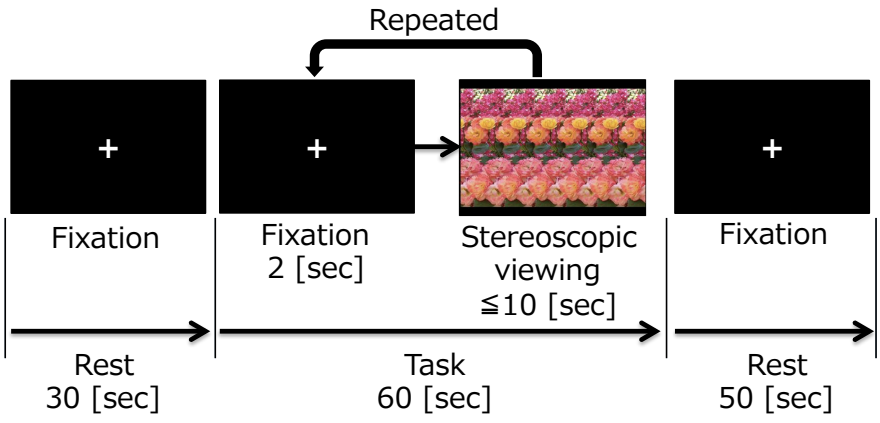


Fig. 5.5 Experimental design for stereoscopic task

Room temperature and humidity were kept at 21.1 – 23.9 [degC] and 28 – 46 [%] respectively, through the experiment.

5.5.4 Setup for numerical experiments

For the design mode analysis, the fNIRS signal during the task period is extracted and represented as decision variables. As the task period is 60 sec and the sampling period is 0.1 sec, the number of decision variables is 600. For a single person data, 116-channels data can be totally obtained, but the data with experimental error (warned by measurement system ETG-7100) would be eliminated. DTW distance is utilized for the distance measure in k-means data clustering and the approximation error of design approximation. In Section 4.3.2, SSE is used as the approximation error, but the DTW distance is substituted for it in the case of fNIRS data analysis.

5.5.5 Results and discussions

Figure 5.6 shows the fNIRS data clusters after the design mode analysis. A dataset consists of 96-channel signals of a single subject (20 channels are removed due to the experimental error).

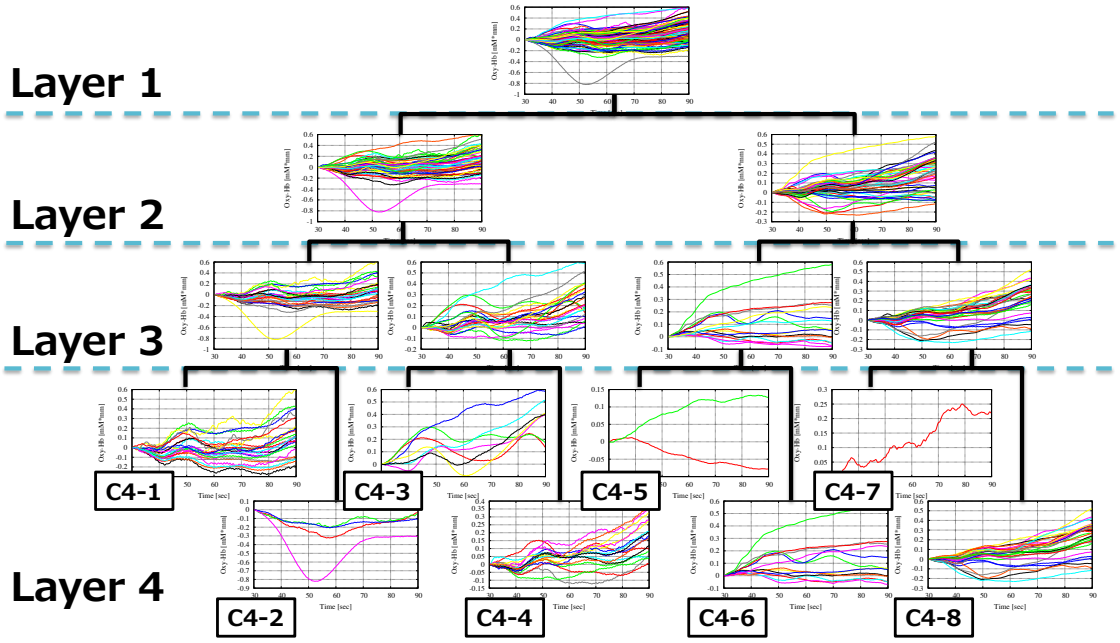


Fig. 5.6 Clusters obtained by design mode analysis (Note that each signal is plotted by cumulatively summing each design variables x_i.)

It can be seen that similar waveforms are gathered within each cluster. Here, 8 clusters in layer 4 (C4-1 to C4-8) are examined based on the framework of design mode analysis. First of all, the channel distribution of each cluster is represented in Figure 5.7. In Chapter 4 author revealed that the characteristics of each cluster could be analyzed by examining the mean vector and design mode (eigenvector) obtained by design mode analysis. In the same way, each cluster of fNIRS

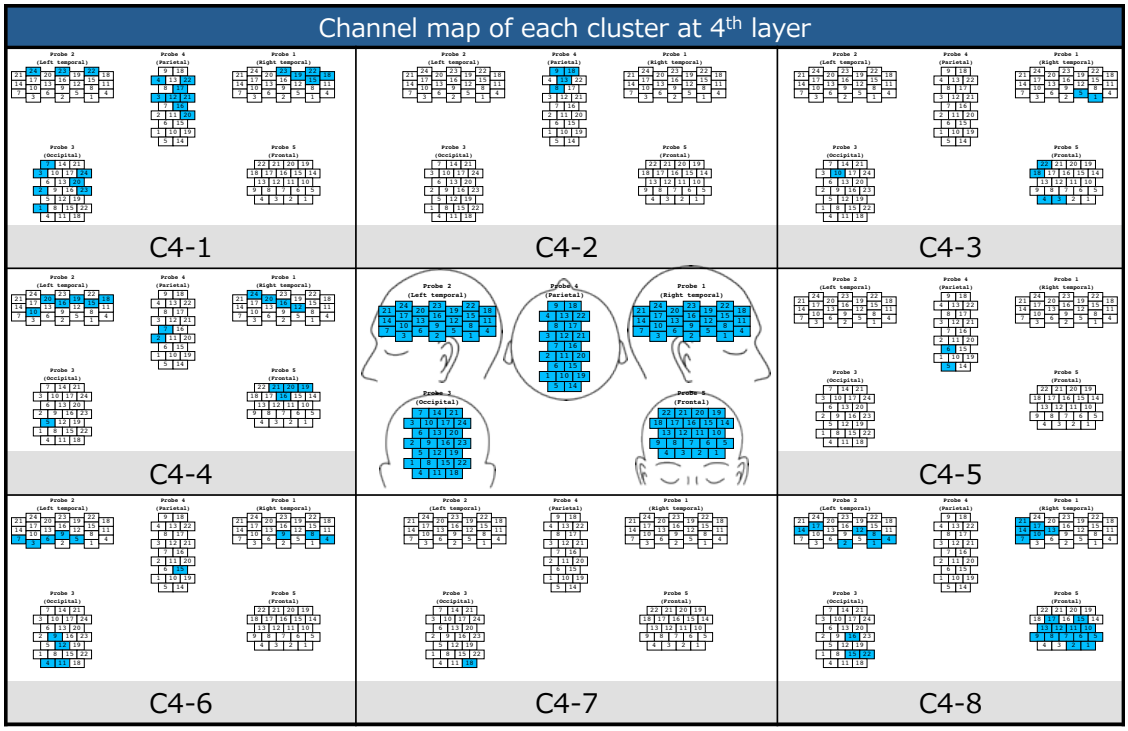


Fig. 5.7 Channel distribution of 8 clusters in layer 4

signal will be analyzed with the mean vector and the design modes. For the mean vector (mean signal), it is expected that the average response to the stimuli may be associated with some kind of brain activities.

In this experiment, the cerebral blood flow change were measured during stereopsis, so that the cerebral region related with the visual tasks should organize the cluster. Once the visual information is given, it is first processed in visual cortex located in occipital area. Then, it takes either of two different passage, dorsal visual pathway and ventral visual pathway. Dorsal pathway is considered to be related with depth perception and motion, while ventral pathway is associated with recognizing color information [61]. Figure 5.8 shows the mean signals and channel maps of three cluster C4-1, C4-4, and C4-6. In C4-1 and C4-4, the mean signals are rising. Channel maps seem to be express the brain activity pathway from the visual cortex to the dorsal visual pathway. On the other hand, C4-4 seem to be the cluster of signals along the ventral visual pathway.

Moreover, the relationship between the channel map of C4-3 and C4-8 and the functional areas of brain are supposed in Figure 5.9. The right inferior frontal gyrus which adopts attentional control [62], and frontal association area, organ of decision of action and prediction, are indicated in Figure 5.9. C4-3 is possibly related to the right inferior frontal gyrus because of its channel distribution. Its mean signal is also rising. With the same reason, C4-8 have the potential to

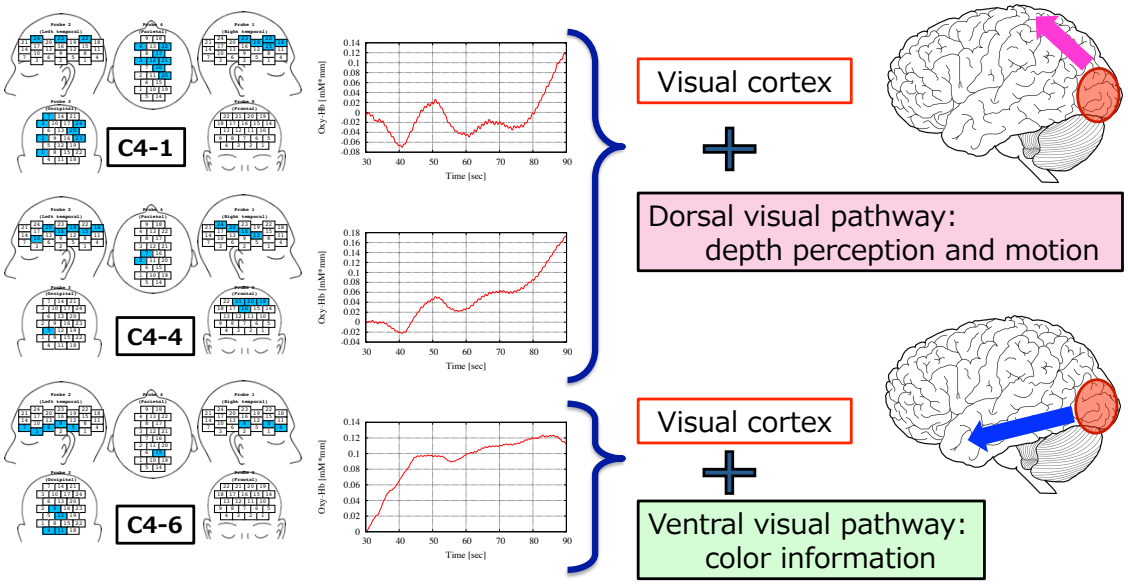


Fig. 5.8 Channel map and mean vector (C4-1, C4-4, C4-6)

associated with frontal association area.

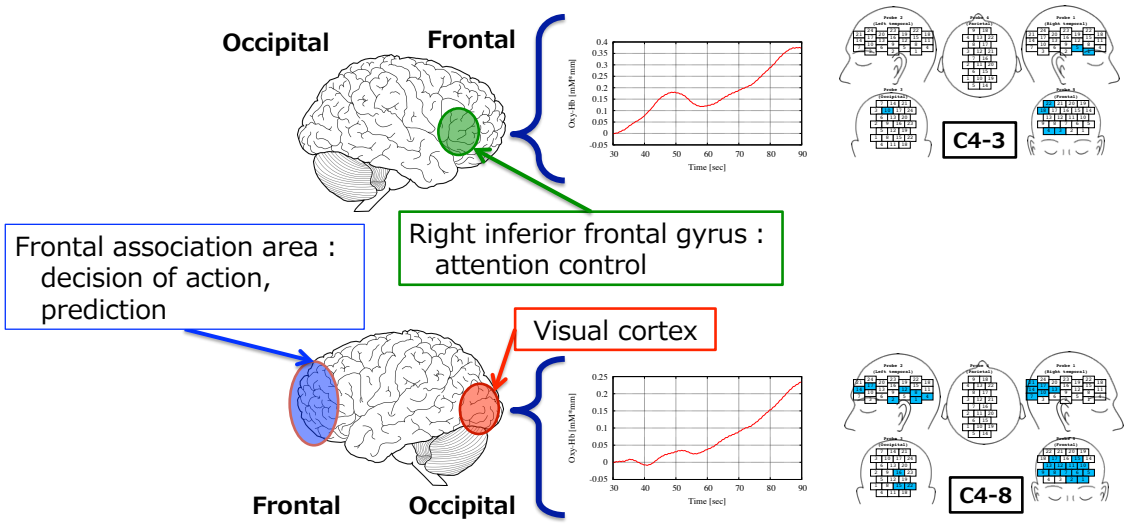


Fig. 5.9 Channel map and mean vector (C4-3 and C4-8)

Secondly, in order to investigate the assumption that the design mode each cluster can be the fluctuation affected by the mean vector of other cluster, for each design mode of a cluster, the nearest mean vector will be found among all the clusters. The DTW distance was used to measure closeness. If the design mode that is close to the mean vector of other cluster exists, it may have possibility to be the fluctuation caused by other brain activity. C4-1 dataset is chosen as an example of this analysis. Figure 5.10 is the result of the analysis. For the first and second

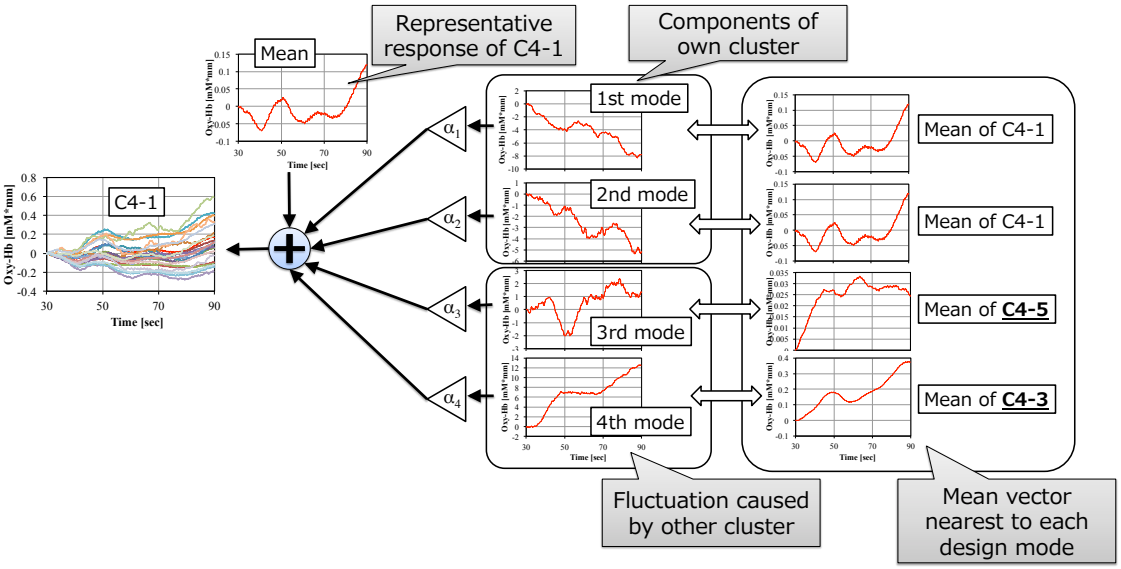


Fig. 5.10 Characterization example of design mode (C4-1)

mode, the nearest mean vector have been found within C4-1, their own cluster. It may be one of the components of C4-1 dataset. However, for the third and fourth design mode, the nearest mean vector exist in other cluster, C4-5 and C4-3, respectively. They may be the fluctuations from the response of C4-5 and C4-3. Another assumption that there are brain network between the C4-5 and C4-3 and the C4-1 cluster may arise.

As such observations on the mean vector and the channel distribution of each cluster are a matter of speculation, statistical tests should be applied for the sake of theoretical righteousness (although it is beyond the focus of this research). However they make us more conscious for generating hypotheses from the dataset.

5.5.6 Summary of this chapter

This chapter demonstrated the another face of proposed design mode analysis, especially on time series data. fNIRS dataset is chosen for this application, and how to incorporate the design mode analysis into the time series data, data representation, interpretation of the mean vector and design mode are discussed. The following conclusion is obtained through the discussion in this chapter:

- Time series data can be analyzed with the framework of design mode analysis proposed in this disseration.

- The mean vector and design mode may be characterized from the point of the brain func-

tions and network.

- We can easily choose ROI channels where the correlations were found by design mode analysis.

Chapter 6

Conclusions

6.1 Contributions

Real-world engineering design problem usually faces multiple conflicting criteria and high-dimensional decision variables. At an early stage of development, especially at the conceptual design, it is necessary to grasp the possible design candidates and extract promising one, for efficiency of development process. For such background, the following important tools were proposed and analyzed:

[I] **Development of novel evolutionary multiobjective optimization (EMO) framework to obtain the abundant variety of candidate design.**

Concept: Novel search scheme to obtain well-converged and well-spread Pareto solutions were proposed and developed.

Effectiveness: Its effectiveness was verified on the mathematical complex optimization problems (WFG test suites).

Versatility: Its general versatility was verified through implementation of proposed framework using two popular MOEAs, NSGA-II and MOEA/D. Even if a new algorithm is developed, we can easily improve its performance by incorporating proposed EMO framework.

[II] **Development of design mode analysis to extract the representative design patterns from the candidates and analyze their characteristics.**

Concept: New concept ”design mode” was incorporated. The essence is that any designs in dataset can be represented by base (mean) design and a linear combination of the orthogonal basis vector, design mode. We can analyze the target design problem by

decomposing them into the mean vector and the orthogonal basis vector (an eigenvector was used in this work).

Effectiveness: Its effectiveness was verified thorough two engineering design problem, multiobjective knapsack problem and the conceptual design of hybrid rocket engine. Both cases revealed that we could extract the representative design patterns and got design strategies from the design mode obtained.

Versatility: As an another application, time series data of fNIRS was analyzed using proposed design mode analysis. How to incorporate the time series data into the proposed method, was introduced. The proposed design mode analysis can handle both non-time series and time series data without modifying the original framework. Engineers will easily use them in their own problems.

6.2 Future work

[I] **Development of novel evolutionary multiobjective optimization (EMO) framework to obtain the abundant variety of candidate design.**

Implementation using other base algorithms: In this dissertation, NSGA-II or MOEA/D were used for MOEA population, and DGA was used for SOEA population. Other popular algorithms should be implemented and verified for the sake of higher versatility.

Application to many-objective problems: Only two or three objective cases were investigated in this work. In case of the real-world problems, many objective functions may exist. The proposed framework should be verified on many-objective cases.

[II] **Development of design mode analysis to extract the representative design patterns from the candidates and analyze their characteristics.**

Improvement of data clustering process: The current form of the design mode analysis divide each cluster into two new clusters. However two clusters is not always true. The automatic determination of the cluster size or other type of the clustering method which does not requires its determination should be investigated.

Evaluation of distance measure: Distance measure is very important in the proposed method because it is used in both data clustering and design approximation. Changing dis-

tance measure may lead to different analytical results. Its effect should be further discussed.

Acknowledgement

本研究を遂行するにあたり多大なるご指導ならびにご支援を賜りました，同志社大学 生命医科学部 廣安知之教授に心より感謝申し上げます．廣安先生は，一度修士課程を卒業し一般企業に就職した私に，再度研究を始めるきっかけを与えてくださっただけでなく，博士課程に入学して研究を進める環境もご提供頂きましたこと，ここでは書き尽くせない感謝の気持ちでいっぱいです．無事このように博士論文をまとめるに至ったのも，廣安先生のご理解とご支援をいただいたお陰です．誠にありがとうございました．この研究を通じて得た知見，考え方を武器に今後も研究開発に邁進致します．

次に，学部時代から修士課程の間のご指導を賜り，博士論文の副査をお引き受け頂きました同志社大学 理工学部 三木光範教授に心より感謝申し上げます．三木先生には最適化の基礎から，研究の進め方，報告の仕方など，研究者としてのベースを教わりました．そして何より，研究の楽しさを教えていただきました．全ては 2004 年 3 月に，知的システムデザイン研究室の門をくぐったことが始まりでした．その入り口に導いてくださった三木先生に心から感謝致します．

同じく，副査をお引き受け頂きました同志社大学 生命医科学部 太田 哲男教授に，心より感謝申し上げます．先生に最初にご挨拶させていただいたのは，私の博士後期課程の面接をご担当頂いたときでした．その後もプロジェクト演習での発表会や博士論文の中間審査等，折に触れて私の研究に対してご指導いただきました．私とは異なる分野での専門家としてのご指導は，私だけでは認識できないであろう新たな気づきを与えてくださいました．何より，私の研究の重要性についてご理解とご支援をいただきましたことが大変嬉しく，感謝の気持ちでいっぱいです．ありがとうございました．

研究活動を行うにあたり，医療情報システム研究室の皆様には，多大なるご迷惑をおかけし，また，それ以上のご支援をいただきました．2010 年の 4 月に OB として突然やってきて研究を始め，2012 年に博士後期課程に入学し，実に 5 年の歳月の中，現役の学生だけでなく，卒業していった皆様まで，多くの学生の皆様にご支援を頂きました．皆様無くして私の研究活動は成り立ちませんでした．本当にありがとうございます．

パナソニック株式会社の同僚の皆様には，研究活動が佳境になった際には業務上のでご

配慮を頂く等，ご支援をいただきました．業務とは何の関係が無い私的な研究活動であるにも関わらず，平日の講義や発表などのためにお休みを頂くこともあり，ご迷惑もおかけしてきました．皆様の多大なるご支援に心より感謝申し上げます．

最後に，仕事と大学での研究を両立させる上で，日々の生活を支えてくれた両親と，最終年度で忙しくなる時期に結婚し，文句一つ言わず応援してくれた妻に感謝します．

Publications

1. Journal articles

1. S. Hiwa, T. Hiroyasu, M. Miki, ”Design Mode Analysis of Pareto Solution Set for Decision-Making Support,” *Journal of Applied Mathematics*, vol. 2014, Article ID 520209, 15 pages (Nov 2014).
2. S. Hiwa, M. Nishioka, T. Hiroyasu, M. Miki, ”Novel Search Scheme for Multi-Objective Evolutionary Algorithms to Obtain Well-Approximated and Widely Spread Pareto Solutions,” *Swarm and Evolutionary Computation*, Volume 22, Pages 30-46, ISSN 2210-6502, http://dx.doi.org/10.1016/j.swevo.2015.01.004. (June 2015)

2. International conference proceedings

3. M. Miki, S. Hiwa, T. Hiroyasu, ”Simulated Annealing using an Adaptive Search Vector”, *Cybernetics and Intelligent Systems, 2006 IEEE Conference on*, 2006, pp.1–6.
4. S. Hiwa, T. Hiroyasu, M. Miki, ”Hybrid optimization using DIRECT, GA, and SQP for global exploration”, in: *Evolutionary Computation, CEC 2007. IEEE Congress on*, 2007, pp.1709–1716.
5. S. Hiwa, T. Hiroyasu, H. Yokouchi, M. Miki, M. Nishioka, ”Reference point-based search scheme for multiobjective evolutionary algorithm,” in: *Soft Computing and Intelligent Systems (SCIS) and 13th International Symposium on Advanced Intelligent Systems (ISIS), 2012 Joint 6th International Conference on*, 2012, pp. 1666–1672.

3. Domestic conference proceedings

6. 三木光範，廣安知之，日和 悟，實田 健「ダミー目的関数をもつ多目的確率的山登り法」情報処理学会 情報科学技術フォーラム, 2004
7. 日和 悟，三木光範，廣安知之「多目的シミュレーテッドアニーリングを用いた知的 LED 照明システム」情報処理学会 第 67 回全国大会, 2005

8. 三木光範，日和 悟，廣安知之「適応的探索ベクトルをもつシミュレーテッドアニーリング」日本機械学会 第 18 回計算力学講演会, 2005
9. 日和 悟，廣安知之，三木光範「大域的最適化のための複数最適化手法の動的制御法」日本機械学会 第 16 回設計工学・システム部門講演会, 2006
10. 日和 悟，廣安知之，三木光範「大域的最適化のための複数最適化手法の動的制御法」日本機械学会 第 7 回最適化シンポジウム, 2006

4. Bulletin papers

11. 三木光範，日和 悟，廣安知之「LED を用いた調色用照明システムの基礎的検討」同志社大学理工学研究報告 Vol.46, No.3, pp 9-18, 2005

Bibliography

[1] Fujita, K.: 設計における上流とその意味 (In Japanese), *Mechanical Engineering Congress, Japan (MEJC)*, Vol. 2004, No. 8, pp. 374–377 (2004).

[2] Asiedu, Y. and Gu, P.: Product life cycle cost analysis: State of the art review, *International Journal of Production Research*, Vol. 36, No. 4, pp. 883–908 (1998).

[3] Bonissone, P., Subbu, R. and Lizzi, J.: Multicriteria decision making (mcdm): a framework for research and applications, *Computational Intelligence Magazine, IEEE*, Vol. 4, No. 3, pp. 48–61 (2009).

[4] Mathur, V. K.: How Well Do We Know Pareto Optimality?, Vol. 22, No. 2, pp. 172–178 (1991).

[5] Chankong, V. and Haimes, Y.: *Multiobjective decision making: theory and methodology*, North-Holland series in system science and engineering, North Holland (1983).

[6] Morizawa, S., Shimoyama, K., Obayashi, S., Funamoto, K. and Hayase, T.: Implementation of visual data mining for unsteady blood flow field in an aortic aneurysm., *J. Visualization*, Vol. 14, No. 4, pp. 393–398 (2011).

[7] Obayashi, S., Jeong, S., Chiba, K. and MORINO, H.: 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).

[8] Oyama, A., Nonomura, T. and Fujii, K.: Data mining of non-dominated solutions using proper orthogonal decomposition, *in Proceedings of the 11th Annual conference on Genetic and evolutionary computation, GECCO '09*, pp. 1935–1936. (2009).

[9] Oyama, A., Nonomura, T. and Fujii, K.: Data Mining of Pareto-Optimal Transonic Airfoil Shapes Using Proper Orthogonal Decomposition, *Journal of Aircraft*, Vol. 47, No. 5, pp. 1756–1762 (2010).

[10] Deb, K., Agarwal, S., Pratap, A. and Meyarivan, T.: A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II, *KanGAL report 200001, Indian Institute of Technology, Kanpur, India* (2000).

[11] Nguyen, T. T., Yang, S. and Branke, J.: Evolutionary Dynamic Optimization: A Survey of the State of the Art, *Swarm and Evolutionary Computation*, Vol. 6, No. 0, pp. 1 – 24 (2012).

[12] Deb, K. and Sundar, J.: Reference Point Based Multi-Objective Optimization Using Evolutionary Algorithms, *GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pp. 635–642 (2006).

[13] Li, K., Kwong, S., Cao, J., Li, M., Zheng, J. and Shen, R.: Achieving Balance between Proximity and Diversity in Multi-Objective Evolutionary Algorithm, *Information Sciences*, Vol. 182, No. 1, pp. 220 – 242 (2012). Nature-Inspired Collective Intelligence in Theory and Practice.

[14] Knowles, J., Thiele, L. and Zitzler, E.: A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers, *TIK Report 214, Computer Engineering and Networks Laboratory (TIK), ETH Zurich* (2006).

[15] Zitzler, E. and Knzli, S.: Indicator-based Selection in Multiobjective Search, *in Proc. 8th International Conference on Parallel Problem Solving from Nature (PPSN VIII*, Springer, pp. 832–842 (2004).

[16] Emmerich, M., Beume, N. and Naujoks, B.: An EMO Algorithm Using the Hypervolume Measure as Selection Criterion, 2005 Int 'l Conference, March 2005, Springer, pp. 62–76 (2005).

[17] Beume, N., Naujoks, B. and Emmerich, M.: SMS-EMOA: Multiobjective Selection Based on Dominated Hypervolume, *European Journal of Operational Research*, Vol. 181, No. 3, pp. 1653–1669 (2007).

[18] Bader, J. and Zitzler, E.: HypE: An Algorithm for Fast Hypervolume-Based Many-Objective Optimization, *Evol. Comput.*, Vol. 19, No. 1, pp. 45–76 (2011).

[19] Zhang, Q. and Li, H.: MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition, *Evolutionary Computation, IEEE Transactions on*, Vol. 11, No. 6, pp. 712 –731 (2007).

[20] Martínez, S. Z. and Coello, C. A. C.: A Direct Local Search Mechanism for Decomposition-based Multi-Objective Evolutionary Algorithms, *IEEE Congress on Evolutionary Computation*, pp. 1–8 (2012).

[21] Nelder, J. A. and Mead, R.: A Simplex Method for Function Minimization, *Computer Journal*, Vol. 7, pp. 308–313 (1965).

[22] Bosman, P. A. N.: On Gradients and Hybrid Evolutionary Algorithms for Real-Valued Multiobjective Optimization, *Evolutionary Computation, IEEE Transactions on*, Vol. 16, No. 1, pp. 51–69 (2012).

[23] Okuda, T., Hiroyasu, T., Miki, M. and Watanabe, S.: DCMOGA: Distributed Cooperation model of Multi-Objective Genetic Algorithm, *Advances in Nature-Inspired Computation: The PPSN VII Workshops*, pp. 25–26 (2002).

[24] Ishibuchi, H. and Shibata, Y.: Mating Scheme for Controlling the Diversity-Convergence Balance for Multiobjective Optimization, *GECCO (1)*, pp. 1259–1271 (2004).

[25] Dubois-Lacoste, J., Lpez-Ibez, M. and Sttzle, T.: Combining Two Search Paradigms for Multi-objective Optimization: Two-Phase and Pareto Local Search, Vol. 434, pp. 97–117 (2013).

[26] Paquete, L. and Sttzle, T.: A Two-Phase Local Search for the Biobjective Traveling Salesman Problem, Vol. 2632, pp. 479–493 (2003).

[27] Wickramasinghe, U. K. and Li, X.: Integrating User Preferences with Particle Swarms for Multi-Objective Optimization, *Proceedings of the 10th annual conference on Genetic and evolutionary computation*, GECCO '08, New York, NY, USA, ACM, pp. 745–752 (2008).

[28] Molina, J., Santana, L. V., Hernandez-Daz, A. G., Coello Coello, C. A. and Caballero, R.: g-Dominance: Reference Point Based Dominance for Multiobjective Metaheuristics, *European Journal of Operational Research*, Vol. 197, No. 2, pp. 685–692 (2009).

[29] Asad Mohammadi, M. N. O. and Li, X.: Reference Point Based Multi-objective Optimization Through Decomposition, *Proceedings of 2012 IEEE World Congress on Computational Intelligence (IEEE WCCI 2012)*, pp. 1150–1157 (2012).

[30] Jaimes, A. L. and Coello, C. A. C.: MRMOGA: Parallel Evolutionary Multiobjective Optimization using Multiple Resolutions, *in 2005 IEEE Congress on Evolutionary Computation (CEC'2005)*, pp. 2294–2301 (2005).

[31] Zitzler, E., Laumanns, M. and Thiele, L.: SPEA2: Improving the Performance of the Strength Pareto Evolutionary Algorithm, *Technical Report 103, Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH) Zurich* (2001).

[32] Kukkonen, S. and Lampinen, J.: An Empirical Study of Control Parameters for The Third Version of Generalized Differential Evolution (GDE3), *Evolutionary Computation, 2006. CEC 2006. IEEE Congress on*, pp. 2002 –2009 (2006).

[33] Bandyopadhyay, S., Saha, S., Maulik, U. and Deb, K.: A Simulated Annealing-Based Multiobjective Optimization Algorithm: AMOSA, *Evolutionary Computation, IEEE Transactions on*, Vol. 12, No. 3, pp. 269 –283 (2008).

[34] Tanese, R.: Distributed Genetic Algorithms, *Proceedings of the Third International Conference on Genetic Algorithms (ICGA)*, pp. 434–439 (1989).

[35] Goldberg, D. E.: *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesly (1989).

[36] Li, H. and Zhang, Q.: Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II, *Evolutionary Computation, IEEE Transactions on*, Vol. 13, No. 2, pp. 284 –302 (2009).

[37] Zapotecas Martínez, S. and Coello Coello, C. A.: A Multi-Objective Particle Swarm Optimizer Based on Decomposition, *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, GECCO '11, New York, NY, USA, ACM, pp. 69–76 (2011).

[38] Li, H. and Zhang, Q.: A Multiobjective Differential Evolution Based on Decomposition for Multiobjective Optimization with Variable Linkages, *Proceedings of the 9th international conference on Parallel Problem Solving from Nature*, PPSN'06, Berlin, Heidelberg, Springer-Verlag, pp. 583–592 (2006).

[39] Chiang, T.-C. and Lai, Y.-P.: MOEA/D-AMS: Improving MOEA/D by an Adaptive Mating Selection Mechanism, *Evolutionary Computation (CEC), 2011 IEEE Congress on*, pp. 1473 –1480 (2011).

[40] yan Tan, Y., chang Jiao, Y., Li, H. and kuan Wang, X.: MOEA/D + Uniform Design: A New Version of MOEA/D for Optimization Problems with Many Objectives, *Computers and Operations Research*, No. 0, pp. – (2012).

[41] Huband, S., Hingston, P., Barone, L. and While, L.: A Review of Multiobjective Test Problems and A Scalable Test Problem Toolkit, *Evolutionary Computation, IEEE Transactions on*, Vol. 10, No. 5, pp. 477 –506 (2006).

[42] Bradstreet, L., Barone, L., While, L., Huband, S. and Hingston, P.: Use of the WFG Toolkit and PISA for Comparison of MOEAs, *Computational Intelligence in Multicriteria Decision Making, IEEE Symposium on*, pp. 382 –389 (2007).

[43] Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. and da Fonseca, V.: Performance Assessment of Multiobjective Optimizers: An Analysis and Review, *Evolutionary Computation, IEEE Transactions on*, Vol. 7, No. 2, pp. 117 – 132 (2003).

[44] Bradstreet, L., While, L. and Barone, L.: A Fast Incremental Hypervolume Algorithm, *Evolutionary Computation, IEEE Transactions on*, Vol. 12, No. 6, pp. 714 –723 (2008).

[45] Huang, V. L., Qin, A. K., Deb, K., Zitzler, E., Suganthan, P. N., Liang, J. J., Preuss, M. and Huband, S.: Problem Definitions for Performance Assessment of Multi-objective Optimization Algorithms, Technical report, Nanyang Technological University, Singapore (2007).

[46] R.B., D. K. . A.: Simulated binary crossover for continuous search space, *Complex System*, No. 9, pp. 115 – 148 (1995).

[47] Durillo, J. J. and Nebro, A. J.: jMetal: A Java framework for multi-objective optimization, *Advances in Engineering Software*, Vol. 42, No. 10, pp. 760 – 771 (2011).

[48] Ishibuchi, H., Sakane, Y., Tsukamoto, N. and Nojima, Y.: Evolutionary many-objective optimization by NSGA-II and MOEA/D with large populations, *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on*, pp. 1758–1763 (2009).

[49] Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II, *Evolutionary Computation, IEEE Transactions on*, Vol. 6, No. 2, pp. 182–197 (2002).

[50] Gandibleux, X.: The MOCO Numerical Instances Library, `http://xgandibleux.free.fr/MOCOlib/index.html`.

[51] Watanabe, S., Ohe, S. and Minato, R.: Development of a multi-granularity design support system using hierarchical clustering for non-dominated solutions, *in World Automation Congress (WAC)*, pp. 1–6 (2010).

[52] Oyama, A.: Design optimization problems in space science, `http://flab.eng.isas.jaxa.jp/member/oyama/realproblems_j.html`.

[53] Kosugi, Y., Oyama, A., Kanazaki, M. and Fujii, K.: Multidisciplinary and Multi-objective Design Exploration Methodology for Conceptual Design of a Hybrid Rocket, *Infotech@Aerospace 2011, American Institute of Aeronautics and Astronautics* (2011).

[54] Chiba, K.: Performance comparison of evolutionary algorithms applied to hybrid rocket problem, *in Soft Computing and Intelligent Systems (SCIS) and 13th International Symposium on Advanced Intelligent Systems (ISIS), 2012 Joint 6th International Conference on*, pp. 1673–1678. (2012).

[55] Kraft, D.: Algorithm 733: TOMP—Fortran modules for optimal control calculations, *ACM Trans. Math. Softw.*, Vol. 20, No. 3, pp. 262–281 (1994).

[56] Tak, S. and Ye, J. C.: Statistical analysis of fNIRS data: A comprehensive review, *NeuroImage*, Vol. 85, Part 1, No. 0, pp. 72 – 91 (2014). Celebrating 20 Years of Functional Near Infrared Spectroscopy (fNIRS).

[57] Raichle, M. E.: Functional Brain Imaging and Human Brain Function, *The Journal of Neuroscience*, Vol. 23, No. 10, pp. 3959–3962 (2003).

[58] Etzel, J. A., Gazzola, V. and Keysers, C.: An introduction to anatomical ROI-based fMRI classification analysis, *Brain Research*, Vol. 1282, No. 0, pp. 114 – 125 (2009).

[59] Sakoe, H. and Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition, *Acoustics, Speech and Signal Processing, IEEE Transactions on*, Vol. 26, No. 1, pp. 43–49 (1978).

[60] Fu, A. W.-C., Keogh, E., Lau, L. Y., Ratanamahatana, C. A. and Wong, R. C.-W.: Scaling and Time Warping in Time Series Querying, *The VLDB Journal*, Vol. 17, No. 4, pp. 899–921 (2008).

[61] Mishkin, M., Ungerleider, L. G. and Macko, K. A.: Object vision and spatial vision: two cortical pathways, *Trends in Neurosciences*, Vol. 6, No. 0, pp. 414 – 417 (1983).

[62] Hampshire, A., Chamberlain, S. R., Monti, M. M., Duncan, J. and Owen, A. M.: The role of the right inferior frontal gyrus: inhibition and attentional control, *NeuroImage*, Vol. 50, No. 3, pp. 1313 – 1319 (2010).