Combination of EDA and DE for Continuous Bi-objective Optimization

Aimin Zhou, Qingfu Zhang, Yaochu Jin, Bernhard Sendhoff

2008

Combination of EDA and DE for Continuous Biobjective Optimization

Aimin Zhou, Qingfu Zhang, Yaochu Jin and Bernhard Sendhoff

***Abstract*— The Pareto front (set) of a continuous optimization problem with m objectives is a $(m-1)$ dimensional piecewise continuous manifold under some mild conditions. Based on this property in the decision space, we have recently developed several multiobjective estimation of distribution algorithms. However, the property in the objective space has not yet utilized. In this paper, a simple EDA is proposed to model the Pareto front in the objective space for guiding the selection procedure and model building procedure the in decision space. Since the location information is ignored in EDAs, a combination of EDA and DE is suggested for improving the algorithmic performance. Experimental results has shown that the algorithm with the proposed strategy is very promising.**

I. INTRODUCTION

We consider the following continuous multiobjective optimization problem (MOP):

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

where $X = \prod_{i=1}^{n} [\underline{x}_i, \overline{x}_i] \subset R^n$ is the decision space and $x = (x_1, \dots, x_n)^T \in R^n$ is the decision variable vector. $F : X \rightarrow R^m$ consists of m real-valued continuous objective functions $f_i(x)$ $(i = 1, \dots, m)$. R^m is the objective space.

Since the objectives of a MOP usually conflict with each other, no single solution can minimize all the objectives simultaneously. The Pareto optimal solutions, which characterize the best tradeoffs, need to be found in many real-world applications. The set of all Pareto optimal solutions is called the Pareto set (PS) and its image in the objective space is called Pareto front (PF). Multiobjective evolutionary algorithms (MOEAs) try to find a good approximation of PF [2].

Under certain mild conditions, the PF (PS) of (1) is an $(m-1)$-dimensional manifold embedded in the m-dimensional objective space (the n-dimensional decision space) [4]. However, this property has been ignored by most researchers in the community of evolutionary multiobjective optimization [5]. In recently years, we have systematically studied and applied the regularity property of PSs in designing of MOEAs. A conceptual algorithm, called regularity model based multiobjective estimation of distribution algorithm (RM-MEDA) has been reported in [15]. Local principal component analysis [6] and generative topographic mapping [1], have been used for modelling the distribution of promising solutions [16]. By combing with special designed biased initialization and biased crossover operators, RM-MEDA is enhanced to tackle problems with many local PFs [17]. This regularity property is also applied in dynamic environments to trace the movement of PFs [18]. We have also noticed that a Pareto Path Following (PPF) strategy, proposed by Harada, et al. [3], is also based on the regularity property of PSs. However, the regularity of PFs has not yet exploited. One major purpose of this paper is to study how to make the advantage of the regularity of PFs in designing multiobjective evolutionary algorithms.

Estimation of distribution algorithms (EDAs) extract statistical information from the population for offspring generation and the individual location information is ignored [14]. On the other hand, differential evolution (DE) [11] extracts differential information from the parents for generating trial solutions and does not consider the global population distribution information. Sun et al have advocated combination EDA and DE in [12] and proposed a DE/EDA hybrid algorithm for scalar objective optimization. The second purpose of this paper is to study if combination EDA and DE is beneficial for multiobjective optimization.

Based on the regularity property of PFs in the objective space, a simple EDA method is proposed to model the PF in the objective space for guiding the selection procedure and model building procedure the in decision space. Since the location information is ignored in EDAs, a combination of EDA and DE is suggested for improving the algorithmic performance. We assume in this paper that there are only two objectives.

The remainder of the paper is organized as follows. Section II gives the algorithm framework and presents the details of model building both in the decision space and the objective space. Section III presents the experimental results. The paper is then concluded in Section IV.

II. HYBRID MULTIOBJECTIVE EVOLUTIONARY ALGORITHM

In this section, the framework of a hybrid multiobjective evolutionary algorithm with EDA and DE, called EDA+DE, is firstly presented. The approximation model based selection procedure, probability model building and offspring generating are then introduced in detail.

A. Algorithm Framework

Parameters

A. Zhou and Q. Zhang are with Department of Computing and Electronic Systems, University of Essex, Wivenhoe Park, Colchester, CO4 3SQ, U.K.

Y. Jin and B. Sendhoff are with Honda Research Institute Europe, Carl-Legien-STD. 30, 63073 Offenbach, Germany.

N: population size.
T: maximum number of generations.
K: number of line segments used to approximate the centroid of the population.
CR: crossover probability in DE.
F: parameter used by DE.

Notations

t: generation count.
P_t: population at generation t.
$Q_{1,t}$: offspring created by EDA operator at generation t.
$Q_{2,t}$: offspring created by DE operator at generation t.

EDA+DE

// Initialization
Step 0 Set $t := 0$ and $P_0 := random\_initialize()$.
// Stopping condition
Step 1 If $stop\_condition() = true$, stop and return the nondominated solutions in P_t.
// Offspring Generation
Step 2 $Q_{1,t} := EDA\_generate(P_t, N/2, K)$.
Step 3 $Q_{2,t} := DE\_generate(P_t, N/2, CR, F)$.
// Selection
Step 4 $P_{t+1} := AM\_select(P_t, Q_{1,t} \cup Q_{2,t}, N)$.
Step 5 Set $t := t + 1$ and go to **Step 1**.

In Step 0, the initial population is uniformly randomly sampled from the search space $X \subset R^n$. In Step 1, the algorithm will stop when the generation count t exceeds a predefined maximum value T.

B. Approximation Model based Selection

In the case of two objectives, The PF of a continuous MOP should be a piecewise continuous 1-D curve in the objective space. This property has not been used in current MOEAs. A possible application of this regularity property in evolutionary multiobjective optimization is on selection procedure. Suppose the PF of a MOP is known, the solutions, which approximate the PF properly, can be selected into the next generation. Although its PF is unknown in a real-world problem, a model can be built to approximate the PF based on the regularity property of MOPs.

Based on the above idea, an approximate model based selection (AMS) is proposed in this paper for biobjective problems. AMS works as follows: (a) build a linear model to approximate the current population, (b) uniformly select some points in the linear model as target points, and (c) select those points in current population which are closest to the target points into the next generation.

The procedure of $P_{t+1} := AM\_select(P_t, Q, M)$ works as follows in details.

Step 1 Find two extreme points from P_t by

$$x_i^* = \arg\min_{x \in P_t} \{\alpha_1 f_1(x) + (1.0 - \alpha_2) f_2(x)\},$$

where $i = 1, 2$, $\alpha_1 = 0.95$ and $\alpha_2 = 0.05$.

Step 2 Let $F_i^* = F(x_i^*), i = 1, 2$. Find a line L such that (a) L is parallel to line $F_1^* F_2^*$, (b)no point in L can be dominated by any points in the current population $P_t \cup Q$, and (c) the distance between L and $F_1^* F_2^*$ should be as small as possible.

Step 3 Let A_1 and A_2 be the projections of F_1^* and F_2^* on L. Extend the line segment $A_1 A_2$ along each direction by 25% and obtain the line segment LS. Uniformly sample M target points $R_1, R_2, \ldots, R_M$ in the line segment LS.

Step 4 For $i = 1, \ldots, M$, select the point closest to R_i in the current population $P_t \cup Q$ into the next generation P_{t+1}.

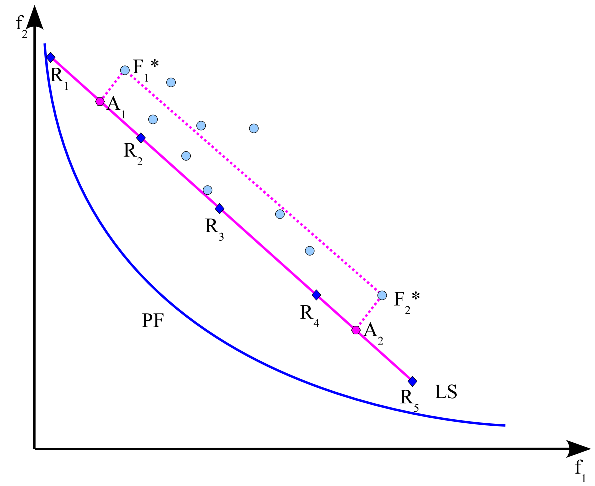

Fig. 1. Illustration of approximation model based selection.

It should be pointed that in Step 1, α_i are set to be 0.95 and 0.05 instead of 1.0 and 0.0. In Step 3, $A_1 A_2$ is extended along the direction. The purpose is to make the approximation model more stable.

C. EDA based Offspring Generator

We will use a set of linear model to approximate the centroid of the population in the decision space as in [15]. The population is firstly partitioned into several clusters. Then in each cluster, the principal component analysis (PCA) is applied to find the central line and the variance. Finally, new trial solutions are sampled from the models.

The procedure $Q := EDA\_generate(P, M, K)$ works as follows.

Step 1 Partition the population P into K clusters by

$$\{P^1, \cdots, P^K\} = partition(P, K).$$

Step 2 For each cluster P^k, build a probability model Φ_k.
Step 3 Sample M new trial solutions from the models built in Step 2 and store them in Q.

In the following, we give the details of the above procedure.

1) Population Partitioning:

$$\{P^1, \cdots, P^K\} = partition(P, K)$$

Let $R_k, k = 1, \ldots, K$ be K uniformly distributed reference points in LS (see Section II-B for details).

For each reference point R_k, select $2|P|/K$ closest points from P to form the k-th cluster P^k, where $|P|$ is the cardinality of P.

2) Probability Model Building: The mean of the k-th cluster P^k is

$$\bar{x}^k = \frac{1}{|P^k|} \sum_{x \in P^k} x.$$

The covariance matrix of P^k is

$$Cov^k = \frac{1}{|P^k|-1} \sum_{y \in P^k} (y - \bar{x}^k)(y - \bar{x}^k)^T.$$

The ith principal component ν_i^k is a unity eigenvector associated with the ith largest eigenvalue λ_i^k of the matrix Cov^k.

The following linear model with Gaussian noise, denoted as Φ_k, is used for modelling the distribution of the points in P^k:

$$\Phi_k : x = \bar{x}^k + \sum_{i=1}^{m-1} s_i \nu_i^k + \varepsilon^k$$

where s_i is a random variable uniformly distributed in $[s_i^{k,min}, s_i^{k,max}]$,

$$s_i^{k,min} = \min_{x \in P^k} (x - \bar{x}^k)^T \nu_i^k$$

and

$$s_i^{k,max} = \max_{x \in P^k} (x - \bar{x}^k)^T \nu_i^k.$$

$\varepsilon^k \sim N(0, \delta^k I)$, I is the identity matrix and

$$\delta^k = \frac{1}{n-m+1} \sum_{i=m}^{n} \lambda_i^k.$$

3) Probability Model Sampling: A new trial solutions x is sampled as follows:

Step 1 A model Φ_k is uniformly randomly select from $\{\Phi_k | k = 1, \ldots, K\}$.

Step 2 Uniformly randomly generate $s_i \in [s_i^{k,min} - 0.25(s_i^{k,max} - s_i^{k,min}), s_i^{k,max} + 0.25(s_i^{k,max} - s_i^{k,min})]$, $i = 1, \cdots, m-1$.

Step 3 Generate a noise vector $\varepsilon^k \sim N(0, \delta^k I)$.

Step 4 Generate a new trial solution $x' = \bar{x}^k + \sum_{i=1}^{m-1} s_i \nu_i^k + \varepsilon^k$.

Step 5 a new trial solution x is generated with boundary checking,

$$x_i = \begin{cases} 0.5(\underline{x}_i + x''_i) & \text{if } x'_i < \underline{x}_i \\ 0.5(\overline{x}_i + x''_i) & \text{if } x'_i > \overline{x}_i \\ x'_i & \text{otherwise} \end{cases},$$

where $i = 1, \ldots, n$ and x'' is randomly selected from P.

D. DE based Offspring Generator

We use the DE [11] operator

$$Q := DE\_generate(P, M, CR, F),$$

with parent population P, parameters CR and F to generate M new trial solutions in this paper.

A new trial solution is generated as the following DE/rand/1 scheme,

Step 1 Randomly select three different parents x^1, x^2, x^2 from population P.

Step 2 Uniformly randomly generate an index $jnd \in \{1, \ldots, n\}$.

Step 3 Generate a temporal solution x',

$$x'_i = \begin{cases} x_i^1 + F(x_i^2 - x_i^3) & rand() < CR \text{ or } i = jnd \\ x_i^1 & otherwise \end{cases},$$

where $i = 1, \ldots, n$ and $rand()$ returns a random number from $[0.0, 1.0]$.

Step 4 A new trial solution x is generated with boundary checking,

$$x_i = \begin{cases} 0.5(\underline{x}_i + x_i^1) & \text{if } x'_i < \underline{x}_i \\ 0.5(\overline{x}_i + x_i^1) & \text{if } x'_i > \overline{x}_i \\ x'_i & \text{otherwise} \end{cases},$$

where $i = 1, \ldots, n$.

III. Experimental Results

To assess the performance of the proposed algorithm EDA+DE, four biobjective problems, S_ZDT1, S_ZDT2 [13] and their variants, S_ZDT11 and S_ZDT21 are tested. The proposed algorithm is also compared with GDE3 [8] and the pure EDA version of the algorithm in which the DE part is removed.

A. Test Instances

The S_ZDT test suite [13] is introduced to overcome the shortcomings of ZDT test suite [2] such as (a) the parameters have the same value for dimension 2 to n, (b) the global optimum lines on the bounds, and (c) the problems are separable. However, S_ZDT have linear linkages between decision variables which may favor some offspring generators [15]. A transformation is added to the $g(\cdot)$ to introduce nonlinear linkages [9] in ZDT test suite. The same strategy is used in this paper by replacing

$$g(x) = 1 + 9(\sum_{i=2}^{D} {z'}_i)/(D-1)$$

with

$$g(x) = 1 + 9[\sum_{i=2}^{D} ({z'}_i - {z'}_0^5)^2]/(D-1).$$

The modified S_ZDT1 and S_ZDT2 are called S_ZDT11 and S_ZDT21 respectively. The details of S_ZDT test suite are referred to [13].

B. Performance Metrics

The inverted general distance is used in assessing the performance of the algorithms.

Let P^* be a set of uniformly distributed points in the objective space along the PF. Let P be an approximation to the PF, The general distance from P^* to P is defined as:

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

where $d(v, P)$ is the minimum Euclidean distance between v and the points in P. If $|P^*|$ is large enough to represent the PF very well, $D(P^*, P)$ could measure both the diversity and convergence of P in a sense. To have a low value of $D(P^*, P)$, P must be very close to the PF and cannot miss any part of the whole PF.

In our experiments, we select 1000 evenly distributed points on PF and let these points to be P^* for each test instance.

Another indicator called difference of hypervolume (I_H^-) [7] is also tried. Since the results are very consistent with those of the D-metric, not all results with I_H^- are reported in this paper.

C. Algorithm Parameters

Three algorithms, the hybrid one (denoted as EDA+DE), the pure EDA version of which the offspring is only generated by EDA operator in Section II, and GDE3 [1].

For all test instances, the variable dimension is 30. The population size is 200, and the maximum generation is 500. The test results are based on 20 independent runs.

The parameters for EDA+DE are as follows: number of line segments used to approximate the centroid of the population, K, is 5 in EDA offspring generator, and in DE offspring generator, $CR = 0.1$ and $F = 0.5$ as suggested in [8].

The parameters for EDA and GDE3 are the same as in EDA+DE.

D. Results and Analysis

TABLE I

STATISTICAL RESULTS($mean \pm std.$)

		EDA+DE	EDA	GDE3
S_ZDT1	D	**0.0033**±0.0001	0.0261±0.0096	0.0049±0.0001
	I_H^-	**0.0108**±0.0019	0.1420±0.0423	0.0202±0.0017
S_ZDT2	D	0.0340±0.1355	0.7125±0.0252	**0.0051**±0.0002
	I_H^-	0.1747±0.7436	4.0050±0.1393	**0.0261**±0.0026
S_ZDT11	D	**0.0044**±0.0004	**0.0043**±0.0002	0.0366±0.0051
	I_H^-	**0.0127**±0.0029	**0.0119**±0.0020	0.4602±0.0208
S_ZDT21	D	**0.0114**±0.0009	0.0128±0.0009	0.0621±0.0063
	I_H^-	**0.0164**±0.0015	0.0194±0.0018	1.3790±0.0773

The statistical values of D-metric and I_H^- on the results obtained in the final runs are shown in Table I. The best run according to D-metric is shown in Fig. 2 and the final results of all runs are shown in Fig. 3.

[1]It is implemented by ourselves in C++.

1) S_ZDT1: S_ZDT1 has a convex PF and its PS is a line segment. Although EDA+DE has the best performance both on D-metric and I_H^-, from Figs. 2-3, we can see that both EDA and GDE3 can have good approximations after 500 generations although GDE3 is slight better than EDA.

2) S_ZDT2: S_ZDT2 has a concave PF and its PS is a line segment. GDE3 is slight better than EDA+DE. However, EDA converges to a single point in all runs. The reason might be that the AMS selection strategy mislead the search process.

3) S_ZDT11: S_ZDT11 is a modified version of S_ZDT1 of which the PS is a curve. The performances of EDA+DE and EDA are quite similar although EDA is slightly better. For GDE3, we can see that no single run can cover the whole PF.

4) S_ZDT21: S_ZDT21 is a modified version of S_ZDT2 of which the PS is a curve. The results are quite similar to those of S_ZDT11 except that EDA+DE is slightly better than EDA.

Fig. 4 shows the evolution of the average metric values of the population with the number of generations in the three algorithms. For S_ZDT1, the performance of EDA+DE is similar to that of GDE3 and for S_ZDT11 and S_ZDT21, the performances of EDA+DE are similar to those of EDA.

From these test results, we can conclude that by hybridizing EDA and DE offspring generators, EDA+DE works well for the four test instances in this paper.

IV. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a new algorithm, EDA+DE, for continuous multiobjective optimization. Based on the regularity property of PFs, an approximation model based selection (AMS) was designed to (a) select solutions into the next generation, and (b) guide the model building process in the offspring generation. To utilize both the population statistical information and individual location information, an EDA and DE combination was introduced.

The proposed EDA+DE method was compared with GDE3 and pure EDA method on four test instances with linear or nonlinear linkages among decision variables. The experimental results have shown that EDA+DE is more stable than GDE3 and EDA on these test problems.

In this paper, only a line segment is used to approximate the PF of a biobjective problem. This line segment might not be a very good choice if the PF of a biobjective problem is a nonlinear curve. On the other hand, the PFs of problems with more than two objectives are rather complicated. How to approximate these complicated PFs to guide the selection procedure is an interesting topic for future research. The PSs of the problems used in this paper are rather simple, and the geometric shapes are lines or curves. It is worth to improve the proposed method and test it on problems with complicated PSs [10] in the future.

REFERENCES

[1] Christopher M. Bishop, Markus Svensén, and Christopher K. I. Williams. GTM:the generative topographic mapping. *Neural Computation*, 10(1):215–234, 1998.

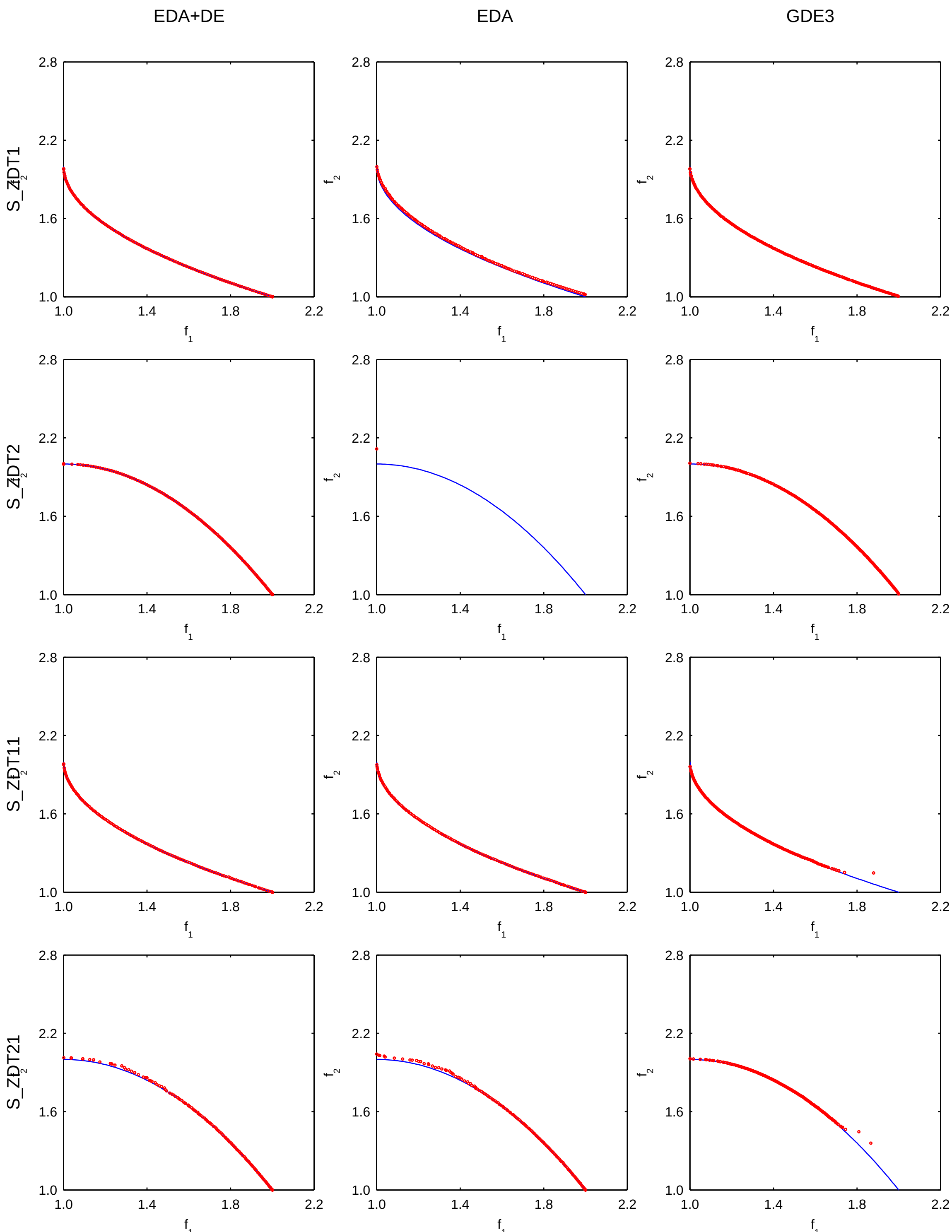

Fig. 2. The final nondominated fronts with the lowest D-metric found by the algorithms.

[2] Kalyanmoy Deb. *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley & Sons, LTD, Baffins Lane, Chichester, 2001.

[3] Ken Harada, Jun Sakuma, and Shigenobu Kobayashi. Uniform sampling of local pareto-optimal solution curves by pareto path following and its applications in multi-objective ga. In *Proceedings of the conference on Genetic and evolutionary computation (GECCO 2007)*, pages 813–820, London, England, United Kingdom, 2007. ACM Press.

[4] Claus Hillermeier. *Nonlinear Multiobjective Optimization - A Generalized Homotopy Approach*. Birkhäuser, 2001.

[5] Yaochu Jin and Bernhard Sendhoff. Connectedness, regularity and the success of local search in evolutionary multi-objective optimization. In *Proceedings of the Congress on Evolutionary Computation (CEC 2003)*,

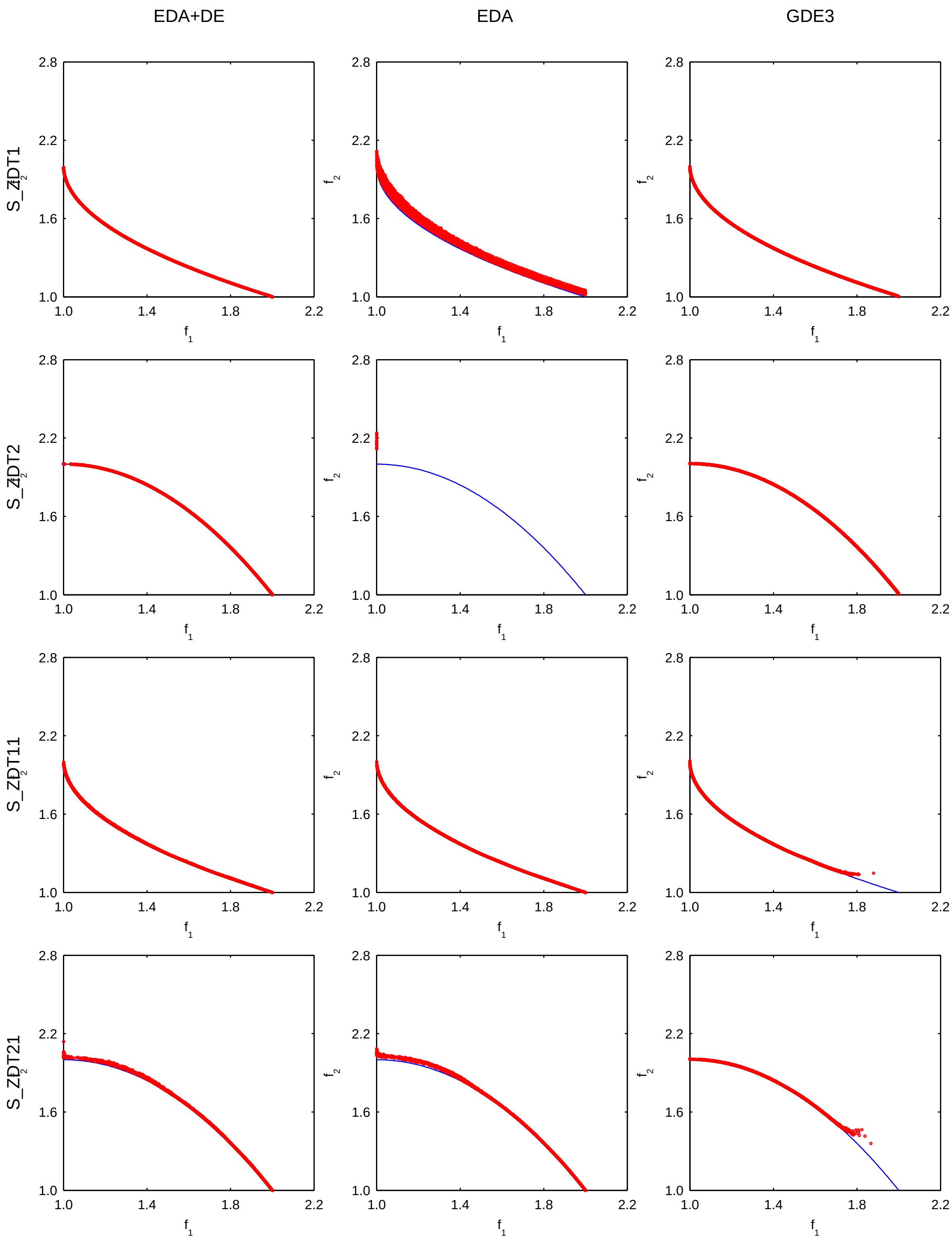

Fig. 3. The final 20 nondominated fronts found by the algorithms.

pages 1910–1917, Canberra, Australia, December 2003. IEEE Press.

[6] Nandakishore Kambhatla and Todd K. Leen. Dimension reduction by local principal component analysis. *Neural Computation*, 9(7):1493–1516, October 1997.

[7] Joshua D. Knowles, Lothar Thiele, and Eckart Zitzler. A tutorial on the performance assessment of stochastic multiobjective optimizers. Technical Report 214, Computer Engineering and Networks Laboratory, ETH Zurich, Gloriastrasse 35, 8092 Zurich, Switzerland, 2006.

[8] Saku Kukkonen and Jouni Lampinen. Performance assessment of generalized differential evolution 3 (GDE3) with a given set of problems. In *Proceedings of the Congress on Evolutionary Computation (CEC 2007)*, pages 3593–3600, Singapore, 2007. IEEE Press.

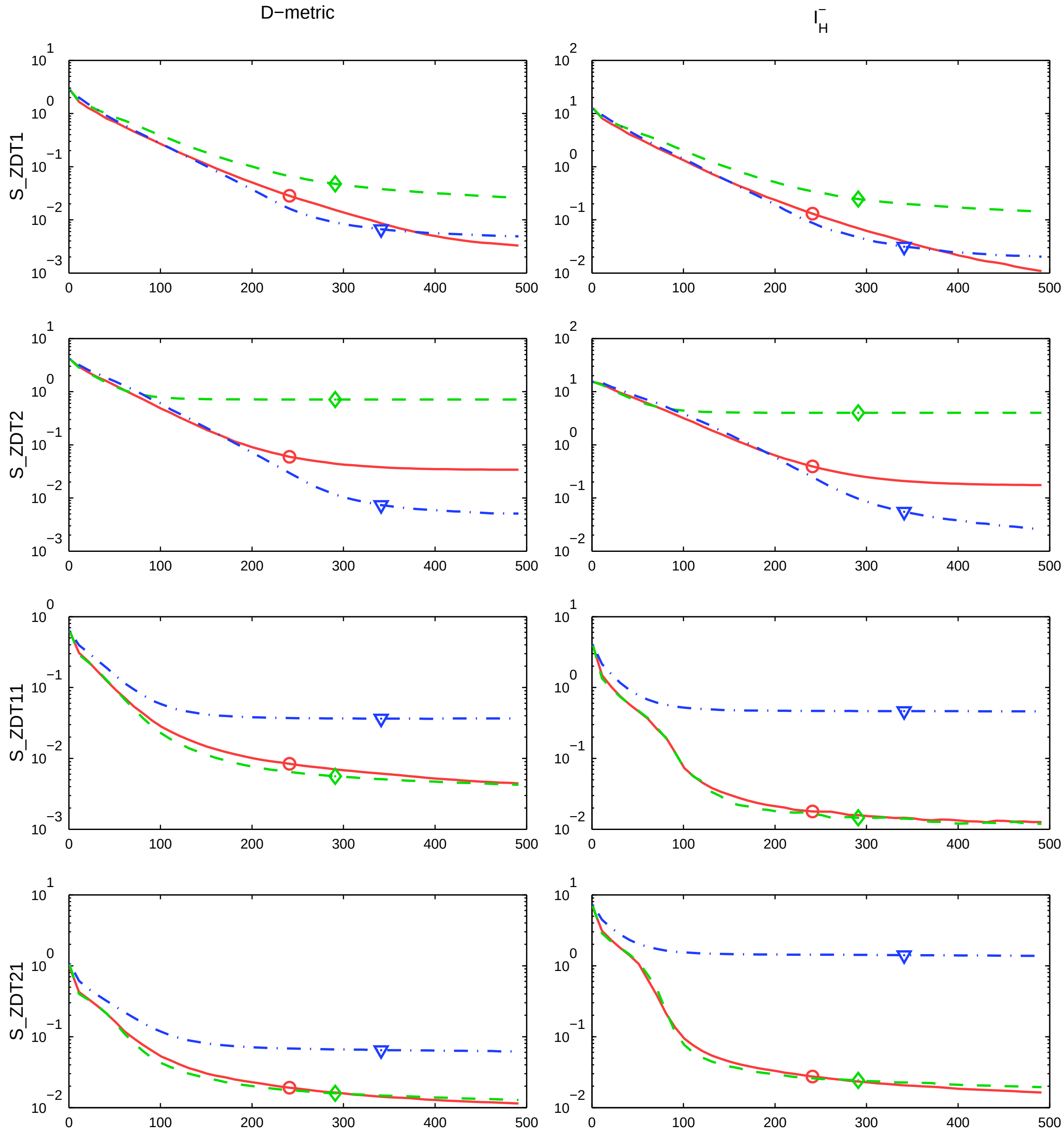

Fig. 4. The evolution of metrics vs. the number of generations. (The solid line is for EDA+DE, dash line for EDA and dot dash line for GDE3.)

[9] Hui Li and Qingfu Zhang. A multiobjective differential evolution based on decomposition for multiobjective optimization with variable linkages. In *Parallel Problem Solving from Nature (PPSN IX)*, volume 4193 of *Lecture Notes in Computer Science*, pages 583–592, Reykjavik, Iceland, 2006. Springer.

[10] Hui Li and Qingfu Zhang. Comparison between nsga-ii and moea/d on a set of multiobjective optimization problems with complicated pareto sets. Technical Report CES-476, University of Essex, 2007.

[11] Kenneth V Price. *New Ideas in Optimization*, chapter An Introduction to Differential Evolution, pages 79–108. McGraw-Hill, 1999.

[12] Jianyong Sun, Qingfu Zhang, and Edward Tsang. DE/EDA: A new evolutionary algorithm for global optimization. *Information Sciences*, 169(3):249–262, 2005.

[13] K. Deb E. Zitzler P. N. Suganthan J. J .Liang M. Preuss V. L. Huang, A. K. Qin and S. Huband. Problem definitions for performance assessment of multi-objective optimization algorithms. Technical report, Nanyang Technological University, Singapore, 2007.

[14] Qingfu Zhang, Jianyong Sun, and Edward Tsang. Evolutionary algorithm with the guided mutation for the maximum clique problem. *IEEE Transactions on Evolutionary Computation*, 9(2):1–9, April 2005.

[15] Qingfu Zhang, Aimin Zhou, and Yaochu Jin. RM-MEDA: A regularity model based multiobjective estimation of distribution algorithm. *IEEE Transactions on Evolutionary Computation*, 2007. in press.

[16] Aimin Zhou, Qingfu Zhang, Yaochu Jin, Bernhard Sendhoff, and Edward Tsang. Modelling the population distribution in multi-objective optimization by generative topographic mapping. In *Parallel Problem Solving From Nature (PPSN IX)*, volume 4193 of *Lecture Notes in Computer Science*, pages 443–452, Reykjavik, Iceland, September 2006.

Springer-Verlag.

[17] Aimin Zhou, Qingfu Zhang, Yaochu Jin, Bernhard Sendhoff, and Edward Tsang. Global multiobjective optimization via estimation of distribution algorithm with biased initialization and crossover. In *Proceedings of the conference on Genetic and evolutionary computation (GECCO 2007)*, pages 617–622, London, July 2007.

[18] Aimin Zhou, Qingfu Zhang, Yaochu Jin, Bernhard Sendhoff, and Edward Tsang. Prediction-based population re-initialization for evolutionary dynamic multi-objective optimization. In *Fouth International Conference on Evolutionary Multi-Criterion Optimization (EMO 2007)*, volume 4403 of *Lecture Notes in Computer Science*, pages 832–846, Matsushima,Japan, March 2007. Springer.