METHODOLOGIES AND APPLICATION

Improved RM-MEDA with local learning

Yangyang Li · Xia Xu · Peidao Li · Licheng Jiao

Abstract In this paper, local learning is proposed to improve the speed and the accuracy of convergence performance of regularity model-based multiobjective estimation of distribution algorithm (RM-MEDA), a typical multi-objective optimization algorithm via estimation of distribution. RM-MEDA employs a model-based method to generate new solutions, however, this method is easy to generate poor solutions when the population has no obvious regularity. To overcome this drawback, our proposed method add a new solution generation strategy, local learning, to the original RM-MEDA. Local learning produces solutions by sampling some solutions from the neighborhood of elitist solutions in the parent population. As it is easy to search some promising solutions in the neighborhood of an elitist solution, local learning can get some useful solutions which help the population attain a fast and accurate convergence. The experimental results on a set of test instances with variable linkages show that the implement of local learning can accelerate convergence speed and add a more accurate convergence to the Pareto optimal.

Keywords Multi-objective optimization problem · Estimation of distribution algorithm · Local learning

1 Introduction

In real world application, most of the optimization problems involve several objectives. Different from single objective optimization problems, objectives in a multi-objective optimization problem (MOP) often conflict with each other and these objectives cannot be optimized by a single solution at the same time. So there is a need to find a set of solutions to approximate the Pareto optimal set (PS) for decision maker to make choice. As multi-objective evolutionary algorithms (MOEAs) use a population-based approach which can produce a group of solutions to approximate the PS in a single run, they become the most effective methods in solving multi-objective optimization problems.

During the past two decades, MOEAs have been obtained an increasing attention among optimization community. A number of evolutionary algorithms have been developed for multi-objective problems. The Niched Pareto Genetic Algorithm (NPGA) (Horn et al. 1994), the Non-dominated Sorting Genetic Algorithm (NSGA) (Srinivas and Deb 1994) and the Strength Pareto Evolutionary Algorithm (SPEA) (Zitzler and Thiele 1999) constitute the pioneering multi-objective approaches which mainly using selection mechanisms based on Pareto ranking and fitness sharing to maintain diversity. In the past few years, MOEAs characterized by the use of elitism strategy were proposed such as the Pareto Archived Evolution Strategy (PAES) (Knowles and Corne 2000), the Pareto Envelope based Selection Algorithm (PESA) (Corne et al. 2001a), the Micro Genetic Algorithm (micro-GA) (Coello and Pulido 2001), the revised version of PESA with region based selection (PESA-II) (Corne et al. 2001b), the improved version of SPEA (SPEA2) (Zitzler et al. 2002) and the improved version of NSGA (NSGA-II) (Deb et al. 2002). SPEA2 employs a nearest neighbor density estimation technique, an enhanced archive truncation method and a revised fitness assignment strategy to account the number of individuals it dominates and it is dominated by. In NSGA-II, a more efficient nondominated sorting method, elitism and a crowded comparison operator without specifying any addi-

Y. Li (✉) · X. Xu · P. Li · L. Jiao
Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China, Xidian University, Xi'an 710071, China
e-mail: lyy_791@163.com; yyli@xidian.edu.cn

tional parameters or diversity maintaining are derived. In the current, some new dominance concepts have been introduced in traditional MOEAs, such as ε-dominance (Laumanns et al. 2002), adaptive ε-dominance (Hernández-Díaz et al. 2007) and r-dominance (BenSaid et al. 2010). Also, many attempts (using swarm intelligence) have been made to improve the performance of MOEAs. A new version of MOEA/D based on differential evolution is proposed in Li and Zhang (2009). Ghoseiria and Nadjari (2010) presented an algorithm based on multiobjective ant colony optimization to solve the bi-objective shortest path problem. And Jamuna and Swarup (2012) proposed a multiobjective biogeography based optimization algorithm to design optimal placement of phasor measurement units.

These widely known traditional MOEAs (Horn et al. 1994; Srinivas and Deb 1994; Zitzler and Thiele 1999; Knowles and Corne 2000; Corne et al. 2001a,b; Coello and Pulido 2001; Zitzler et al. 2002; Deb et al. 2002) mainly adopt crossover and mutation to produce new solutions, they work well when solving problems with simple PS. However, as is mentioned in Deb et al. (2006), it is usually difficult for them to solve MOPs with complex PS. This is mainly due to the fact that only a few parent solutions are directly involved in these reproduction operations. And there is no mechanism in conventional MOEAs for extracting global statistical information from the previous search and using it for guiding the further search. Thus, traditional MOEAs usually have a limited capacity for discovering.

Estimation of distribution algorithms (EDAs) are recognized as an important computing paradigm in evolutionary computation (Larrañaga and Lozano 2001). One of their main features is that they do not adopt traditional crossover or mutation operators to generate new solutions. Instead, they learn the global statistical information of the population to build a probability distribution model. And then new solutions are sampled from this model. According to the modeling of interaction between variables of optimization problems, EDAs presented are categorized into three different classes (Pelikan et al. 2000a). In the first class, relationship between parameters for a problem is assumed to be independent, thus the probability distribution of solutions can be factored as a product of independent univariate probabilities. The univariate marginal distribution algorithm (UMDA) (Mühlenbein and Paass 1996) and the probabilistic incremental learning (PBIL) (Baluja 1994) are the representative of these algorithms. Recent developments in the field of EDAs take into account possible interactions between variables. Modeling bivariate dependencies and multivariate variable interactions represent the second and the third class of EDAs, respectively. These EDAs are implemented by e.g., the mutual information maximization for input clustering (MIMIC) algorithm (De Bonet et al. 1997), the combining optimizers with mutual information trees algorithm

(COMIT) (Baluja and Davies 1998) and the Bayesian optimization algorithm (BOA) (Pelikan et al. 2000b). Compared with traditional MOEAs, EDAs take more the population distribution information into consideration rather than the individual local information. Recently, some EDAs (Khan et al. 2003; Laumanns and Ocenasek 2002; Okabe et al. 2004; Bosman and Thierens 2005; Kukkonen and Lampinen 2005; Pelikan et al. 2005) have been developed to solve continue MOPs. Khan combined the selection strategy in NSGA-II with BOA for multi-objective and hierarchically difficult problems (Khan et al. 2003). Laumanns and Ocenasek (2002) combined SPEA2 with BOA to solve multi-objective knapsack problem. However, these EDAs do not make use of the distribution regularity of PS in building model. As the probability modeling techniques under regularity have been widely used in the area of statistical learning (Zhou et al. 2005, 2006), it is suitable to take advantage of the regularity in the design of EDAs for a continuous MOP. Inspired by this fact, Zhang et al. developed a regularity model-based multiobjective estimation of distribution algorithm (RM-MEDA) (Zhang et al. 2008). In RM-MEDA, the model is built by the regularity that the Pareto optimal front (PF) and PS of a continuous MOP are piecewise continuous $(m-1)$-dimensional manifolds under mild conditions (Miettinen 1999). RM-MEDA outperforms traditional MOEAs on problems with complex PS. However, as the location information of elitist solutions found in parent population is not directly used to produce new solutions in EDA, it is difficult for a single EDA to solving some complicated optimization problems (Zhang et al. 2005). Therefore, how to combine EDAs with other techniques represents an important research direction (Zhang et al. 2007a).

Recently, combinations of MOEAs and local search heuristics, often called memetic algorithms (MAs), have been shown to outperform traditional evolutionary algorithms in a wide variety of scalar objective optimization problems. MAs have now been well used across a wide range of problem domains including multi-objective problems (Smith 2007). As early as 1998, Ishibuchi and Murata (1998) firstly propose a multi-objective genetic local search algorithm. Subsequently, numbers of MOEAs combined with memetic algorithm (Knowles and Corne 2002; Jaszkiewicz 2002, 2003; Liu et al. 2007) were presented. These studies show that the performance of evolutionary multi-objective optimization algorithms can be improved by hybridization with local search. Inspired by the above study, combining EDAs with local search or elitist strategy becomes a hot research direction in multi-objective optimization recently. To improve the convergence speed of RM-MEDA, elitist strategy based RM-MEDA (Li et al. 2010) was proposed, in which only some good solutions are selected to build the probability distribution model. However, as only elitist solutions can be used to generate new individuals, it is easy for RM-MEDA to fall into the local Pareto fronts. A hybrid algorithm combining

genetic algorithm (GA) with RM-MEDA was proposed to accelerate the convergence speed in Dai et al. (2009). Wang et al. (2012) proposed a reducing redundant cluster operator (RRCO) to build more precise model during the evolution, and combined it with RM-MEDA. In order to enhance the global search ability of RM-MEDA, Zhou and Zhang et al. added two operations, biased crossover and biased initialization to traditional RM-MEDA in Zhang et al. (2007b). It can effectively solve problems with many local Pareto fronts.

Although these strategies can enhance the performance of RM-MEDA, the shortcoming caused by the reproduction in RM-MEDA is not really solved. In this paper, a novel local learning strategy is designed for overcoming this drawback in RM-MEDA. We propose an improved RM-MEDA with local learning which is named as IRM-MEDA. Different from the original algorithm, two reproduction methods are employed in this IRM-MEDA. The model-based reproduction learns the global statistic information to generate a part of new solutions. And local learning reproduction produces another part of new solutions by using the neighborhood information of some elitist individuals. Experimental results on MOPs with variable linkages show that IRM-MEDA has superiority over the original RM-MEDA.

The rest of this paper is organized as follows. Section 2 describes related background including the definitions of multi-objective optimization and theoretical basic. The motivation and details of ours improved RM-MEDA are discussed in Sect. 3. In Sect. 4 experiment results and analysis are presented. And a conclusion is drawn in Sect. 5.

2 Related work

2.1 Multi-objective optimization problem

In this study, we consider the following continuous MOPs (Deb 2001; Coello et al. 2002).

$$\min \ F(x) = (f_1(x), f_2(x), \ldots, f_m(x))^T \\ s.t. \ x \in \Omega \tag{1}$$

where Ω is a continuous search space and x is a decision variable vector. $F : \Omega \rightarrow R^m$ is the map of decision variable space to m real-valued continuous objective space.

Definition 1 A solution $a = (a_1, a_2, \ldots, a_m)$ is said to dominate $b = (b_1, b_2, \ldots, b_m)$, denoted as $a \succ b$ if and only if the following equation is satisfied.

$$\forall i \in \{1, \ldots, m\}, a_i \leq b_i \wedge \exists j \in \{1, \ldots, m\} : a_j < b_j \tag{2}$$

Definition 2 A vector $x^* \in \Omega$ is called a Pareto optimal solution or nondominated solution if and only if the following equation is satisfied.

$$\neg \exists x \in \Omega : x \succ x^* \tag{3}$$

All the Pareto optimal solutions consist of the Pareto optimal set. The corresponding image of the Pareto optimal set in the objective space is what we call the Pareto optimal front. The purpose of all the multi-objective optimization algorithms is to find solutions as close to the PF as possible and maintain the diversity at the same time.

Under certain smoothness conditions, it can be induced from the Karush–Kuhn–Tucker condition that the PS of a continuous MOP is a $(m - 1)$-dimensional piecewise continuous manifold in decision space (Zhang et al. 2007a; Schütze et al. 2003). Therefore the PS of a continuous MOP with two objectives is a piecewise continuous curve in Ω, while the PS of a three-objectives MOP is a piecewise continuous surface.

2.2 RM-MEDA

We hope that the solution set in decision space in a MOEA for Eq. (1) will approximate the PS as close as possible and at the same time, can locate around the PS. Figure 1 illustrates the distribution of solutions in the decision space in a successful MOEA. So the solution in the population can be supposed as an independent observation of a random vector $\xi \in R^n$ whose central is the PS. As indicated in the theoretical basic, the PS of MOEA for Eq. (1) is an $(m - 1)$-dimensional piecewise continuous manifold, therefore ξ can be described by

$$\xi = \xi + \varepsilon \tag{4}$$

where ξ is uniformly distributed over the manifold, and ε is an n-dimensional zero-mean noise vector.

According to the regularity of the Pareto set discussed above, RM-MEDA builds the probability model. In the procedure of the modeling-based reproduction, the population $P(t)$ is first partitioned into K disjoint clusters by Local PCA algorithm (Kambhatla and Leen 1997). And then a model for Eq. (5) is built for each cluster. Finally new solutions are sampled from these models. Figure 2 shows the procedure of RM-MEDA and the detail of each steps can be further studied in Zhang et al. (2008). In the following figure, N
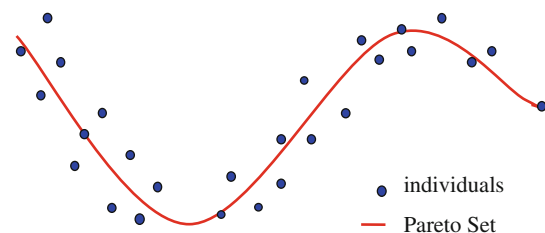

Fig. 1 The solutions distribution in the decision space in a successful MOEA

Procedure 1:

Step1 Initialize: Initialize the population $P(t)$ and evaluate $P(t)$, t=0.

Step2 Modeling-based Reproduction:

    i).   Learn the distribution of the solutions in $P(t)$ and build the probability model.

    ii). Sample N individuals from the probability model and merge them into $Q(t)$.

Step3 Evaluation: Evaluate $Q(t)$.

Step4 Selection: Select N solutions from $P(t) \cup Q(t)$ to create $P(t+1)$ by NDS-Selection method, t=t+1.

Step5 Stopping Condition: If the stop condition is satisfied, then output the Pareto solutions from $P(t+1)$, otherwise go to **Step2**.

Fig. 2 The procedure of RM-MEDA

Procedure 2:

Step1 Initialize: Initialize the population $P(t)$ and evaluate $P(t)$, t=0.

Step2 Sort: Sort $P(t)$ with the fast nondominated sorting approach [9].

Step3 Local Learning Reproduction:

    i).   Select an elitist individual X from $P(t)$ with tournament selection [50].

    ii). If X is a Pareto solution, then set the length coefficient as r, else set the length coefficient as $2r$.

    iii). Sample an individual solution from Br randomly and merge it into $Q(t)$.

    iv). Return to i) until $\theta \times N$ solutions are generated by local learning.

Step4 Modeling-based Reproduction: Sample $(1-\theta) \times N$ individuals by modeling-based reproduction and merge them to $Q(t)$.

Step5 Evaluation: Evaluate $Q(t)$.

Step6 Selection: Select N solutions from $P(t) \cup Q(t)$ to create $P(t+1)$ by NDS-Selection method, t=t+1.

Step7 Stopping Condition: If the stop condition is satisfied, then output the Pareto solutions from $P(t+1)$, otherwise go to **Step3**.

Fig. 3 The produce of IRM-MEDA

is the population size. $P(t)$ and $Q(t)$ represents the parent population and offspring population, respectively.

3 RM-MEDA with local learning

3.1 Motivation

Based on the regularity of the PS, RM-MEDA builds a probability distribution and then samples from the model to produce new solutions. It works well when the regularity is obvious. However, new solutions obtained are always poor at the start of the iteration when the population has no obvious regularity. This makes the population converge very slowly. One of the reasons leading poor convergence is that the model-based sampling does not directly use the location information of high-quality solutions in the current population to generate new solutions. To overcome this drawback, we add a new solution generation strategy, local learning, to the original RM-MEDA. Local learning produces solutions by sampling some solutions from the neighborhood of elitist solutions in the parent population. As it is easy to search some promising solutions in the neighborhood of a good solution, local learning can get some useful solutions to help the population attain a fast and accurate convergence.

3.2 Algorithm framework

The framework of IRM-MEDA will be discussed in this part. Firstly, the major steps in IRM-MEDA are listed in Fig. 3. In this figure, N is the population size. θ is a proportional coefficient and r is a length coefficient. $P(t)$ and $Q(t)$ represents the parent population and offspring population, respectively. Br denotes the neighborhood of the solution X and it is defined as follow:

$$Br = \{Y | Y_i \in [X_i - r_i \times (X_{uppi} - X_{lowi}), \\ X_i + r_i \times (X_{uppi} - X_{lowi})]\} \qquad (5)$$

where X_{upp} and X_{low} are the value limits of X. The central of Br is X and the length of Br is $r \times (X_{upp} - X_{low})$.

The difference of our improved method with RM-MEDA is the addition of the Step2 and Step3 in our proposed method. In order to introduce the location information of elitist solutions in the parent population into the offspring population directly, local learning is adopted as a method of reproduction in this improved algorithm.

Firstly, $\theta \times N$ high-quality solutions are selected from the parent population $P(t)$ with tournament selection (Miller and Goldberg 1995). In the first iteration, Step2 employs the fast nondominated sorting approach in NSGA-II (Deb et al.
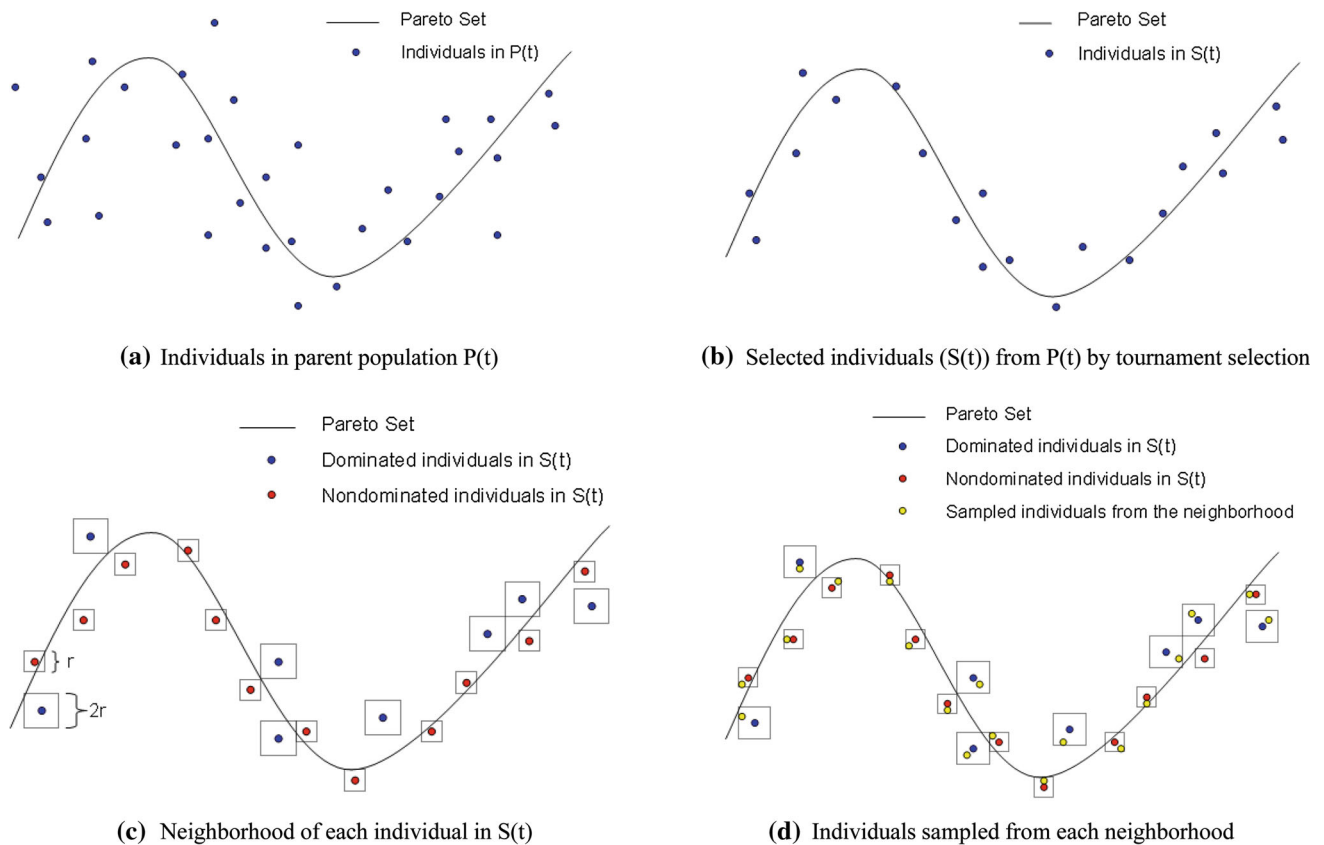
(a) Individuals in parent population P(t)

(b) Selected individuals (S(t)) from P(t) by tournament selection

(c) Neighborhood of each individual in S(t)

(d) Individuals sampled from each neighborhood

Fig. 4 The presentation chart of Local Learning to generate the offspring population from the parent population

2002) to sort the parent population. Solutions with lower nondominated front will be set a smaller number. And the individual with a smaller number will have a higher priority to be selected in the tournament selection (Miller and Goldberg 1995). In the rest of the iterations, parent population has been sorted after making a NDS-selection (Okabe et al. 2004) in the last iteration. Then tournament selection (Miller and Goldberg 1995) is done based on the result of this sorting.

Then, individuals are sampled from the neighborhood of these elitist solutions. To find a balance between exploration and exploitation, this study defines that the neighborhood of a nondominated solution is smaller than a dominated solution. If the length coefficient of the neighborhood of a nondominated solution is r, that of a dominated solution will be set as $2r$. By this, excellent solutions can be sampled from the neighborhood of the nondominated solutions to improve convergence. And solutions sampled from the neighborhood of the dominated solutions can increase the diversity. Finally, these individuals obtained are merged into the offspring population $Q(t)$. In addition, if an element of a solution generated by the local learning reproduction is out of the feasible decision space, we simply reset its value to be

a randomly selected value inside the feasible decision space. This method is the same with that used in RM-MEDA.

Here is a simple example to illustrate how Local Learning generates the offspring population from the parent population.

Step i) Input the parent population P(t). As Fig. 4a shows, the black curve represent the pareto set and the blue points is the individuals in P(t);

Step ii) Sort P(t) with the fast nondominated sorting approach (Deb et al. 2002), and select $\theta \times N$ high-quality individuals to S(t) from the parent population P(t) with tournament selection (Miller and Goldberg 1995). N is the population size and θ is the proportional coefficient (one of the parameters in Local Learning). Figure 4b shows the selected individuals in S(t);

Step iii) Determine the neighborhood length (Br) of each individual in S(t). In Fig. 4c, the red points are the nondominated individuals whose neighborhood length is r. And the blue points (the dominated individuals) have a neighborhood length of $2r$. r is length coefficient set beforehand.

Step iv) Sample one individual solution from each neighborhood randomly as the yellow points in Fig. 4d. The yellow points are just the solutions generated by Local Learning.

4 Experiment results and discussion

Experiments designed for learning the performance of IRM-MEDA will be discussed in this section. Firstly, test problems and performance metric will be studied in Part A. Then, the parameters involved in IRM-MEDA are discussed in Part B. In Part C, our proposed method is compared with RM-MEDA. Finally, two presented algorithms are cited to compared with IRM-MEDA in Part D.

4.1 Test problems and performance metric

Nine problems have been taken from reference Zhang et al. (2008) to validate the performance of IRM-MEDA. Instances are listed in Table 1. These test instances involve three types of problems. Test instances F1–F4 are problems with linear variable linkages which are obtained by performing the following linear mapping on the variables in ZDT1, ZDT2, ZDT6 (Zitzler et al. 2000), and DTZL2 (Deb et al. 2005):

$$x_1 \to x_1, \; x_i \to x_i - x_1, \; i = 2, \ldots, n$$

F1–F3 have the same PS which is a line segment. The PS of F4 is a 2-D rectangle. Test instances F5-F8 are instances with nonlinear variable linkages. These test problems are obtained by performing nonlinear mapping on the variable in the original ZDT and DTZL problems:

$$x_1 \to x_1, \; x_i \to x_i^2 - x_1, \; i = 2, \ldots, n$$

The PS of F5-F7 is a bounded continuous curve and The PS of F8 is a 2-D continuous bounded surface which is more complex. And F9 is an instance with nonlinear variable linkages. Furthermore, this problem has many local Pareto fronts, which makes it more difficult for a MOEA to approximate the true Pareto front.

This study adopts the inverted generational distance (IGD) as the performance metric to measure the obtained approximation nondominated solutions. This performance metric used in this study is introduced from Zhang et al. (2008).

IGD is an inverted variation of the widely used generational distance (GD) performance metric (Veldhuizen and Lamont 1998). The GD indicates the average distance of the points in an approximation to the true Pareto front. However, the GD performance metric cannot measure the diversity of the approximation effectively. Let P^* be a set of uniformly distributed points in the objective space along the PF, and let P be an approximation to the PF. The definition of the

Table 1 Test instances

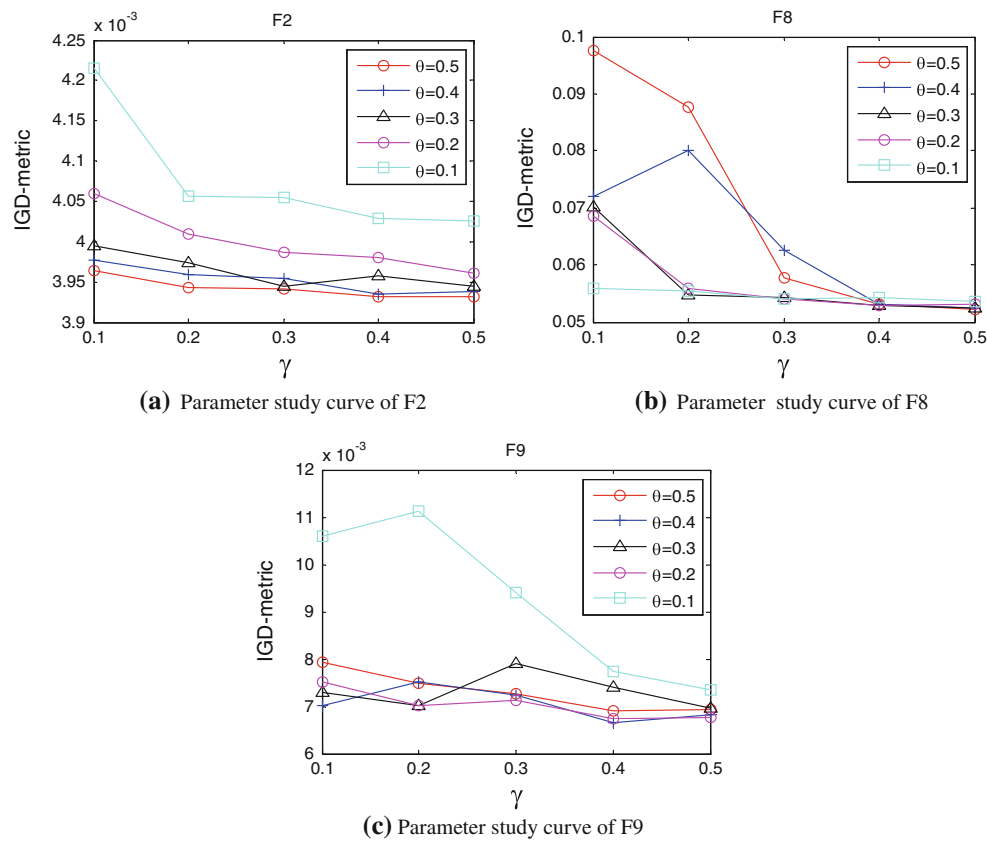| Instances	Variable	Objectives	Evaluation
F1	$[0, 1]^n, n = 30$	$f_1(x) = x_1 \; f_2(x) = g(x)[1 - \sqrt{f_1(x)/g(x)}]$ $g(x) = 1 + 9(\sum_{i=2}^{n}(x_i - x_1)^2)/(n - 1)$	10, 000
F2	$[0, 1]^n, n = 30$	$f_1(x) = x_1 \; f_2(x) = g(x)[1 - (f_1(x)/g(x))^2]$ $g(x) = 1 + 9(\sum_{i=2}^{n}(x_i - x_1)^2)/(n - 1)$	10, 000
F3	$[0, 1]^n, n = 30$	$f_1(x) = 1 - \exp(-4x_1)\sin^6(6\pi x_1)$ $f_2(x) = g(x)[1 - (f_1(x)/g(x))^2]$ $g(x) = 1 + 9(\sum_{i=2}^{n}(x_i - x_1)^2)/(n - 1)$	100, 000
F4	$[0, 1]^n, n = 30$	$f_1(x) = \cos(\frac{\pi}{2}x_1)\cos(\frac{\pi}{2}x_2)(1 + g(x))$ $f_2(x) = \cos(\frac{\pi}{2}x_1)\sin(\frac{\pi}{2}x_2)(1 + g(x))$ $f_3(x) = \sin(\frac{\pi}{2}x_1)(1 + g(x))g(x) = \sum_{i=3}^{n}(x_i - x_1)^2$	40, 000
F5	$[0, 1]^n, n = 30$	$f_1(x) = x_1 \; f_2(x) = g(x)[1 - \sqrt{f_1(x)/g(x)}]$ $g(x) = 1 + 9(\sum_{i=2}^{n}(x_i^2 - x_1)^2)/(n - 1)$	10, 000
F6	$[0, 1]^n, n = 30$	$f_1(x) = \sqrt{x_1} \; f_2(x) = g(x)[1 - (f_1(x)/g(x))^2]$ $g(x) = 1 + 9(\sum_{i=2}^{n}(x_i^2 - x_1)^2)/(n - 1)$	10, 000
F7	$[0, 1]^n, n = 30$	$f_1(x) = 1 - \exp(-4x_1)\sin^6(6\pi x_1)$ $f_2(x) = g(x)[1 - (f_1(x)/g(x))^2]$ $g(x) = 1 + 9[(\sum_{i=2}^{n}(x_i^2 - x_1)^2)/9]^{0.25}$	100, 000
F8	$[0, 1]^n, n = 30$	$f_1(x) = \cos(\frac{\pi}{2}x_1)\cos(\frac{\pi}{2}x_2)(1 + g(x))$ $f_2(x) = \cos(\frac{\pi}{2}x_1)\sin(\frac{\pi}{2}x_2)(1 + g(x))$ $f_3(x) = \sin(\frac{\pi}{2}x_1)(1 + g(x))g(x) = \sum_{i=3}^{n}(x_i^2 - x_1)^2$	40, 000
F9	$[0, 1] \times [0, 10]^{n-1}$, $n = 10$	$f_1(x) = x_1 \; f_2(x) = g(x)[1 - \sqrt{f_1(x)/g(x)}]$ $g(x) = \frac{1}{4000}\sum_{i=2}^{n}(x_i^2 - x_1)^2 - \prod_{i=2}^{n}\cos(\frac{x_i^2 - x_1}{\sqrt{i-1}}) + 2$	100, 000

(a) Parameter study curve of F2

(b) Parameter study curve of F8

(c) Parameter study curve of F9

Fig. 5 The average IGD-metric value versus the length coefficient r under the different settings of the proportional coefficient θ

IGD from P^* to P is illustrated as follow, where $d(v, P)$ represents the minimum Euclidean distance between v and the points in P.

$$IGD(P^*, P) = \frac{\sum_{v \in P^*} d(v, P)}{|P^*|} \quad (6)$$

If the $|P^*|$ is large enough to represent the true PF well, the IGD-metric value can measure the diversity and convergence at the same time. A smaller IGD-metric value indicates a better performance. In our experiments, we select 1000 evenly distributed points in the PF and let these points be the P^* in all the test instances.

4.2 Runtime environment and parameters study

In this paper, all of our simulations are executed on a 2.4-GHz Pentium IV PC with 1G RAM. And the software environment is Matlab [R2010b, 32-bit (win32)].

In IRM-MEDA, there are two parameters need to be fixed in the procedure of local learning reproduction. The proportional coefficient θ represents the number of solutions generated by local learning and the length coefficient r indicates the neighborhood size of a solution. Local learning can help the population attain the regularity early and converge faster. However, if unsuitable θ and r are selected, the population

may fall into the local optimal and local learning may make no sense. So a proper selection of θ and r is important.

We study the effect of the variation of θ and r from 0 to 0.5 on three types of problems in terms of the average IGD-metric in 20 runs. Test instance F2 represents problem with linear variable linkages. F8 represents problem with non-linear variable linkages and it is a three-objective MOP. F9 represents problem with many local Pareto fronts.

The results are shown in Fig. 5. As clearly shown in this figure, IRM-MEDA can get a small IGD-metric value when r is in [0.3 0.5], especially on F8. And IRM-MEDA is able to attain a better IGD-metric value when θ is 0.4 or 0.5 and r is in [0.3 0.5]. From the above results, it can be said that IRM-MEDA achieves better results when r is 0.4 and θ is 0.4 in these three test instances. Therefore, we set $\theta = 0.4$ and $r = 0.4$ for all the instances in our experiments.

4.3 Longitudinal comparison with RM-MEDA

In this part, three longitudinal comparisons are shown with RM-MEDA. Firstly, a comparison with conventional setting is given. And it will display IRM-MEDA's outperform than RM-MEDA. Then, to farther show the advantage of our algorithm, we reduce the maximal number of function evaluations and run the two algorithms again. And the results are really

impressive. At last, it is a comparison of convergence property.

4.3.1 Comparison with conventional setting

By reference Zhang et al. (2008), the population size in both of the algorithms is set to be 100 for bi-objective problems and 200 for three-objective problems. The number of clusters K is set to be 5 in RM-MEDA. In IRM-MEDA, the proportional coefficient θ is set to be 0.4 and the neighborhood length coefficient r is set to be 0.4 for all the instances. The algorithms stop after a given number of function evaluations. The maximal function evaluations are shown in Table 1. Both of the algorithms are run independently 20 times for each instance.

The IGD-metric values, i.e., mean value (Mean), best value (Best), worst value (Worst) and standard deviations (Std) of the final nondominated solutions obtained by RM-MEDA and IRM-MEDA are shown in Table 2. In order to show how well the true Pareto front is covered by the final nondominated solutions obtained, we plot all the 20 nondominated fronts found together for showing their distribution ranges in the objective space in Fig. 6. This is another way to measure the performance of these two algorithms.

From Table 2, we can see that for most of the instances, IRM-MEDA outperforms RM-MEDA in terms of IGD-metric value. The superiority of IRM-MEDA on F3 and F6 is obvious. The worst IGD-metric values obtained by IRM-MEDA are better than the best IGD-metric values achieved by RM-MEDA on these two problems. It is clear from Fig. 6 that for F3 and F6, the final nondominated solutions attained by IRM-MEDA can approximate the true PF very well. While, there is still a distance between some nondominated solutions get by RM-MEDA and the true PF.

F3 is the hardest among four test instances—F1, F2, F3 and F4. The distribution of the Pareto optimal solutions in the objective space in this instance is very different from that in the other three. If we uniformly sample a number of points in the PS of F3 in the decision space, most of the corresponding Pareto optimal vectors in the objective space will be more likely to be in the left part of the PF. Thus, it is not easy for an algorithm to approximate the whole PF. Although RM-MEDA works well on F3, our improved IRM-MEDA does much better than RM-MEDA. This could be attributed to the fact that even there are a few of solutions in the right part of PF, IRM-MEDA can quickly learn the information about these usefulness solutions and use it to create offspring solutions.

F6 and F2 are both the variants of ZDT2. Compared with F2, it is more difficult to approximate the PF of F6. RM-MEDA can't converge every well in the given maximal function evaluations. Nevertheless, the Local Learning with using the local information of high-quality solutions make IRM-MEDA converge better than RM-MEDA.

As for test instances F1, F2, F4, F5 and F8, IRM-MEDA still has slightly superiority over RM-MEDA. The difference of the best IGD-metric values obtained by these two algorithms is not apparent on F1, F2 and F5, while the worst IGD-metric values attained by IRM-MEDA are much smaller than that of RM-MEDA on these instances. This is mainly due to the fact that IRM-MEDA can get a more stable convergence compared with RM-MEDA.

For instance F7, it is obvious from Fig. 6 that most of the nondominated solutions obtained by RM-MEDA are closer to the true PF. However, the mean IGD-metric value get by RM-MEDA is still larger than that of IRM-MEDA on this instance. This is because that the number of the final nondominated solutions attained by RM-MEDA sometimes is smaller than the population size, which leads to a very poor IGD-metric value.

F9 is an instance with many local Pareto fronts. It is easy for RM-MEDA to jump into local Pareto fronts on this instance. While from Table 2 and Fig. 6, we can see that the performance of IRM-MEDA on F9 is still very well. The rea-

Table 2 The IGD-Metric performance on F1–F9

IGD-metric	RM-MEDA				IRM-MEDA			
	Mean	Best	Worst	Std	Mean	Best	Worst	Std
F1	0.0054	0.0046	0.0068	5.4177e−004	0.0040	0.0039	0.0041	3.3158e−005
F2	0.0063	0.0050	0.0119	0.0019	0.0039	0.0039	0.0040	2.7375e−005
F3	0.0088	0.0036	0.0177	0.0053	0.0022	0.0021	0.0024	6.4967e−005
F4	0.0460	0.0440	0.0478	0.0010	0.0419	0.0411	0.0428	5.1249e−004
F5	0.0123	0.0058	0.0277	0.0057	0.0049	0.0047	0.0052	1.3602e−004
F6	0.2463	0.0222	0.4058	0.1130	0.0050	0.0046	0.0055	2.3223e−004
F7	0.1545	0.0829	1.0238	0.2103	0.0384	0.0360	0.0416	0.0013
F8	0.0630	0.0581	0.0749	0.0040	0.0532	0.0501	0.0560	0.0015
F9	0.0204	0.0051	0.0673	0.0173	0.0066	0.0048	0.0132	0.0024

(a1) PF of F1 by RM-MEDA   **(b1)** PF of F2 by RM-MEDA   **(c1)** PF of F3 by RM-MEDA

(a2) PF of F1 by IRM-MEDA   **(b2)** PF of F2 by IRM-MEDA   **(c2)** PF of F3 by IRM-MEDA

(d1) PF of F4 by RM-MEDA   **(e1)** PF of F5 by RM-MEDA   **(f1)** PF of F6 by RM-MEDA

(d2) PF of F4 by IRM-MEDA   **(e2)** PF of F5 by IRM-MEDA   **(f2)** PF of F6 by IRM-MEDA
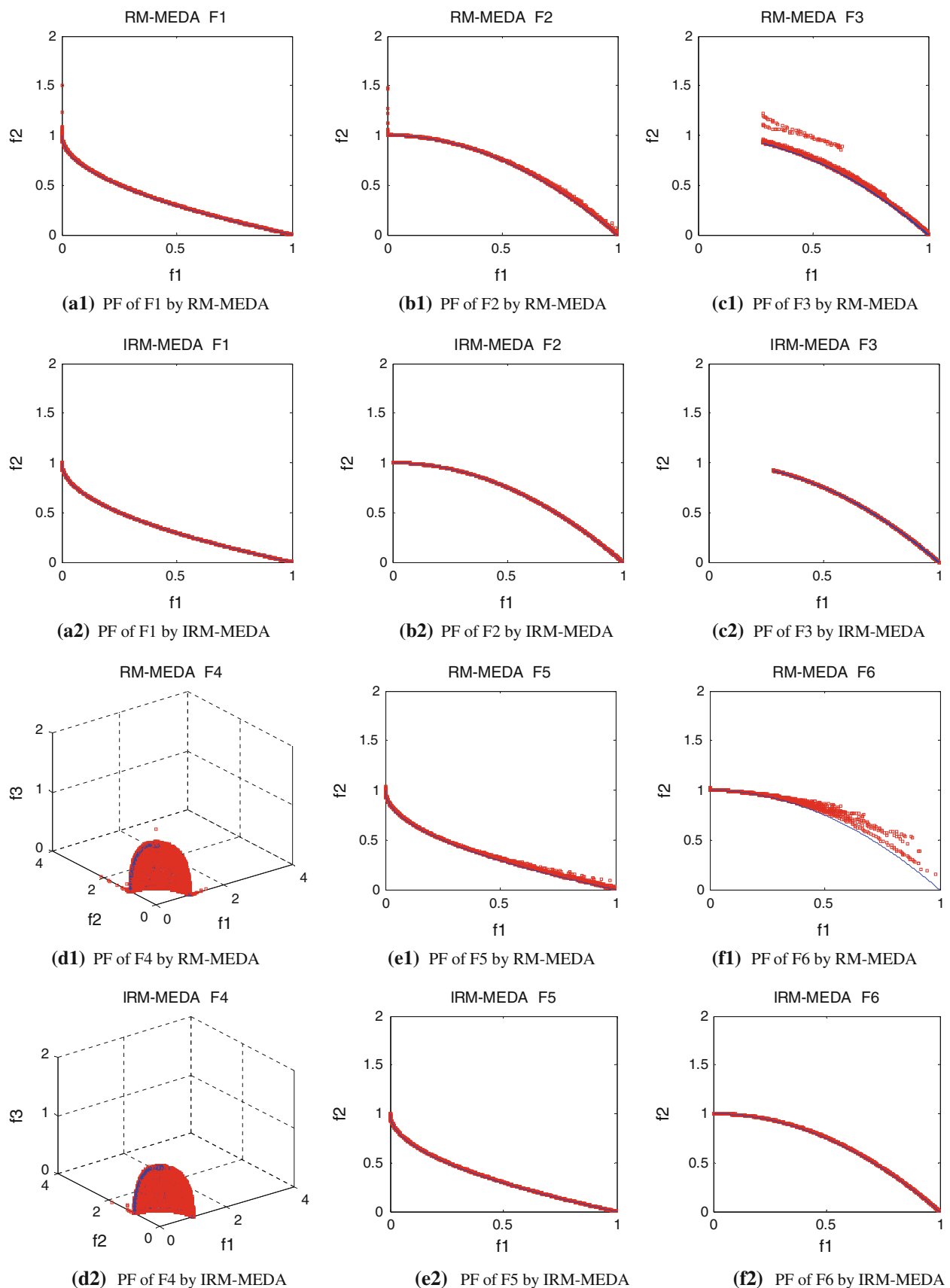
Fig. 6 The final nondominated fronts found in 20 runs on F1–F9

(g1) PF of F7 by RM-MEDA **(h1)** PF of F8 by RM-MEDA **(i1)** PF of F9 by RM-MEDA

(g2) PF of F7 by IRM-MEDA **(h2)** PF of F8 by IRM-MEDA **(i2)** PF of F9 by IRM-MEDA
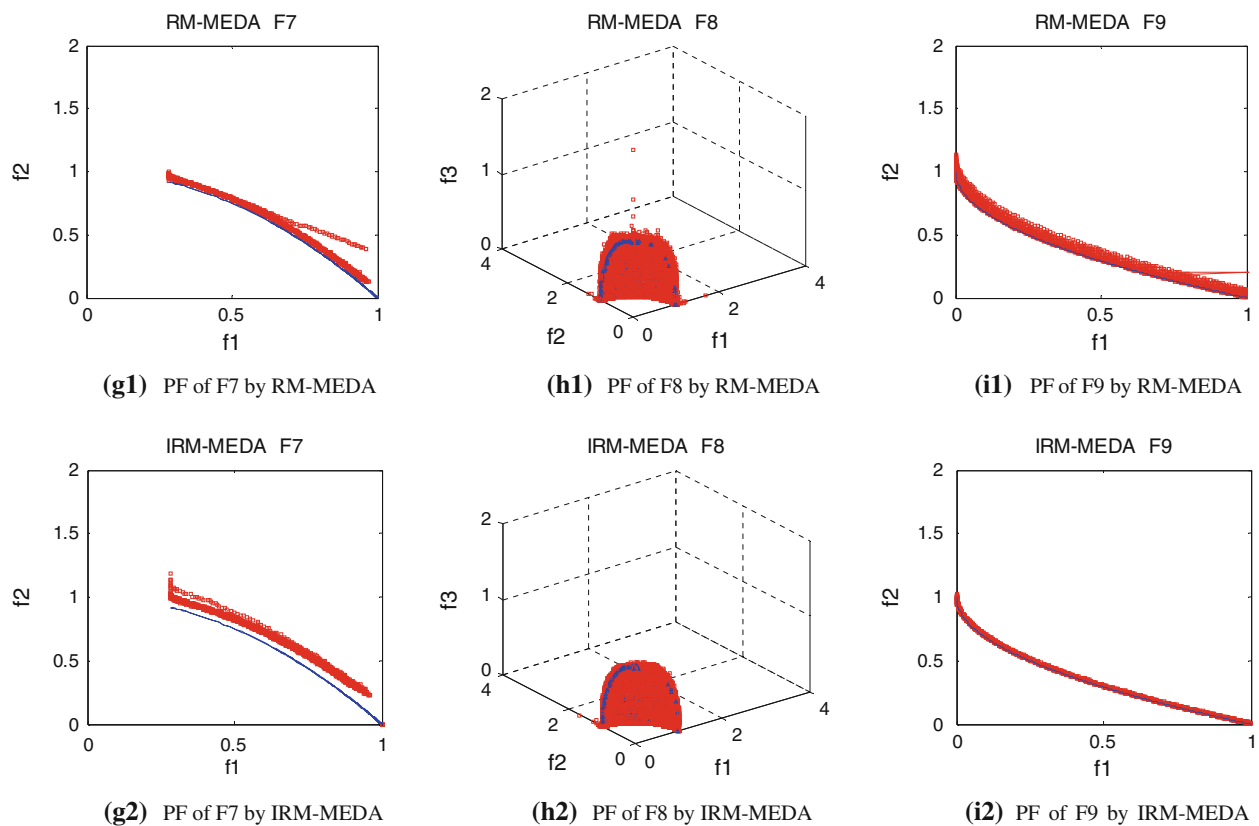
Fig. 6 continued

Table 3 The IGD-Metric performance on F1–F9 (runned with half of primitive maximal number of function evaluations)

IGD-metric	RM-MEDA				IRM-MEDA			
	Mean	Best	Worst	Std	Mean	Best	Worst	Std
F1	0.0338	0.0144	0.0733	0.0229	0.0041	0.0039	0.0041	5.4e−005
F2	0.1130	0.0273	0.2179	0.0795	0.0039	0.0039	0.0040	4.0e−005
F3	0.0804	0.0562	0.1529	0.0281	0.0022	0.0021	0.0023	6.8e−005
F4	0.0619	0.0543	0.0821	0.0090	0.0427	0.0421	0.0432	0.0004
F5	0.1966	0.1012	0.3111	0.0796	0.0059	0.0054	0.0064	0.0003
F6	0.4497	0.3207	0.5373	0.0577	0.0063	0.0055	0.0069	0.0005
F7	0.3357	0.1066	1.0304	0.3725	0.0494	0.0456	0.0561	0.0031
F8	0.1291	0.1047	0.1722	0.0223	0.0599	0.0559	0.0622	0.0019
F9	0.0290	0.0058	0.1681	0.04956	0.0067	0.0053	0.0099	0.0018

son is that local learning improves the local search ability of IRM-MEDA. It directly uses the location information of the previous good solutions to generate offspring, which makes IRM-MEDA perform better than RM-MEDA at refining a solution, particularly, when it is close to the PS.

From the above experimental results, we can conclude that IRM-MEDA outperforms on most of the instances. As mentioned before, IRM-MEDA is different from RM-MEDA only in its individual generation in heuristic search. Therefore, the better performance of IRM-MEDA in contrast to RM-MEDA results from the unique local learning solution

generation in IRM-MEDA. This local learning mechanism in IRM-MEDA can enhance the local search ability and make the population attain regularity early, so that a faster convergence is achieved.

4.3.2 Comparison with reduced maximal number of function evaluations

The local learning gives IRM-MEDA stronger ability of local search than RM-MEDA. It will increase its search speed and make it faster to find the solution of problems. To farther show

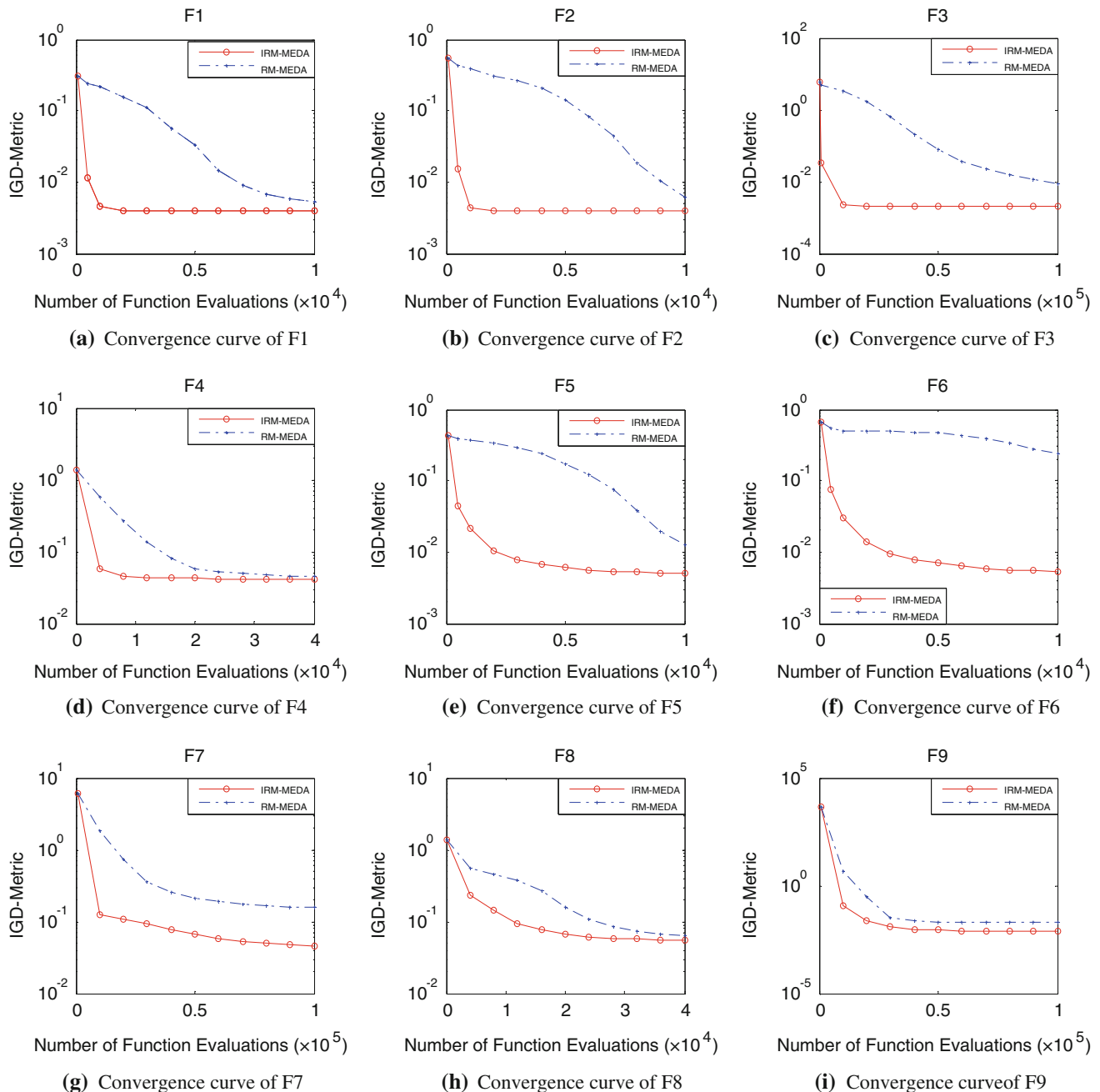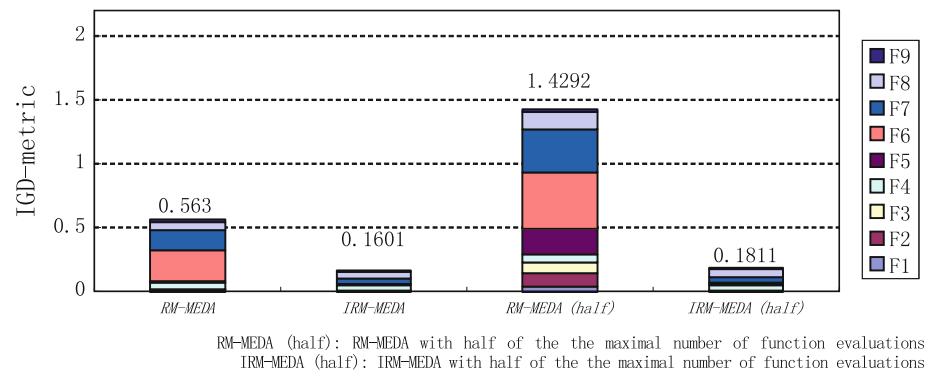**Fig. 7** The total IGD-metric of RM-MEDA and IRM-MEDA on F1–F9

RM-MEDA (half): RM-MEDA with half of the the maximal number of function evaluations
IRM-MEDA (half): IRM-MEDA with half of the the maximal number of function evaluations

(a) Convergence curve of F1

(b) Convergence curve of F2

(c) Convergence curve of F3

(d) Convergence curve of F4

(e) Convergence curve of F5

(f) Convergence curve of F6

(g) Convergence curve of F7

(h) Convergence curve of F8

(i) Convergence curveof F9

Fig. 8 The average IGD-Metric of the final nondominated solutions in 20 runs versus the number of function evaluations on F1–F9

display the advantages of IRM-MEDA, we reduce the maximal number of function evaluations to half of its primitive value and keep others parameters unchanged. Table 3 shows the new result and form it IRM-MEDA shows an enough advantage than RM-MEDA. It is obvious that IRM-MEDA have greatly surpassed RM-MEDA on all problems except for F4.

Figure 7 shows the total IGD-metric of RM-MEDA and IRM-MEDA on F1–F9. From this figure, the performance of RM-MEDA degenerated very fast when reduce the maximal number of function evaluations. But IRM-MEDA still maintains a good property. It just illustrate that the Local Learning we proposed is powerful and IRM-MEDA can search the solution faster than RM-MEDA.

4.3.3 Comparison of convergence property

To compare the abilities of the two algorithm's convergence, more comparisons are displayed in this part. Figure 8 shows the evolution of the average IGD-metric of the nondominated solutions among 20 independent runs with the number of function evaluations on all the instances. It clearly presents the converge speed of these two algorithms. We can see from this figure that IRM-MEDA converges faster than RM-MEDA on these instances, especially on F3, F5 and F6. RM-MEDA needs about 100,000 evaluations to converge well on F3. However, 10,000 evaluations can make IRM-MEDA

obtain an excellent convergence. The fast convergence is mainly due to the implement of local learning in IRM-MEDA which makes the population attain regularity early.
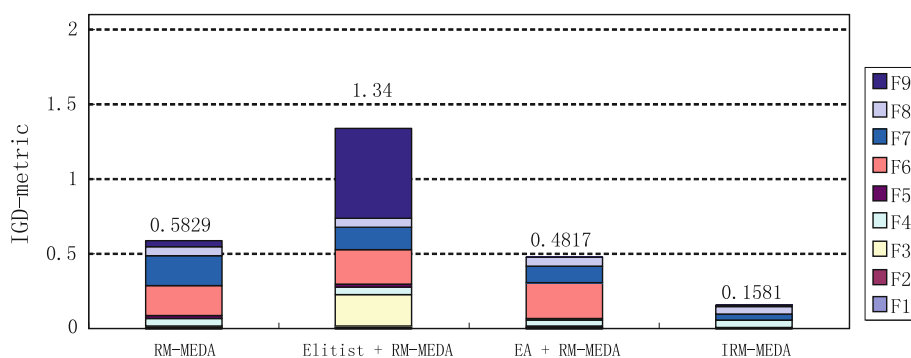
4.4 Crosswise comparison with previous methods

A crosswise comparison of IRM-MEDA with two improved methods, elitist strategy based RM-MEDA (Li et al. 2010) and a hybrid algorithm combining genetic algorithm with RM-MEDA (Dai et al. 2009) is made in this part. We also add RM-MEDA in this comparison to show the result more clearly. The results obtained by the four algorithms are listed in Table 4. In order to make a fair comparison, the population size and the maximal function evaluations setting in IRM-MEDA are the same as those in Li et al. (2010) or Dai et al. (2009).

The results reveal that the IGD-metric obtained by IRM-MEDA is smaller than most of that obtained by the previous methods (Li et al. 2010; Dai et al. 2009). Although, there is an elitist strategy (Elitist + RM-MEDA) and a hybrid GA (EA + RM-MEDA) method to help RM-MEDA to get the regularity of PS early in these previous methods, the local information of elitist solutions is still not directly used when sampling the offspring solutions. Furthermore, as only half of population is used to build the probability model in Li et al. (2010), it is every easy for this algorithm to fall into the local Pareto fronts. This can be clearly learned from the result on F3, F6 and F7.

Table 4 The IGD-Metric performance of the four algorithms on F1–F9

IGD-metric	RM-MEDA	Elitist + RM-MEDA	EA + RM-MEDA	IRM-MEDA
F1	0.0057	0.0056	0.0049	0.0040
F2	0.0061	0.0068	0.0047	0.0039
F3	0.0065	0.2149	0.0048	0.0022
F4	0.0463	0.0456	0.0474	0.0417
F5	0.0175	0.0251	0.0060	0.0049
F6	0.2055	0.2255	0.2348	0.0050
F7	0.1980	0.1532	0.1167	0.0372
F8	0.0606	0.0630	0.0543	0.0527
F9	0.0367	0.6004	0.0083	0.0066

Fig. 9 The total IGD-metric of the four algorithms on F1–F9

1→RM-MEDA   **2→ Elitist + RM-MEDA 3→EA + RM-MEDA**   **4 →IRM-MEDA**

(a) The boxplot of F1   **(b)** The boxplot of F2   **(c)** The boxplot of F3

(d) The boxplot of F4   **(e)** The boxplot of F5   **(f)** The boxplot of F6

(g) The boxplot of F7   **(h)** The boxplot of F8   **(i)** The boxplot of F9
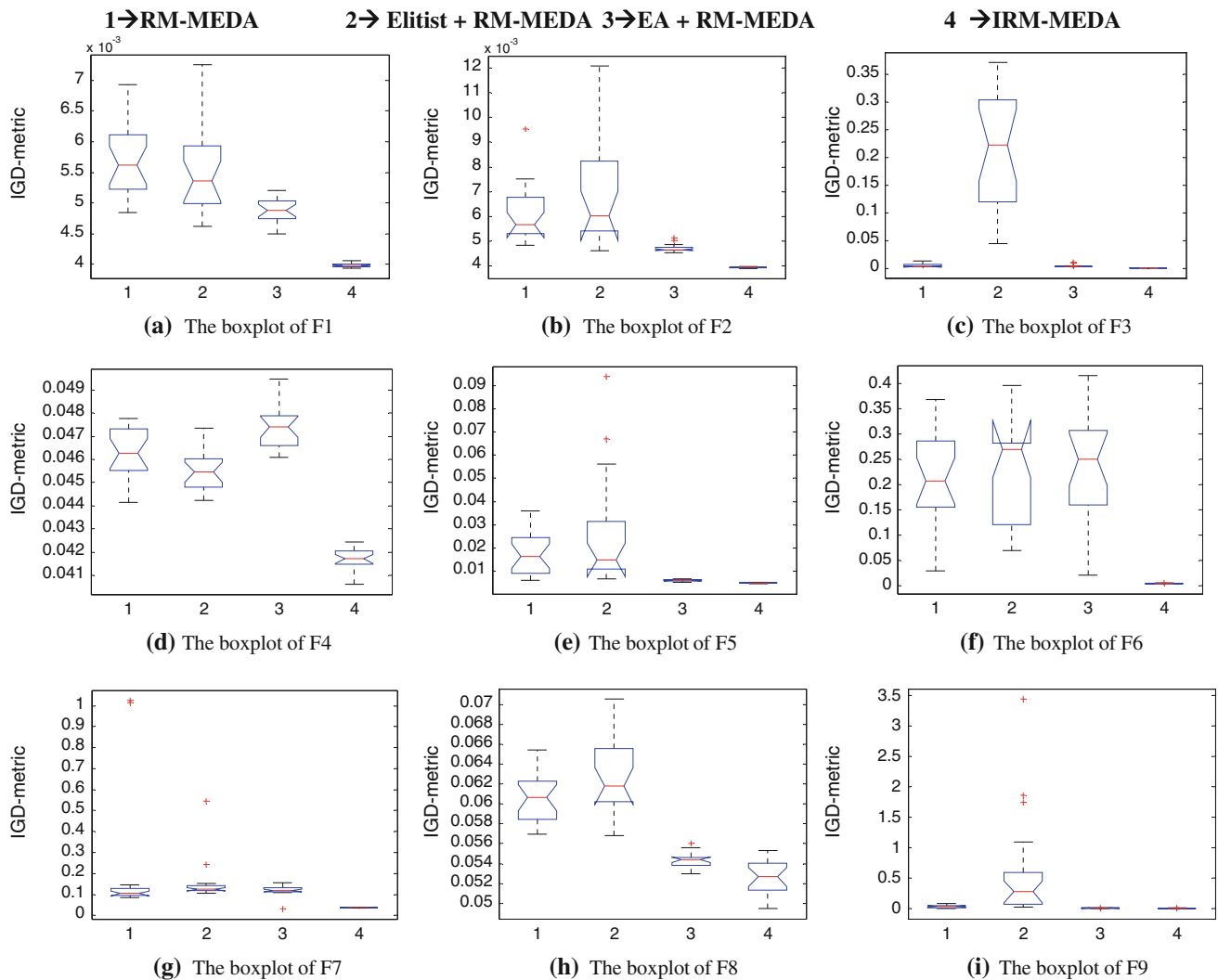
Fig. 10 The statistical IGD-metric of the four algorithms on F1–F9

From Fig. 9, it is obvious that an elitist strategy method (Elitist + RM-MEDA) did very badly on F3 and F9. Just because of this, it got the worst result in the four algorithms. The reason is that the elitist strategy made it fall into local optimization prematurely. But remove the two problems, it has the same result as RM-MEDA. IRM-MEDA did much better than other algorithms on F6 and F7. And on other problems, it still has advantages.

For analyzing the stable performance of IRM-MEDA, statistical IGD-metric of four algorithms of 20 independent runs is obtained, and it is shown by boxplot in Fig. 10. Boxplot is a very popular method in the Genetic Algorithm. It can clearly reflect the maximum, minimum, mean, wild – value and distribution of the samples. The shorter the box, the more stable the method.

From Fig. 10, the stable performance of IRM-MEDA is much higher than other algorithms on F1, F2, F4 F6 and F7.

And on F3 and F5, it still has advantages than a hybrid GA (EA + RM-MEDA). In Fig. 10, a hybrid GA (EA + RM-MEDA) also displays it outperformance than RM-MEDA and an elitist strategy method (Elitist + RM-MEDA).

5 Conclusion

In this paper, local learning has been introduced to improve the local search ability of RM-MEDA. Experimental results show that our algorithm is much better than the traditional RM-MEDA on the convergence capability. When reducing the maximal number of function evaluations, the result of the new algorithm changes very small. It denotes the Local Learning really worked well in this algorithm. The comparison with other two improved versions of RM-MEDA also illustrated the advantages of our algorithm.

However, our algorithm introduced two important parameters which heavily affect the effect of this algorithm. This is worth being on the search of the adaptive of these values. In addition, the algorithm has not a strategy to improve the ability of the global search. This may lead to local optimum. In our future work, we are also interested in introducing other techniques to improve the global search ability of this algorithm.

References

Baluja S (1994) Population-based incremental learning: a method for integrating genetic search based function optimization and competitive learning. Carnegie Mellon Univ, Pittsburgh, Tech Rep CMU-CS-94-163

Baluja S, Davies S (1998) Fast probabilistic modeling for combinatorial optimization. In Proceedings of the national conference on artificial, pp 469–476

BenSaid L, Bechikh S, Ghedira K (2010) The r-dominance: a new dominance relation for interactive evolutionary multicriteria decision making. IEEE Trans Evol Comput 14(5):801–818

Bosman PAN, Thierens D (2005) The naive MIDEA: a base line multi-objective EA. The 3rd international conference on evolutionary multi-criterion optimization (EMO 2005), LNCS, vol 3410. Springer, Berlin, pp 428–442

Coello CAC, Pulido GT (2001) Multiobjective optimization using a micro-genetic algorithm. In: Proceedings of genetic and evolutionary computation conference (GECCO 2001), pp 274–282

Coello CAC, Van Veldhuizen DA, Lamont GB (2002) Evolutionary algorithms for solving multi-objective problems. Kluwer, Dordrecht

Corne DW, Knowles J, Oates M (2000a) The Pareto-envelope based selection algorithm for multiobjective optimization. In: Parallel problem solving from nature (PPSN-VI). Lecture notes in computer science, vol 1917. Springer, Berlin, pp 869–878

Corne DW, Jerram NR, Knowles JD, Oates MJ (2001b) PESA-II: region-based selection in evolutionary multiobjective optimization. In: Proceedings of the genetic and evolutionary computation conference (GECCO 2001), pp 283–290

Dai G, Wang J, Zhu J (2009) A hybrid multiobjective algorithm using genetic and estimation of distribution based on design of experiments. In: IEEE international conference on intelligent computing and intelligent systems (ICIS 2009), vol 1, pp 284–288

De Bonet JS, Isbell CL, Viola P (1997) MIMIC: finding optima by estimating probability densities. Adv Neural Inf Process Syst 9:424–424

Deb K (2001) Multi-objective optimization using evolutionary algorithms. Wiley, Chichester

Deb K, Pratap A, Agarwa S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 6:182–197

Deb K, Thiele L, Laumanns M, Zitzler E (2005) Scalable test problems for evolutionary multiobjective optimization. In: Evolutionary multiobjective optimization, theoretical advances and applications, pp 105–145

Deb K, Sinha A, Kukkonen S (2006) Multi-objective test problems, linkages, and evolutionary methodologies. In: Proceedings of the genetic and evolutionary computation conference (GECCO 2006), pp 1141–1148

Ghoseiria K, Nadjari B (2010) An ant colony optimization algorithm for the biobjective shortest path problem. Appl Soft Comput 10(4):1237–1246

Hernández-Díaz AG, Santana-Quintero LV, Coello CAC, Molina J (2007) Pareto-adaptive epsilon-dominance. Evol Comput 15(4):493–517

Horn J, Nafpliotis N, Goldberg DE (1994) A niched Pareto genetic algorithm for multiobjective optimization. In: Proceeding of the First IEEE conference on evolutionary computation, vol 1, pp 82–87

Ishibuchi H, Murata T (1998) A multi-objective genetic local search algorithm and its application to flowshop scheduling. IEEE Trans Syst Man Cybern Part C Appl Rev 28(3):392–403

Jamuna K, Swarup KS (2012) Multi-objective biogeography based optimization for optimal PMU placement. Appl Soft Comput 12(5):1457–1620

Jaszkiewicz A (2002) Genetic local search for multiple objective combinatorial optimization. Eur J Oper Res 137(1):50–71

Jaszkiewicz A (2003) Do multiple-objective metaheuristics deliver on their promises? A computational experiment on the set-covering problem. IEEE Trans Evol Comput 7(2):133–143

Kambhatla N, Leen TK (1997) Dimension reduction by local principal component analysis. Neural Comput 9:1493–1516

Khan N (2003) Bayesian optimization algorithm for multi-objective and hierarchically difficult problem. University of Illinois at Urbanachampainge

Knowles JD, Corne DW (2000) Approximating the non-dominated front using the Pareto archived evolution strategy. Evol Comput 8:149–172

Knowles JD, Corne DW (2002) M-PAES: a memetic algorithm for multiobjective optimization. In: Proceedings of the congress on evolutionary computation (CEC 2002), pp 325–332

Kukkonen S, Lampinen J (2005) GDE3: the third evolution step of generalized differential evolution. In: Proceedings of the congress on evolutionary computation (CEC 2005), vol 1, pp 443–450

Larrañaga P, Lozano JA (2001) Estimation of distribution algorithms a new tool for evolutionary computation. Kluwer, Dordrecht

Laumanns M, Ocenasek J (2002) Bayesian optimization algorithm for multi-objective optimization. In: Parallel problem solving from nature (PPSN-VII). Lecture notes in computer science, vol 2439, pp 298–307. Springer, Berlin

Laumanns M, Thiele L, Deb K, Zitzler E (2002) Combining convergence and diversity in evolutionary multi-objective optimization. Evol Comput 10(3):263–282

Li H, Zhang Q (2009) Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II. IEEE Trans Evol Comput 12(2):284–302

Li M, Dai G, Zhu (2010) The RM-MEDA based on elitist strategy. In: The 5th international conference on advances in computation and intelligence (ISICA 2010), vol 6382, pp 229–239

Liu D, Tan KC, Goh CK, Ho WK (2007) A multiobjective memetic algorithm based on particle swarm optimization. IEEE Trans Syst Man Cybern Part B Cybern 37(1):42–50

Miettinen K (1999) Nonlinear multiobjective optimization. Kluwer, Dordrecht

Miller BL, Goldberg DE (1995) Genetic algorithms, tournament selection, and the effects of noise. Complex Syst 9(3):193–212

Mühlenbein H, Paass G (1996) From recombination of genes to the estimation of distributions I. Binary parameters. In: Parallel problem solving from nature (PPSN-IV). Lecture notes in computer science, vol 1141, pp 178–187. Springer, Berlin

Okabe T, Jin Y, Sendhoff B, Olhofer M (2004) Voronoi-based estimation of distribution algorithm for multi-objecbive optimization. In: Proceedings of the congress on evolutionary computation (CEC 2004), vol 2, pp 1594–1601

Pelikan M, Goldberg D, Lobo F (2000a) A survey of optimization by building and using probabilistic model. Proc Am Control Conf 5:3289–3293

Pelikan M, Goldberg DE, Cantu-Paz E (2000b) Linkage problem, distribution estimation, and Bayesian networks. Evol Comput 8:311–340

Pelikan M, Sastry K, Goldberg D (2005) Multiobjective HBOA, clustering, and scalability. In: Proceedings of the genetic and evolutionary computation conference (GECCO 2005), pp 663–670

Schütze O, Mostaghim S, Dellnitz M, Teich J (2003) Covering paretosets by multilevel evolutionary subdivision techniques. The 2nd international conference on evolutionary multi-criterion optimization (EMO 2003), LNCS, vol 2632. Springer, Berlin, pp 118–132

Smith J (2007) Co-evolving memetic algorithms: a review and progress report. IEEE Trans Syst Man Cybern Part B Cybern 37(1):6–17

Srinivas N, Deb K (1994) Multiobjective optimization using nondominated sorting in genetic algorithms. Evol Comput 2:221–248

Van Veldhuizen DA, Lamont GB (1998) Evolutionary computation and convergence to a Pareto front. In: Late breaking papers at the genetic programming 1998 conference, pp 221–228

Wang Y, Xiang J, Cai ZX (2012) A regularity model-based multiobjective estimation of distribution algorithm with reducing redundant cluster operator. Appl Soft Comput 12(11):3526–3538

Zhang Q, Sun J, Tsang E (2005) An Evolutionary algorithm with guided mutation for the maximum clique problem. IEEE Trans Evol Comput 9(2):192–200

Zhang Q, Sun J, Xiao G, Tsang E (2007a) Evolutionary algorithms refining a heuristic: a hybrid method for shared-path protections in WDM networks under SRLG constraints. IEEE Trans Syst Man Cybern Part B Cybern 37:51–61

Zhang Q, Zhou A, Jin Y (2007b) Global multiobjective optimization via estimation of distribution algorithm with biased initialization and crossover. In Proceedings of the genetic and evolutionary computation conference (GECCO 2007), pp 617–622

Zhang Q, Zhou A, Jin Y (2008) RM-MEDA: a regularity model-based multiobjective estimation of distribution algorithm. IEEE Trans Evol Comput 12:41–63

Zhou A, Zhang Q, Jin Y, Tsang E, Okabe T (2005) A model-based evolutionary algorithm for bi-objective optimization. In: Proceedings of the congress on evolutionary computation (CEC 2005), pp 2568–2575

Zhou A, Jin Y, Zhang Q, Sendhoff B, Tsang E (2006) Combining model-based and genetics-based offspring generation for multi-objective optimization using a convergence criterion. In: Proceedings of the congress on evolutionary computation (CEC 2006), pp 3234–3241

Zitzler E, Thiele L (1999) Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. IEEE Trans Evol Comput 3:257–271

Zitzler E, Deb K, Thiele L (2000) Comparison of multiobjective evolution algorithms: empirical results. IEEE Trans Evol Comput 8:173–195

Zitzler E, Laumanns M, Thiele L (2002) SPEA2: improving the strength pareto evolutionary algorithm for multi-objective optimization. In Proceedings of the evolutionary methods for design, optimization and control with applications to industrial problems, pp 19–26