IPSJ Transactions on Mathematical Modeling and Its Applications Vol. 4 No. 1 48–61 (Jan. 2011)

Regular Paper

Performance Analysis of Path Relinking on Many-objective NK-Landscapes

Joseph M. Pasia,[†1,†2] Hernán Aguirre[†1] and Kiyoshi Tanaka[†1]

Path relinking is a population-based heuristic that explores the trajectories in decision space between two elite solutions. It has been successfully used as a key component of several multi-objective optimizers, especially for solving bi-objective problems. Its unique characteristic of performing the search in the objective and decision spaces makes it interesting to study its behavior in many objective optimization. In this paper, we focus on the behavior of pure path relinking, propose several variants of the path relinking that vary on their strategies of selecting solutions, and analyze its performance using several many-objective NK-landscapes as instances. In general, results of the study show that the path relinking becomes more effective in improving the convergence of the algorithm as we increase the number of objectives. Also, it is shown that the selection strategy associated to path relinking plays an important role to emphasize either convergence or spread of the algorithm. This study provides useful insights for practitioners on how to exploit path relinking to enhance multi-objective evolutionary algorithms for complex combinatorial optimization problems.

1. Introduction

Multi-objective optimization is the process of simultaneously finding the set of solutions to problems with two or more objectives. It is often called as many objective optimization when there are at least four objectives. In recent years, many-objective optimization has attracted the interest of many researchers because of the poor performance of the state-of-the-art multi-objective evolutionary algorithms (MOEAs) that are known to be efficient in solving multi-objective problems. One main reason for the inability of MOEAs to generate good solutions is the substantially large number of solutions in every Pareto front levels when the number of objectives of the problem is high [1)–3)]. Since most of these MOEAs use ranking by Pareto dominance, ranking becomes coarser, thus weakening their convergence property as they assign the same ranks to a good many solutions [4)–7)].

Several approaches have been proposed to improve the performances of MOEAs. Most of them introduce improvements that are either based on ranking improvement, dimensionality reduction, use of preference information or weighted sums of objective functions [7)], or objective space partitioning. To improve ranking among solutions, one may modify the concept of Pareto dominance. Examples of this strategy include the contracting or expanding the dominance area of solutions [8)] and using a relaxed form of Pareto dominance via ϵ-dominance [9)] or α-domination [10)]. In dimensionality reduction, the number of objectives are reduced by removing the objectives that are considered redundant using principal component analysis [11),12)] or using a mathematical model that seeks the minimum objective subset while maintaining the dominance structure with a given error, or that finds the objective subset with given size and having the least change in dominance structure [13)]. Methods that use preference information aim to provide MOEAs with a reference point so that it can focus its search on a specific region of the Pareto front [14),15)]. One may also use weighted sums of objectives that forms a scalarizing function as supplement to the fitness values of the individual solutions or as a strategy to obtain the optimal solutions in the convex regions of the search space [16),17)]. Recently, a strategy that performs the search on the subspaces obtained by partitioning the objective space has been implemented [18)]. All these approaches are focused mainly on the management of the objective space taking no or just slight consideration of the decision space. In this paper, we study the behavior of a population-based evolutionary procedure that creates solutions by doing its search in *both* the objective and decision spaces. This procedure is called *path relinking* [19)].

Path relinking was originally proposed by Glover (1996) as a strategy that intensifies the search between elite solutions obtained by a tabu search approach. Because of its intrinsic characteristic of forming a sequence of solutions between and around elite solutions, it has been used to solve several multi-objective combi-

†1 Faculty of Engineering, Shinshu University
†2 Institute of Mathematics, University of the Philippines-Diliman

natorial optimization problems as strategy to enhance the performance of several MOEAs. For example, it has been used effectively as the second phase of a two phase algorithm. The first phase usually applies weighted sum-based methods to find efficient solutions. Using these efficient solutions, the second phase applies path relinking to find other efficient solutions located around them or between two efficient solutions [20)–22)]. In some other applications, path relinking is embedded in ant colony approach [23)], genetic algorithms [24)] and memetic algorithms [25)] to either improve the solutions obtained by these methods or search for more promising solutions. Path relinking has also been used as post-optimization strategy that improves the solutions obtained by metaheuristic approaches [26)].

Although the efficacy of path relinking in solving multi-objective problems has been demonstrated, it has not been used as a stand-alone algorithm but only as key component of different optimizers. For example, the results about the contribution of path relinking reported in Ref. 24) and Ref. 23) were generated also by local search procedures. Moreover, except for Ref. 27) where it considered four-objective knapsack problems, it has not been applied to solve complex many-objective optimization problems.

In this study, we focus on pure path relinking and propose different variants for implementing path relinking for complex combinatorial optimization problems having many objectives. The performances of the different path relinking variants will be useful tools for describing the behavior of path relinking. We also investigate how the different selection strategies associated to path relinking can emphasize either convergence or spread of the algorithm.

To analyze the performance of path relinking on multi- and many-objective combinatorial optimization problems, we use the MNK-landscape models [1),5)] as test instances. We study the behaviour of path relinking on landscapes with $2 \leq M \leq 10$ objectives, $N = 100$ bits, and $0 \leq K_i \leq 50$, $i = 1, 2 \ldots, M$ epistatic interactions. We also assess the performance of path relinking using the results of conventional NSGA-II [28)] as reference. For each combination of M, N and K_i, 50 different landscape models are used. It is important to note that we do not aim to propose a pure path relinking as alternative search procedure for many-objective optimization problems. Rather, we study the performance of path relinking on MNK-landscape to provide useful insights for practitioners on how to exploit the strengths of path relinking to enhance existing MOEAs.

2. Many-Objective Optimization and MNK-Landscapes

Multi-objective optimization (MO) involves simultaneously optimizing a set of two or more, and often conflicting, objective functions. Formally, it can be stated as follows [⋆1]:

$$\max_{\boldsymbol{x} \in \mathtt{X}} \boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_M(\boldsymbol{x})) \quad (1)$$

where $\mathtt{X} \subset \mathtt{R}^N$ is the feasible space, $\boldsymbol{f} : \mathtt{R}^N \rightarrow \mathtt{R}^M$ is a vector-valued objective function, f_i denote the individual objective functions and M is the number of objectives. When M is 4 or more, MO is commonly referred to as *many*-objective optimization (MaO). Moreover, when $\mathtt{X} \subset \mathtt{Z}^N$, MO is known as multi-objective *combinatorial* optimization (MOCO).

In general, there are several optimal solutions, also called efficient, Pareto optimal or nondominated solutions, to MO problems. A solution $\boldsymbol{x}$ is nondominated if there exists no other feasible solution $\boldsymbol{y}$ such that $f_i(\boldsymbol{y}) \geq f_i(\boldsymbol{x})$, for $i = 1, 2, \ldots, M$ and $f_i(\boldsymbol{y}) > f_i(\boldsymbol{x})$ for some i. If such solution $\boldsymbol{y}$ exists, then we say that $\boldsymbol{y}$ *dominates* $\boldsymbol{x}$ ($\boldsymbol{y} \succ \boldsymbol{x}$). The image of the nondominated solutions in the objective space is called Pareto front or efficient frontier.

Other dominance relations between any two feasible solutions $\boldsymbol{x}$ and $\boldsymbol{y}$ also exist. For example, $\boldsymbol{y}$ *strictly dominates* $\boldsymbol{x}$ ($\boldsymbol{y} \succ\succ \boldsymbol{x}$) if $\boldsymbol{f}(\boldsymbol{y}) > \boldsymbol{f}(\boldsymbol{x})$ [⋆2] and $\boldsymbol{y}$ *weakly dominates* $\boldsymbol{x}$ ($\boldsymbol{y} \succeq \boldsymbol{x}$) if $\boldsymbol{f}(\boldsymbol{y}) \geq \boldsymbol{f}(\boldsymbol{x})$. These dominance relations can be extended to approximation sets – sets whose elements do not weakly dominate each other. For example, set A weakly dominates set B ($A \succeq B$) if every element of B is weakly dominated by at least one element of A.

The MNK-landscape is an extension of Kauffman's NK-landscape models of epistatic interaction [29)] to multi-objective combinatorial optimization problems. Formally, the MNK-landscape is defined as a vector function mapping binary strings of length N into M real numbers $\boldsymbol{f} = (f_1, f_2, \ldots, f_M) : \mathtt{Z}^N \rightarrow \mathtt{R}^M$, where $\mathtt{Z} = \{0, 1\}$. $\boldsymbol{K} = \{K_1, K_2, \ldots, K_M\}$ is a set of integers where each K_i gives the number of bits in the string that interact with each bit in the ith landscape. Each

⋆1 Throughout the paper, we assume maximization of the objective functions.

⋆2 The relation $\boldsymbol{f}(\boldsymbol{y}) > \boldsymbol{f}(\boldsymbol{x})$ means that $f_i(\boldsymbol{y}) > f_i(\boldsymbol{x}) \quad \forall i = 1, 2, \ldots, M$. Likewise, $\boldsymbol{f}(\boldsymbol{y}) \geq \boldsymbol{f}(\boldsymbol{x})$ implies $f_i(\boldsymbol{y}) \geq f_i(\boldsymbol{x}) \quad \forall i = 1, 2, \ldots, M$.

$f_i(\cdot)$ is given by the average of N functions by

$$f_i(\boldsymbol{x}) = \frac{1}{N}\sum_{j=1}^{N} f_{i,j}(x_j, z_1^{(i,j)}, z_2^{(i,j)}, \ldots, z_{K_i}^{(i,j)}) \quad (2)$$

where $f_{i,j} : \mathtt{Z}^{K_i+1} \rightarrow \mathtt{R}$ gives the fitness contribution of x_j to $f_i(\cdot)$, and $z_1^{(i,j)}, z_2^{(i,j)}, \ldots, z_{K_i}^{(i,j)}$ are the K_i bits interacting with x_j in string $\boldsymbol{x}$.

3. General Concepts of Path Relinking

Path relinking (PR) is a population-based search technique that generates a sequence of solutions in the decision space by exploring the trajectories that connect two high quality solutions called the *initiating solution* and *guiding solution*. Starting from the initiating solution, it creates new solutions by performing moves in the decision space that progressively incorporate the attributes (e.g., edges, nodes and sequence positions) of the guiding solution [30] that are not present in the initiating solution. Since there are usually several paths that can be formed between any two solutions, a selection strategy that guide which path to realize is necessary.

In general, PR requires the following: (i) neighborhood structure for the moves, (ii) solution attribute, (iii) a metric that measures the difference in attributes between two solutions, (iv) selection criteria for initiating and guiding solutions, and (v) selection criteria for the path. The neighborhood structure, solution attribute and its metric are usually problem dependent while the initiating and guiding solutions are characterized as high quality solutions. For the multi-objective case, these solutions are usually drawn from the set of potentially efficient solutions [21)–24),27),31)–33)]. In Ref. 27), it considered solutions generated by a multi-start approach that are not weakly dominated by a reference point as initiating and guiding solutions. For the path selection strategy, one may use scalarizing function [23)–25),33)] or random selection from only among efficient or promising candidate solutions [22)].

In most applications of PR to multi-objective case, the candidate solutions that are considered efficient with respect to some heuristic bounds are allowed to undergo local search procedure to generate better solutions [20)–23)]. Moreover, some apply PR on many pairs of solutions in the potentially efficient set before updating this set with the newly found nondominated solutions.

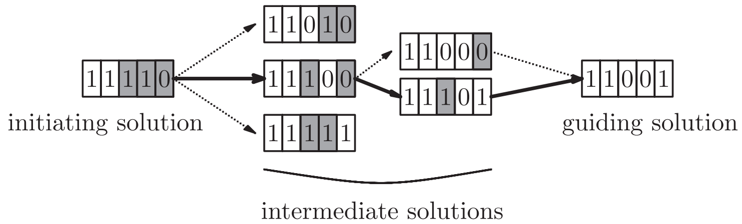


Fig. 1 Path relinking *links* the initiating solution (11110) and the guiding solution (11001) by performing moves on the solution 11110 that progressively generate a *path* of intermediate solutions that are more similar to the solution 11001. The shaded cells represent the difference between the solutions and the solution 11001 while the solid line represents the path that is chosen by a path selection strategy.

Figure 1 provides an illustration for the path relinking. In this figure, we assume that the solutions are represented as binary strings and the attributes associated to them are their bit-values. Thus, the shaded cells are the positions where the initiating and guiding solutions differ in their bit-values. Using a 1-bit flip operator, the neighborhood solutions of the initiating solution (11110) that have more similar attributes as the guiding solution (11001), i.e., solutions that reduce the Hamming distance between the two solutions, are visited and further explored. Since there may be several of these 1-bit neighbors, a path selection strategy chooses one of them. The selected solution, which is 11100 in the example, then becomes the current initiating solution and the process continues. The selected solutions, also referred to as *intermediate* solutions, form a *path* from the initiating to guiding solution which is the solid line in Fig. 1.

At this point, it is worthy to mention that the crossover operator called *deterministic multi-step crossover fusion* (dMSXF) [34)] has resemblance to PR. Like PR, dMSXF is carried-out between two parent solutions p_1 and p_2 and performs the search by progressively tracking the neighborhood of p_1 in the direction of p_2. Although dMSXF also performs moves on p_1 that creates solutions more similar to p_2, it is different from PR since it allows moves that simultaneously introduce two or more attributes of p_2. However, it is possible to mimic the PR by further restricting the moves associated to dMSXF.

4. Path Relinking for Many-objective Optimization

4.1 General Flow of the Path Relinking

Like any other implementation of path relinking in multi-objective case, we perform path relinking between two solutions that belong to the set $\mathcal{P}$ of potentially efficient solutions. However, since we deal primarily with many-objective problems where the size of $\mathcal{P}$ could be large, we propose selecting the initiating and guiding solutions only from the extreme solutions. Moreover, we use several forms of scalarizing functions to select intermediate solutions to form the path. Whereas all applications perform local search procedures within path relinking to intensify the search towards the optimal Pareto front, we do not implement any such procedure in order to clearly reveal the behavior of pure path relinking algorithm. **Figure 2** provides the algorithmic framework for the path relinking used in this study.

The proposed path relinking algorithm starts by creating a randomly generated

1: $\mathcal{P} \leftarrow$`Generate()`;
2: **repeat** {/*iteration loop*/}
3: $\mathcal{I} \leftarrow$`Define(`$\mathcal{P}$`)`;
4: $\mathcal{S} \leftarrow \{\}$;
5: **for all** $(\boldsymbol{i_s}, \boldsymbol{g_s}) \in \mathcal{I}$ **do**
6: **repeat** {/*Path Generation*/}
7: $F \leftarrow$`PathRelink(`$\boldsymbol{i_s}, \boldsymbol{g_s}$`)`;
8: $\boldsymbol{x} \leftarrow$`PathSelect(`$F$,$\omega$`)`;
9: $\mathcal{S} \leftarrow \mathcal{S} \cup \{\boldsymbol{x}\}$;
10: $\boldsymbol{i_s} \leftarrow \boldsymbol{x}$;
11: **until** $\gamma(\boldsymbol{i_s}, \boldsymbol{g_s}) < d_0$
12: **end for**
13: $\mathcal{P} \leftarrow$ Nondominated$(\mathcal{P} \cup \mathcal{S})$;
14: $\mathcal{P} \leftarrow$`Select(`$\mathcal{P}, l$`)`;
15: **until** termination condition is satisfied
16: **return** nondominated set $\mathcal{P}$

Fig. 2 Path relinking algorithm.

distinct extreme solutions that initially forms the set $\mathcal{P}$ of potentially efficient solutions via the `Generate` procedure. In every iteration, the procedure `Define` then initializes the set $\mathcal{I}$ of pairs of initiating ($\boldsymbol{i_s}$) and guiding ($\boldsymbol{g_s}$) solutions from the set $\mathcal{P}' \subset \mathcal{P}$ that only contains all the extreme solutions. It forms the $\boldsymbol{i_s}$-$\boldsymbol{g_s}$ pairs via two proposed methods discussed in Section 4.2.

For each element $(\boldsymbol{i_s}, \boldsymbol{g_s})$ of $\mathcal{I}$, two procedures– `PathRelink` and `PathSelect`, iteratively generate the set $\mathcal{S}$ of intermediate solutions until $\boldsymbol{i_s}$ have near "similar" attributes as $\boldsymbol{g_s}$ (Lines 6-11). Thus, if γ is a metric that measures the difference in attributes between $\boldsymbol{i_s}$ and $\boldsymbol{g_s}$, then the generation of intermediate solutions between the current $\boldsymbol{i_s}$-$\boldsymbol{g_s}$ pair terminates when the γ value between the two solutions reaches a threshold given by d_0. In many cases, d_0 is set to 1 which indicate that $\boldsymbol{i_s}$ is now one move closer to $\boldsymbol{g_s}$.

When all the $\boldsymbol{i_s}$-$\boldsymbol{g_s}$ pairs are done generating their intermediate solutions, the set $\mathcal{P}$ is updated by removing its solutions that are dominated by $\mathcal{S}$ and taking all solutions in $\mathcal{S}$ that are not weakly dominated by any other solutions in $\mathcal{P}$ and $\mathcal{S}$ (Line 13).

Since the number of nondominated solutions in every Pareto front increases dramatically with M [5)], it is important to have an archiving strategy that controls the size (say, less than l) of the set $\mathcal{P}$ (Line 14). The method `Select` performs the archiving by selecting the M extreme solutions of $\mathcal{P}$ and randomly selecting solutions from the remaining $l - M$ solutions when $\mathcal{P} > l$. Otherwise, `Select` chooses all solutions of $\mathcal{P}$. The value of l is set to 100.

4.2 Initiating and Guiding Solutions

In defining the $\boldsymbol{i_s}$-$\boldsymbol{g_s}$ pairs, we propose two methods called *Cycle* and *Pair*. Since the $\boldsymbol{i_s}$-$\boldsymbol{g_s}$ pairs are to be drawn from the set $\mathcal{P}'$ of extreme solutions, the two methods try to generate good extremes solutions and uniformly cover the Pareto front. However, they differ in the way they sample the solutions. *Pair* samples the extreme solutions as either $\boldsymbol{i_s}$ or $\boldsymbol{g_s}$ in random manner while *Cycle* uses all the extreme solutions both as $\boldsymbol{i_s}$ and $\boldsymbol{g_s}$ according to some definite rule. Thus, the stochastic nature of the *Pair* allows for a more aggressive search between and around the regions of some of the pair of extreme solutions while *Cycle* balances the search for better solutions between and around all pairs of extreme solutions.

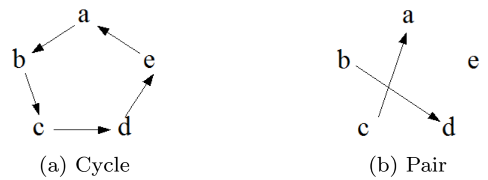


Fig. 3 Concept diagram of selecting initiating and guiding solution via (a) *Cycle* and (b) *Pair*. The head of the arrow points to the guiding solution and the tail corresponds to the initiating solution.

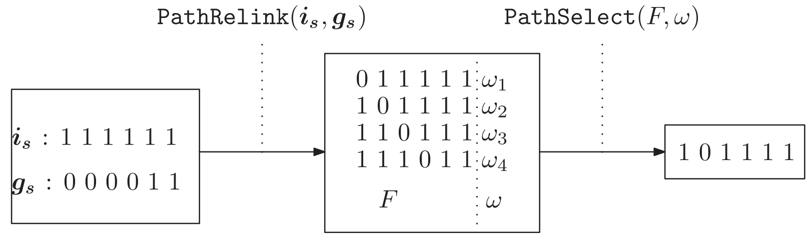


Fig. 4 Path relinking and selection. `PathRelink`() generates several solutions and `PathSelects`() chooses one solution to be an intermediate solution of the path.

Pair is loosely related to randomly selecting the $\boldsymbol{i_s}$-$\boldsymbol{g_s}$ pairs from the set $\mathcal{P}$ of efficient solutions in the bi-objective case (e.g., Refs. 21), 31)). In every iteration, *Pair* forms a set of distinct pairs of $\boldsymbol{i_s}$ and $\boldsymbol{g_s}$ by iteratively drawing two distinct solutions from $\mathcal{P}'$. In every draw, it assures that the two distinct solutions are different from the previous draws. This process of iteratively drawing two distinct solutions continues until it is no longer possible to obtain different pair of solutions from $\mathcal{P}'$. It follows that *Pair* leaves one solution unmatched if $|\mathcal{P}'|$ is odd. The total number of pairs formed is $\lfloor |\mathcal{P}'|/2 \rfloor$.

Cycle arranges the solutions of $\mathcal{P}'$ in random order. Then, the first and second solutions are labeled as $\boldsymbol{i_s}$ and $\boldsymbol{g_s}$, respectively. For the succeeding pairs of solutions, the initiating solution is the guiding solution of the previous pair and the guiding solution is the next solution in $\mathcal{P}'$. The final $\boldsymbol{i_s}$-$\boldsymbol{g_s}$ pair are the last and first solutions in $\mathcal{P}'$, respectively. The total number of pairs formed is $|\mathcal{P}'|$. Thus, *Cycle* performs more path relinking processes than *Pair*. **Figure 3** illustrates the two methods when $M = 5$.

4.3 Path Generation and Selection

The actual generation of the sequence of solutions or path from solution $\boldsymbol{i_s}$ to $\boldsymbol{g_s}$ consists of two procedures. The first procedure `PathRelink` $(\boldsymbol{i_s}, \boldsymbol{g_s})$ returns at each step the set F of neighborhood solutions of $\boldsymbol{i_s}$ that reduces the Hamming distance γ from $\boldsymbol{g_s}$, i.e., $F = \{\boldsymbol{x} \in \mathcal{N}(\boldsymbol{i_s}) : \gamma(\boldsymbol{x}, \boldsymbol{g_s}) < \gamma(\boldsymbol{i_s}, \boldsymbol{g_s})\}$ where $\mathcal{N}$ is defined as the 1-bit neighborhood. Since each call of `PathRelink` may return many solutions i.e., $|F| \geq 1$, then several paths maybe formed. Hence, a path selection mechanism provided by `PathSelect` is used to choose the preferred path. `PathSelect` immediately chooses a single solution from F based on the acceptance criteria given by the real-valued function ω expressed as the weighted sum fitness function by

$$\omega(\boldsymbol{x}) = \boldsymbol{w} \cdot \boldsymbol{f}(\boldsymbol{x}) \tag{3}$$

where $\boldsymbol{w}=(w_1, w_2, \ldots, w_M)$ is a weight vector such that $\sum_{j=1}^{M} w_j = 1$ and $w_j \geq 0\ \forall j$. The solution having the best value of ω is accepted. Note that the use of fitness function is a common strategy in solving MO problems [35].

We define the initial setting of ω to be the objective function where the guiding solution is best. Recall that the guiding solution is chosen from among the extreme solutions found so far by the algorithm. Thus, if the guiding solution is the best solution in the objective function f_i, then we accept the solution having the best value of $\omega(\boldsymbol{x}) = f_i(\boldsymbol{x})$, i.e., $\boldsymbol{w} = (0, 0, \ldots, 1, \ldots, 0)$, or $w_i = 1$ and $w_{j \neq i} = 0$. This strategy clearly prefers moves that are attractive relative to f_i. Moreover, it limits the search from the many objective standpoint to single objective optimization. This is also a natural and novel way of extending the implementation of path selection in single-objective optimization problems [30] to MaO problems. The selected solution in `PathSelect` then becomes an intermediate solution of the path and the next initiating solution for `PathRelink`. These two procedures are illustrated in **Fig. 4**.

It is important to note that `PathRelink` and `PathSelect` have been expressed as a local search that optimizes a lexicographic objective function $\Phi = (\gamma, \omega)$ [25]. Since γ is the Hamming distance and ω is the fitness function, then Φ is clearly composed of functions defined in both the decision and objective spaces.

4.4 Link Direction

In this study, we also consider *re-initiating* the process of path generation in the

Table 1 Four variants of the path relinking algorithm. The symbol $\boldsymbol{i_s} \leftrightarrow \boldsymbol{g_s}$ ($\boldsymbol{i_s} \rightarrow \boldsymbol{g_s}$) indicates that there is (no) reversal of roles between initiating and guiding solutions.

Cycle	$\boldsymbol{i_s} \rightarrow \boldsymbol{g_s}$	PRCycle1
	$\boldsymbol{i_s} \leftrightarrow \boldsymbol{g_s}$	PRCycle2
Pair	$\boldsymbol{i_s} \rightarrow \boldsymbol{g_s}$	PRPair1
	$\boldsymbol{i_s} \leftrightarrow \boldsymbol{g_s}$	PRPair2

opposite direction by interchanging the roles of initiating and guiding solutions. Thus, once the termination criterion in Line 11 of Fig. 2 is satisfied, the roles of initiating and guiding solutions are reversed. Consequently, two paths are formed, thus increasing the number of intermediate solutions between the initiating and guiding solutions. Moreover, these intermediate solutions are now the results of the moves that are attractive relative to the objective function where either the initiating or guiding solution is best. Therefore, this strategy aims to generate better extreme solutions. This technique has been proposed in single objective optimization [30] but it has been implemented only in the bi-objective case in Ref. 27).

Combining the two strategies for forming the sets of initiating solutions ($\boldsymbol{i_s}$) and guiding solutions ($\boldsymbol{g_s}$), and whether to interchange the roles of $\boldsymbol{i_s}$ and $\boldsymbol{g_s}$ initially give us four variants of the PR algorithm. **Table 1** summarizes these variants.

5. Performance Measures

Three different performance metrics are used to evaluate the performances of the different optimizers. These metrics are the hypervolume, coverage, and sum of maximum objective values. They provide information about the different aspects of the quality of approximation sets.

5.1 Hypervolume.

The hypervolume $\mathcal{H}$ is a unary indicator that measures the volume of the objective space that is weakly dominated by an approximation set [36]. It is considered Pareto-compliant i.e., whenever an approximation set A is better than approximation set B then the hypervolume of A is greater than B [37]. It can be expressed as

$$\mathcal{H}(A) = \cup_{i=1}^{|A|}(\mathcal{V}_i - \cap_{j=1}^{i-1}\mathcal{V}_i\mathcal{V}_j) \tag{4}$$

where $\mathcal{V}_i$ is the hypervolume rendered by the point $\boldsymbol{x}_i \in A$ and a user-defined reference point.

The hypervolume indicator is dependent on a reference point – any point dominated by the approximation sets. For example, the contribution of the extreme points to hypervolume is magnified when the reference point is far away from the Pareto front. On the other hand, the central points on the Pareto front are given more weight to the hypervolume if the reference point is near the Pareto front. In this case, the hypervolume describes the convergence property or the closeness of approximation sets towards the Pareto optimal solutions in the objective space. In order to give a meaningful interpretation of the hypervolume, we use different reference points $\mathcal{O}$ defined by the parameter $\alpha \in [0, 1]$. If α equals zero then $\mathcal{O}$ is the origin $\boldsymbol{O} = (0, 0, \ldots, 0)$ and far away from the Pareto front. On the other hand, as the value of α nears 1, $\mathcal{O}$ approaches the Pareto front, i.e., the point W having as coordinates the worst objective values of the solutions found. If $\alpha = 0.5$, then $\mathcal{O}$ is the midpoint of the segment $\overline{OW}$. To calculate $\mathcal{H}$, we use the algorithm presented in Ref. 38).

5.2 Set Coverage Metric.

The coverage $\mathcal{C}(A, B)$ metric is a binary quality indicator that gives the proportion of approximation set B that is weakly dominated by approximation set A [39]. This metric provides complementary information on convergence and is given by

$$\mathcal{C}(A, B) = \frac{|\{\boldsymbol{b} \in B : \exists \boldsymbol{a} \in A \text{ where } \boldsymbol{a} \succeq \boldsymbol{b}\}|}{|B|} \ . \tag{5}$$

If A weakly dominates all members of B then the metric value $\mathcal{C}(A, B)$ is equal to 1. On the other hand, if no member of B is weakly dominated by A, then $\mathcal{C}(A, B)$ is zero.

5.3 Sum of Maximum Objective Values.

The sum of maximum objective values ($\mathcal{S}_{\max}$) measures the quality of the extreme solutions and solutions around the M edges of the Pareto front [7]. Therefore, this metric provides information about the performance of an algorithm in terms of diversity and extent or the spread of its extreme solutions. Like the $\mathcal{H}$

metric, $\mathcal{S}_{\max}$ requires scaling and normalization in order to allow the different objectives to contribute equally to its value. It is given by the formula

$$\mathcal{S}_{\max}(A) = \sum_{i=1}^{M} \max_{\boldsymbol{x} \in A} f_i(\boldsymbol{x}) \ . \quad (6)$$

6. Experimental Results and Analysis

6.1 Performance Varying the Number of Objectives

In this section, we show the performances of the PR variants presented in Table 1 which uses the initial setting of $\boldsymbol{w}$ in Section 4.3, under different numbers of objectives M and a fixed value $K = 7$. It is known that higher number of objectives translates to fewer but denser Pareto fronts [5]. **Figure 5** (a)∼(d) provides the boxplots of the normalized hypervolume values or the average ratios $\mathcal{H}$(PR)/$\mathcal{H}$(NSGA-II) for the PR variants. These figures show that NSGA-II outperforms the PR variants when $2 \leq M \leq 4$. Moreover, there is a dramatic decrease in the hypervolume values when α is 0.99 and $2 \leq M \leq 3$, indicating poor convergence in the part of PR. This is supported by the $\mathcal{C}$ metric in Fig. 5 (e) where it shows that NSGA-II weakly dominates all solutions of PR when $M = 2$ and covers most of the solutions when $M = 3$. The extreme solutions of NSGA-II are also better compared to PR (see Fig. 5 (f)). These results are expected since several studies have shown the effectiveness of NSGA-II in solving multi-objective optimization problems. In addition, it is possible for PR to be trapped early to locally optimal solutions at the extremes, thus preventing it to find better solutions.

As M increases from 4 to 10, the performance of PR variants in terms of convergence improves as shown by the improvements in $\mathcal{H}$ and $\mathcal{C}$ metrics. For example when $M \geq 6$, at least 75% of the runs of PR show improvement in $\mathcal{H}$ (i.e., normalized $\mathcal{H} > 1$) regardless of the values of α. Likewise, although they weakly dominate few solutions of NSGA-II, almost none of their solutions are covered by NSGA-II. On the other hand, the big difference in the performance in $\mathcal{H}$ between $\alpha = 0.5$ and $\alpha = 0.99$ can be attributed to the better convergence in the central region.

Among the different PR variants, it can be seen that the re-initiating strategy or interchanging the roles of the $\boldsymbol{i_s}$ and $\boldsymbol{g_s}$ is indeed beneficial in terms of improving the extreme solutions but only for $M \leq 5$. Figure 5 (f) shows that for $M \leq 5$, the median of the normalized $\mathcal{S}_{\max}$ values of PRCycle2 and PRPair2 are significantly better than that of PRCycle1 and PRPair1. Likewise, the normalized hypervolume when α equals 0 is higher in PRCycle2 and PRPair2 than in PRCycle1 and PRPair1. For higher values of M, path relinking may require other settings that may improve the quality of its extreme solutions.

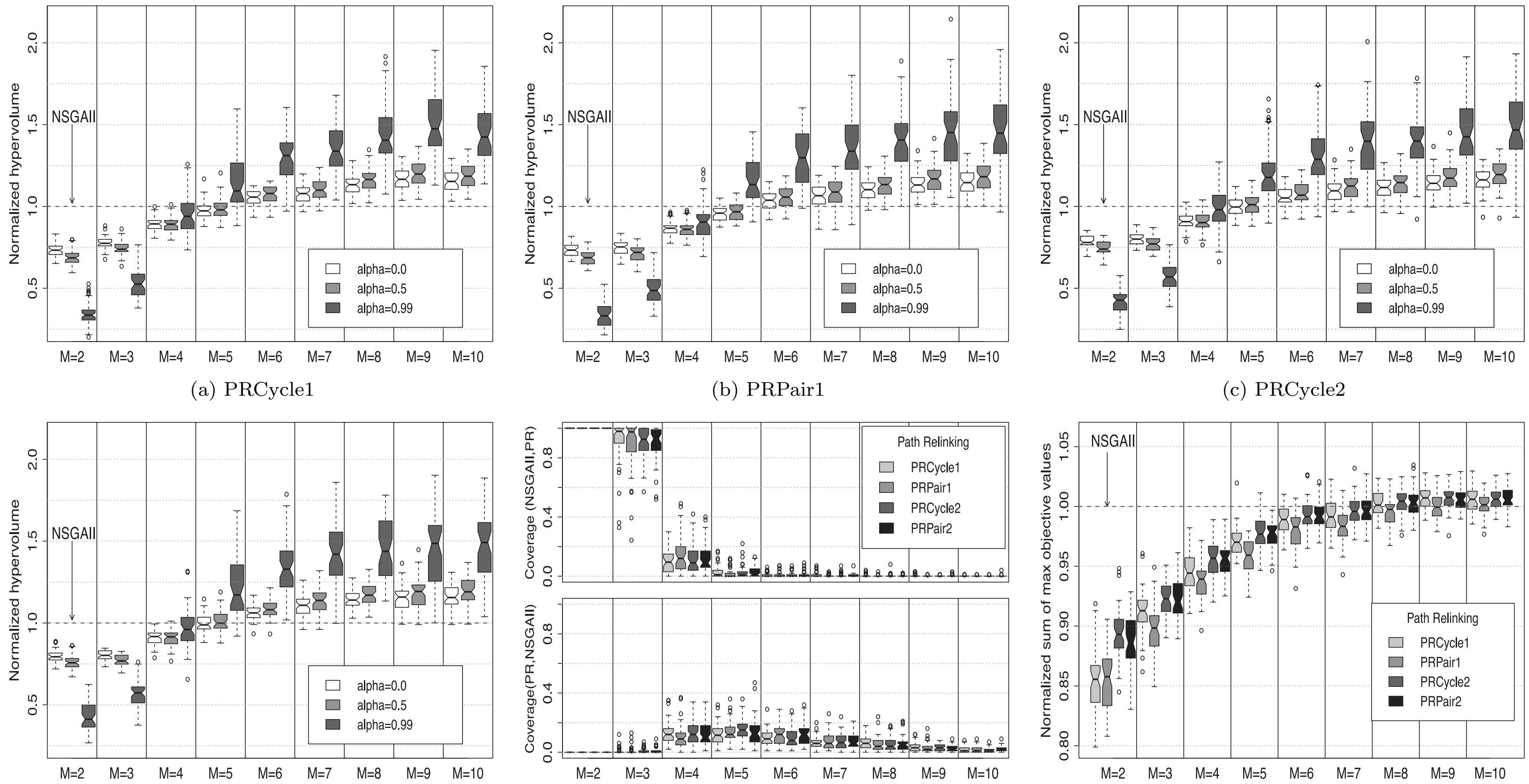
Between *Cycle* and *Pair* strategies, the former shows significant edge over the latter in terms of the $\mathcal{S}_{\max}$ metric but only between PRCycle1 and PRPair1. This edge is insignificant when the re-initiating strategy is implemented. This suggests that reversing the roles of $\boldsymbol{i_s}$ and $\boldsymbol{g_s}$ is more effective in improving the quality of the extreme solutions for the path relinking.

In terms of convergence, there is no strong indication that one variant is better than the others. This suggests that the manner of defining $\boldsymbol{i_s}$ and $\boldsymbol{g_s}$ from the set of extreme solutions and the re-initiating strategy do not have strong influence in the convergence property of the path relinking.

6.2 Performance Varying the Path Selection Strategy

In this study, we analyze also the effects of varying the direction of the search path given by the weight vector $\boldsymbol{w}$. Since the re-initiating strategy provides the better alternatives for PR and *Cycle* gives a balanced search for better solutions as defined in Section 4.2, we choose PRCycle2 as the algorithmic framework. Thus, the resulting new PR variants have the same configurations as PRCycle2 except in their definition of the vector $\boldsymbol{w}$. **Table 2** summarizes the new selection strategies based on the corresponding fitness function ω. In the following, we explain these strategies using **Fig. 6** as illustration. In this figure, an intermediate solution is selected from among the 1-bit neighbors $\boldsymbol{x_1}$, $\boldsymbol{x_2}$, and $\boldsymbol{x_3}$ of $\boldsymbol{i_s}$ depending on the weight vector $\boldsymbol{w}$.

PRCycle2_w1.0. First, we define $\boldsymbol{w}$ as a unit vector such that $w_i = 1$ if and only if $\boldsymbol{i_s}$ is best in objective function f_i. Thus, this strategy prefers moves that are attractive in the direction of $\boldsymbol{i_s}$ in the objective space. It also prefers the search for extreme solutions by considering a single objective function each time. This strategy has been proposed only in the single objective problem [30]. In Fig. 6, its weight vector is given by $\boldsymbol{w_a}$ since $\boldsymbol{i_s}$ is best in f_2 objective. Thus, it selects $\boldsymbol{x_1}$ since this solution provided the largest fitness function value for the



(a) PRCycle1 (b) PRPair1 (c) PRCycle2

(e) PRPair2 (f) $\mathcal{C}$ metric (g) $\mathcal{S}_{\max}$ metric

Fig. 5 (a)–(d) Normalized $\mathcal{H}$ metric (e) $\mathcal{C}$ metric (f) normalized $\mathcal{S}_{\max}$ metric between PR variants and NSGA-II for different M values and $K = 7$.

given weight vector $\boldsymbol{w_a}$.

PRCycle2_w0.5. Another method is to set w_i to 0.5 if and only if either $\boldsymbol{i_s}$ or $\boldsymbol{g_s}$ is best in the objective function f_i, and zero otherwise. This strategy is biased towards the central portion of the particular subspace defined by the objective functions where initiating and guiding solutions are best. Clearly, this strategy performs the search by using two objective functions each time. In Fig. 6, its weight vector is given by $\boldsymbol{w_c}$ and selects the intermediate solution $\boldsymbol{x_2}$.

PRCycle2_aggr. The final strategy uses a scalarizing function that aggregates all the objective functions using Eq. (3) without any reference to the initiating and guiding solutions i.e., it is not strictly attracted in the direction to any of the objective functions. The values of the weights are randomly changed for every call of `PathSelect` (see Line 8 of the PR algorithm in Fig. 2). This strategy

Table 2 Selection strategies for the path relinking algorithm using PRCycle2 variant.

Fitness Function	Path relinking
$\omega(\boldsymbol{x}) = f_i(\boldsymbol{x})$	PRCycle2_w1.0
$\omega(\boldsymbol{x}) = 0.5f_i(\boldsymbol{x}) + 0.5f_j(\boldsymbol{x})$	PRCycle2_w0.5
$\omega(\boldsymbol{x}) = \sum_{i=1}^{M} w_i f_i(\boldsymbol{x})$	PRCycle2_aggr

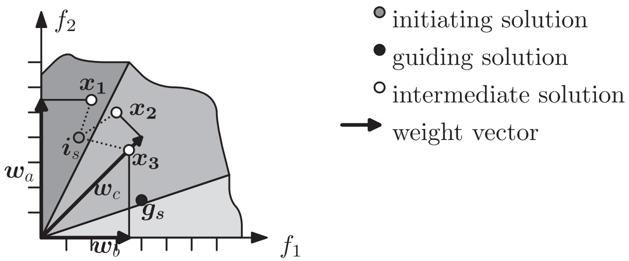


Fig. 6 An intermediate solution is selected depending on the selection strategy given by the weight vector.

is commonly used in the bi-objective case [21)–25)]. In Fig. 6, it selects $\boldsymbol{x_2}$ if the generated weight vector lies in the gray area (middle) while $\boldsymbol{x_1}$ (resp. $\boldsymbol{x_3}$) is chosen if the weight vector falls in the dark gray (resp. light gray) area.

Note that the initial setting of $\boldsymbol{w}$ as used by original PRCycle2 and analyzed in Section 6.1 is given by $\boldsymbol{w_b}$ since $\boldsymbol{g_s}$ is best in f_1 objective thereby selecting the intermediate solution $\boldsymbol{x_3}$.

It can be seen in the boxplots in **Fig. 7** (a)∼(d) that the normalized $\mathcal{H}$ values for the different selection strategies improve as M increases. Remarkably, PRCycle2_aggr posted the biggest improvement. For example when $\alpha = 0.99$, the median $\mathcal{H}$ value of PRCycle2_aggr is at least 100% greater than that of the NSGA-II when $M = 6$ and 250% greater when $M = 10$. This increase in $\mathcal{H}$ value translates to higher number of solutions of NSGA-II being weakly dominated by PRCycle2_aggr. Figure 7 (e) shows that roughly between 40% to 60% of the solutions of NSGA-II are covered by PRCycle2_aggr when $M \geq 6$, while NSGA-II covers nothing of PRCycle2_aggr.

The good convergence of PRCycle2_aggr as shown by its performance in $\mathcal{H}$ and $\mathcal{C}$ metrics expectedly sacrifices the quality of its extreme solutions and its solutions around the edges since the normalized $\mathcal{S}_{\max}$ metric (see Fig. 7 (f)) shows that PRCycle2_aggr is totally outperformed by NSGA-II. It is PRCycle2 and PRCycle2_w1.0 that perform well in terms of $\mathcal{S}_{\max}$ with the latter obtaining the best extreme values. PRCycle_w1.0 even outperforms NSGA-II when $M \geq 6$. All these results suggest that the manner of selecting the intermediate solutions or creating the path is a valuable factor when implementing PR. The different selection strategies exhibit a trade-off between convergence and spread.

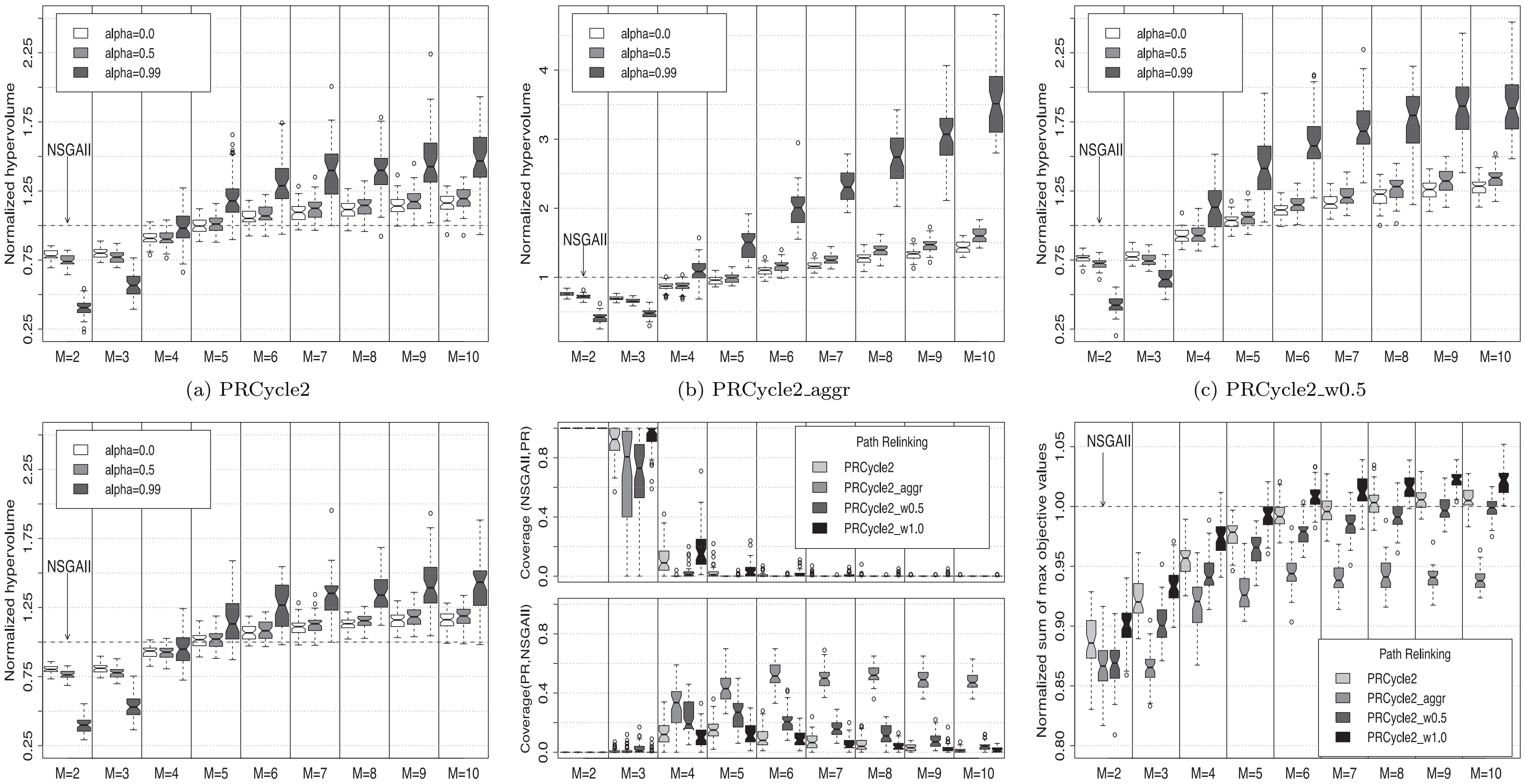
6.3 Performance Varying the Degree of Epistatic Interactions

It was demonstrated in Ref. 5) that the number of solutions in the top Pareto fronts reduces with K for all values of M. Moreover, the optimal solutions and their true Pareto fronts become more "discontiguous", and more non-convex regions appear in the fronts with K. To study the behavior of PR under different number of epistatic interactions, we consider the performances of PRCycle2 and PRCycle2_aggr when K ranges from 0 to 50.

Figure 8 shows the boxplots of the normalized $\mathcal{H}$ given different values of M and K, and $\alpha = 0.99$. It can be observed that for all values of K, the $\mathcal{H}$ values of PR are better than NSGA-II only when M is high. However, increasing the level of epistatic interaction diminishes the edge of PR over NSGA-II. For example when $K = 0$ and $M = 10$, the average normalized $\mathcal{H}$ values of PRCycle2 and PRCycle2_aggr are 1.4627 and 2.9325, respectively. When K increases to 50, the averages fall to 1.0531 and 1.8041. These results also indicate that PRCycle2_aggr still performs better than PRCycle2 in terms of the $\mathcal{H}$ metric.

Figure 9 (a) shows that NSGA-II covers almost all the solutions of PR for all K and $M = 2$. But, PRCycle2 and PRCycle2_aggr weakly dominated more solutions of NSGA-II than NSGA-II can cover them when $M > 4$. PRCycle2_aggr also has higher coverage of NSGA-II compared to PRCycle2. However, the coverage of PRCycle2 and PRCycle2_aggr over NSGA-II decreases as K increases.

In terms of the quality of solutions at the extremes, PRCycle2 still performs better than PRCycle2_aggr but both don't find extreme solutions that are good as NSGA-II as demonstrated by the normalized $\mathcal{S}_{\max}$ in Fig. 9 (b). However, as K increases, there is an improving trend for the normalized $\mathcal{S}_{\max}$ of PRCycle2_aggr.



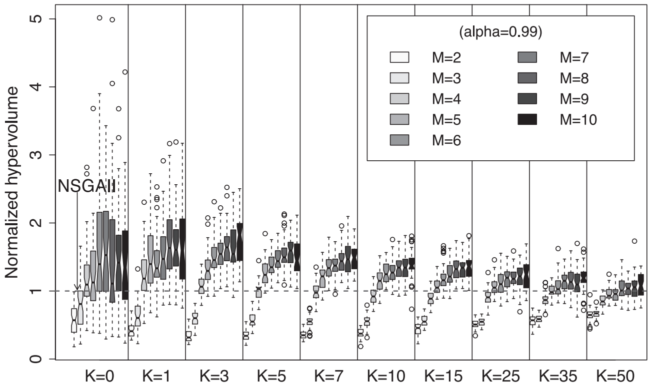
(a) PRCycle2 (b) PRCycle2_aggr (c) PRCycle2_w0.5

(e) PRCycle2_w1.0 (f) $\mathcal{C}$ metric (g) $\mathcal{S}_{\max}$ metric

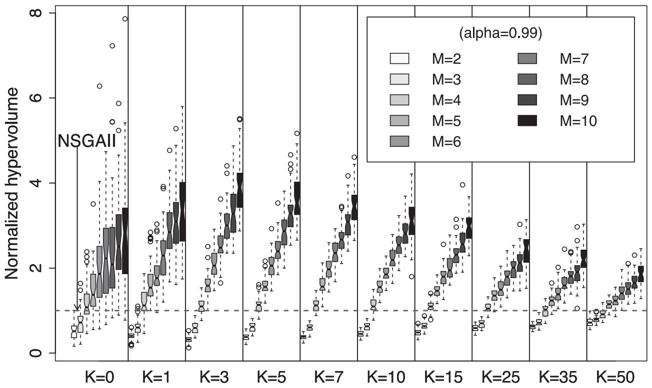
Fig. 7 (a)–(d) Normalized $\mathcal{H}$ metric (e) $\mathcal{C}$ metric (f) normalized $\mathcal{S}_{\max}$ metric between PR variants and NSGA-II for different M values and $K = 7$.

7. Conclusions

In this paper, we have studied the behavior of path relinking (PR) algorithm on MNK-landscape models having different number of objective functions and degree of epistatic interactions. We have designed several variants of PR based on [a] selection criteria for initiating ($\boldsymbol{i_s}$) and guiding ($\boldsymbol{g_s}$) solutions, [b] link direction, and [c] selection strategy for the path. Experiments have shown that the selection of pairs of $\boldsymbol{i_s}$ and $\boldsymbol{g_s}$, and link direction have stronger influence in the quality of the extreme solutions than in the convergence property of PR when $M \leq 5$. In particular, the results suggest that applying *Cycle*, which uses all extreme solutions both as $\boldsymbol{i_s}$ and $\boldsymbol{g_s}$, as selection criteria and a link direction that applies re-initiating strategy are beneficial for PR.

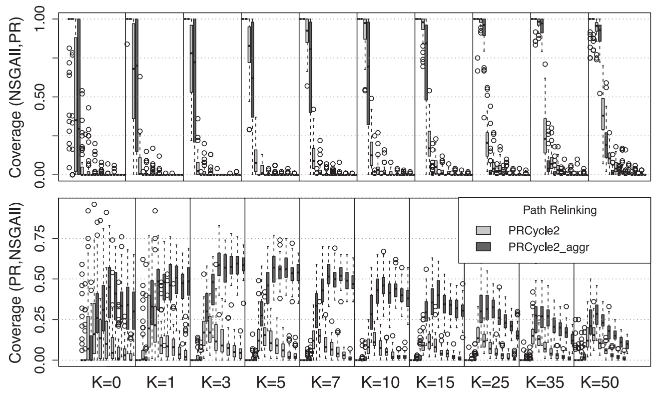


(a) $\mathcal{H}$: PRCycle2

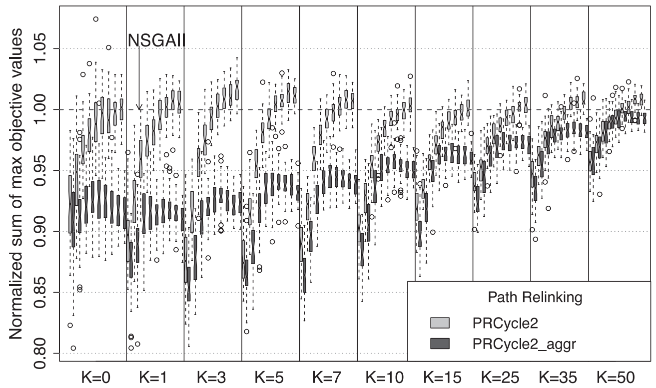


(b) $\mathcal{H}$: PRCycle2_aggr

Fig. 8 Normalized $\mathcal{H}$ metric between NSGA-II and (a) PRCycle2 (b) PRCycle2_aggr for different values of M and K.



(a) $\mathcal{C}$ metric



(b) $\mathcal{S}_{\max}$ metric

Fig. 9 (a) $\mathcal{C}$ metric and (b) normalized $\mathcal{S}_{\max}$ metric of PR variants and NSGA-II for different values of K and M.

For higher values of M, PR can be directed to find either better extreme solutions or better convergence depending on the selection strategy for the path. For example, if we use selection strategy for the path that favors solutions in the direction of $\boldsymbol{i_s}$, then better extreme solutions can be found. In fact, PR has better extreme solutions than NSGA-II when $M \geq 6$ ($K = 7$). On the other hand, if we use a selection strategy for the path that aggregates all objective functions, then solutions with better convergence can be obtained. In fact, PR has better convergence behavior compared to NSGA-II when $M \geq 4$. This good convergence can be seen in a broad range of levels of epistatic interaction K, with its peak improvement around $1 \leq K \leq 10$.

In the future, we want to investigate adaptive strategies that simultaneously improve both convergence and spread. For example, we are interested in studying the efficiency of PR when it dynamically uses the different path selection strategies. We also want to study the ways on how to enhance MOEAs using PR. One possible approach is to strategically embed a PR procedure inside MOEAs. For instance, we have shown that NSGA-II performs poorly in terms of convergence and thus, the presence of PR may enhance NSGA-II's convergence. Depending on the selection strategy, PR may also be used to further improve the quality of the extreme solutions of NSGA-II. When implementing these examples, we want to measure the contribution of PR to the overall performance of NSGA-II to provide additional information about its efficiency and relevance.

References

1) Aguirre, H.E. and Tanaka, K.: Insights on Properties of Multiobjective MNK-Landscapes, *IEEE Congress on Evolutionary Computation* (*CEC'2004*), Vol.1, pp.196–203 (2004).
2) Khare, V., Yao, X. and Deb, K.: Performance scaling of multi-objective evolutionary algorithms, *Evolutionary Multi-Criterion Optimization*, Lecture Notes in Computer Science, Vol.2632, pp.367–390 (2003).
3) Purshouse, R.C. and Fleming, P.: Conflict, harmony, and independence: Relationships in evolutionary multi-criterion optimisation, *Evolutionary Multi-Criterion Optimization*, Lecture Notes in Computer Science, Vol.2632, pp.16–30 (2003).
4) Aguirre, H. and Tanaka, K.: Selection, drift, recombination, and mutation in multiobjective evolutionary algorithms on scalable MNK-Landscapes, *Evolutionary Multi-Criterion Optimization*, Lecture Notes in Computer Science, Vol.3410, pp.355–369 (2005).
5) Aguirre, H.E. and Tanaka, K.: Working principles, behavior, and performance of MOEAs on MNK-landscapes, *European Journal of Operational Research*, Vol.181, No.3, pp.1670–1690 (2007).
6) Hughes, E.: Evolutionary many-objective optimization: Many once or one many?, *IEEE Congress on Evolutionary Computation*, pp.222–227 (2005).
7) Ishibuchi, H., Tsukamoto, N. and Nojima, Y.: Evolutionary many-objective optimization: A short review, *IEEE Congress on Evolutionary Computation*, pp.2424–2431 (2008).
8) Sato, H., Aguirre, H. and Tanaka, K.: Controlling dominance area of solutions and its impact on the performance of MOEAs, *Evolutionary Multi-Criterion Optimization*, Lecture Notes in Computer Science, Vol.4403, pp.5–20 (2007).
9) Laumanns, M., Thiele, L., Deb, K. and Zitzler, E.: Combining convergence and diversity in evolutionary multiobjective optimization, *Evolutionary Computation*, Vol.10, No.3, pp.263–282 (2002).
10) Ikeda, K., Kita, H. and Kobayashi, S.: Failure of Pareto-based MOEAs: Does non-dominated really mean near to optimal?, *IEEE Congress on Evolutionary Computation*, Vol.2, pp.957–962 (2001).
11) Deb, K. and Saxena, D.: Searching for Pareto-optimal solutions through dimensionality reduction for certain large-dimensional multi-objective optimization problems, *IEEE Congress on Evolutionary Computation*, pp.3353–3360 (2006).
12) Saxena, D. and Deb, K.: Non-linear dimensionality reduction procedures for certain large-dimensional multi-objective optimization problems: Employing correntropy and a novel maximum variance unfolding, *Evolutionary Multi-Criterion Optimization*, Lecture Notes in Computer Science, Vol.4403, pp.772–787 (2007).
13) Brockhoff, D. and Zitzler, E.: Are all objectives necessary? On dimensionality reduction in evolutionary multiobjective optimization, *Parallel Problem Solving from Nature*, Lecture Notes in Computer Science, Vol.4193, pp.533–542 (2006).
14) Deb, K. and Sundar, J.: Reference point based multi-objective optimization using evolutionary algorithms, *Proc. 8th Annual Conference on Genetic and Evolutionary Computation*, pp.635–642 (2006).
15) Thiele, L., Miettinen, K., Korhonen, P. and Molina, J.: A Preference-Based Evolutionary Algorithm for Multi-Objective Optimization, *Evolutionary Computation*, Vol.17, No.3, pp.411–436 (2009).
16) Ishibuchi, H. and Murata, T.: A multi-objective genetic local search algorithm and its application to flowshop scheduling, *IEEE Trans. Systems, Man, and Cybernetics*, Vol.28, No.3, pp.392–403.
17) Jaszkiewicz, A.: Genetic local search for multi-objective combinatorial optimization, *European Journal of Operational Research*, Vol.137, No.1, pp.50–71 (2002).
18) Aguirre, H. and Tanaka, K.: Many-objective optimization by space partitioning and adaptive ε-ranking on MNK-Landscapes, *Evolutionary Multi-Criterion Optimization*, Lecture Notes in Computer Science, Vol.5467, pp.407–422 (2009).
19) Glover, F.: Tabu search and adaptive memory programming advances, applications and challenges, *Interfaces in Computer Science and Operations Research*, Barr, R., Helgason, R. and Kennington, J. (Eds.), pp.1–75, Kluwer Academic Publishers (1996).
20) Gandibleux, X., Morita, H. and Katoh, H.: Evolutionary operators based on elite solutions for bi-objective combinatorial optimization, *Applications of Multi-Objective Evolutionary Algorithms*, Coello, C.C. and Lamont, G. (Eds.), Advances in Natural Computation, Vol.1, World Scientific, Chapter 23, pp.555–579 (2004).
21) Parragh, S., Doerner, K., Hartl, R. and Gandibleux, X.: A heuristic two-phase solution approach for the multi-objective dial-a-ride problem, *Networks*, Vol.54, No.4, pp.227–242 (2009).
22) Przybylski, A., Gandibleux, X. and Ehrgott, M.: Two phase algorithms for the bi-objective assignment problem, *European Journal of Operational Research*, Vol.185, No.2, pp.509–533 (2008).

23) Pasia, J., Gandibleux, X., Doerner, K. and Hartl, R.: Local search guided by path relinking and heuristic bounds, *Evolutionary Multi-Criterion Optimization*, Lecture Notes in Computer Science, Vol.4403, pp.501–515 (2007).
24) Basseur, M., Seynhaeve, F. and Talbi, E.: Path relinking in pareto multi-objective genetic algorithms, *Evolutionary Multi-criterion Optimization*, Lecture Notes in Computer Science, Vol.3410, pp.120–134 (2005).
25) Jaszkiewicz, A. and Zielniewicz, P.: Pareto memetic algorithm with path relinking for bi-objective traveling salesperson problem, *European Journal of Operational Research*, Vol.193, No.3, pp.885–890 (2009).
26) Schilde, M., Doerner, K. and Hartl, R.: Metaheuristics for the bi-objective orienteering problem, *Swarm Intelligence*, Vol.3, No.3, pp.179–2001 (2009).
27) Beausoleil, R.P., Baldoquin, G. and Montejo, R.: Multi-start and path relinking methods to deal with multiobjective knapsack problems, *Annals of Operations Research*, Vol.157, No.1, pp.105–133 (2008).
28) Deb, K., Agrawal, S., Pratap, A. and Meyarivan, T.: A Fast Elitist Non-dominated sorting genetic algorithm for multi-objective Optimisation: NSGA-II, *Parallel Problem Solving from Nature*, Lecture Notes in Computer Science, Vol.1917, pp.849–858 (2000).
29) Kauffman, S.A.: *The Origins of Order: Self-Organization and Selection in Evolution*, Oxford University Press (1993).
30) Glover, F., Laguna, M. and Marti, R.: Fundamentals of scatter search and path relinking, *Control and Cybernetics*, Vol.29, No.3, pp.653–684 (2000).
31) Adenso-Diaz, B., Garcia-Carbajal, S. and Gupta, S.M.: A path-relinking approach for a bi-criteria disassembly sequencing problem, *Computers & Operations Research*, Vol.35, No.12, pp.3989–3997 (2008). Part Special Issue: Telecommunications Network Engineering.
32) Nolz, P., Doerner, K.F., Gutjahr, W.J. and Hartl, R.: A bi-objective metaheuristic for disaster relief operation planning, *Advances in Multi-Objective Nature Inspired Computing*, Studies in Computational Intelligence, Vol.272, pp.167–187 (2010).
33) Pasia, J., Hartl, R. and Doerner, K.: Solving bi-objective flowshop problem using pareto-ant colony optimization, *International Workshop on Ant Colony Optimization and Swarm Intelligence*, Lecture Notes in Computer Science, Vol.4150, pp.294–305 (2006).
34) Ikeda, K. and Kobayashi, S.: Deterministic multi-step crossover fusion: A handy crossover composition for GAs, *Parallel Problem Solving from Nature*, Lecture Notes in Computer Science, Vol.2439, pp.162–171 (2002).
35) Deb, K.: *Multi-Objective Optimization using Evolutionary Algorithms*, John Wiley & Sons, Chichester, U.K. (2001).
36) Zitzler, E. and Thiele, L.: Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach, *IEEE Trans. Evolutionary Computation*, Vol.3, No.4, pp.257–271 (1999).
37) Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. and Fonseca, V.: Performance assessment of multiobjective optimizers: An analysis and review, *IEEE Trans. Evolutionary Computation*, Vol.7, pp.117–132 (2003).
38) Fonseca, C., Paquete, L. and López-Ibáñez: An improved dimension-sweep algorithm for the hypervolume indicator, *IEEE Congress on Evolutionary Computation*, pp.1157–1163 (2006).
39) Zitzler, E. and Thiele, L.: Multiobjective optimization using evolutionary algorithms — a comparative case study, *Parallel Problem Solving from Nature*, Lecture Notes in Computer Science, Vol.1498, pp.292–301 (1998).

(Received April 22, 2010)
(Revised June 11, 2010)
(Revised(2) July 31, 2010)
(Accepted September 9, 2010)

Joseph M. Pasia received his B.S. and M.S. degrees at the Department of Mathematics, University of the Philippines-Diliman, Quezon City, Philippines in 1998 and 2001, respectively, where he is currently an assistant professor. In 2006, he received his Dr. rer. soc. oec. degree from the University of Vienna, Vienna, Austria. He was a research fellow at the Faculty of Engineering, Shinshu University, Japan from 2009 to 2010. His research interests include operations research in transportation and logistics, evolutionary computations, and their applications. He is a member of the National Research Council of the Philippines and the Computing Society of the Philippines.

Hernán Aguirre received his Engineer degree in computer systems from Escuela Politécnica Nacional, Quito, Ecuador in 1992. From 1997 to 2003 he was a research scholar sponsored by the Japanese Ministry of Education, Culture, Sports, Science and Technology. He received his M.S. and Ph.D. degrees from Shinshu University, Japan, in 2000 and 2003, respectively. Currently, he is an assistant professor at Shinshu University. His research interests include evolutionary computation, multidisciplinary design optimization, computational intelligence, information security, parallel computing, and their applications. He is a member of IEEE, ACM-SIGEVO, and IPSJ.

Kiyoshi Tanaka received his B.S. and M.S. degrees in Electrical Engineering and Operations Research from National Defense Academy, Yokosuka, Japan, in 1984 and 1989, respectively. In 1992, he received Dr. Eng. degree from Keio University, Tokyo, Japan. In 1995, he joined the Department of Electrical and Electronic Engineering, Faculty of Engineering, Shinshu University, Nagano, Japan, where he is currently a professor. His research interests include image and video processing, information hiding, evolutionary computation, chaos & fractals, and their applications. He is a member of IEEE, IEICE, IPSJ and IIEEJ. He is the vice-chairman of journal editing committee of IIEEJ.