

Hybrid Multiobjective Estimation of Distribution Algorithm by Local Linear Embedding and an Immune Inspired Algorithm

Dongdong Yang, Licheng Jiao, *Senior Member, IEEE*, Maoguo Gong, *Member, IEEE*, and Hongxiao Feng

Abstract—A novel hybrid multiobjective estimation of distribution algorithm is proposed in this study. It combines an estimation of distribution algorithm based on local linear embedding and an immune inspired algorithm. Pareto set to the continuous multiobjective optimization problems, in the decision space, is a piecewise continuous $(m-1)$ -dimensional manifold, where m is the number of objectives. By this regularity, a local linear embedding based manifold algorithm is introduced to build the distribution model of promising solutions. Besides, for enhancing local search ability of the EDA, an immune inspired sparse individual clone algorithm (SICA) is introduced and combined with the EDA. The novel hybrid multiobjective algorithm, named HMEDA, is proposed accordingly. Compared with three other state-of-the-art multiobjective algorithms, this hybrid algorithm achieves comparable results in terms of convergence and diversity. Besides, the tradeoff proportions of EDA to SICA in HMEDA are studied. Finally, the scalability to the number of decision variables of HMEDA is investigated too.

I. INTRODUCTION

In real-world optimization applications, it is often necessary to optimize multiple objectives in one problem synchronously. The simultaneous optimization of multiple objectives is different from single objective optimization in that there is no unique solution to MOPs. A set of optimal tradeoff solutions known as the Pareto-optimal solutions are required. During the past two decades, evolutionary algorithms (EAs) have been recognized to be well suited to multiobjective optimization since early in their development because they can be suitably applied to deal simultaneously with a set of possible solutions. A number of multiobjective evolutionary algorithms (MOEAs) have been developed for MOPs [1], [2], [3]. Recently, the multiobjective estimation of distribution algorithms (EDAs) has attracted much interest from MO researchers [4]. The estimation of distribution algorithms (EDAs) are a new computing paradigm in evolutionary computation [1]. A posterior

probability distribution model based on globally statistical information from the selected solutions is built to generate new solutions for next generation. This new type of algorithms replaces the crossover and mutation operators in traditional genetic algorithms by learning and sampling the probability distribution of the promising individuals at per iteration. Working in such a way, the relationships between the variables involved in the problem could be captured and exploited. Among current multiobjective EDAs [1] [4], the regularity model-based multiobjective EDA (RM-MEDA) [1] solves MOPs unconventionally and shows good performance. RM-MEDA uses local principal component analysis algorithm to build the probability model. New individuals are sampled from the model.

As we know, almost all the EDAs build probability distribution model by globally statistical information, therefore, it seems that they do not care about how to take advantage of the independent behavior of each individual. It is interesting that if we hybridize EDAs with individual behavior oriented algorithms. In recent years, artificial immune systems (AIS) have received significant amount of interest from researchers and industrial sponsors. Applications of AIS include such areas as machine learning, fault diagnosis, computer security, and optimization [5], [6]. Recently, immune inspired multiobjective algorithms have been proposed and obtain good performance [7] [8].

In this study, we design a hybrid multiobjective EDA combining local linear embedding and an immune inspired algorithm. Local linear embedding (LLE) is a manifold learning algorithm [9]. Here, it is used to build the statistical model in the manifold space. Later, some of individuals are generated from the manifold model. Besides, the rest individuals are produced by an immune inspired sparse individual clone algorithm (SICA). The LLE based model is built by global statistical information, while SICA emphasizes the nondominated individuals in the sparse regions. Therefore, hybridization of the two methodologies is interesting to the optimization process. Experiments demonstrating the effectiveness of the hybridization scheme are provided for several state-of-the-art multiobjective problems.

The remainder of this paper is organized as follows: Section II describes related background including the definitions of multiobjective optimization and local linear embedding. Section III describes the main loop of HMEDA. In Section IV, experimental study is presented and experimental results are analyzed and summarized. Finally,

This work was supported by the National Natural Science Foundation of China (Grant Nos. 60703107, 60703108), the National High Technology Research and Development Program (863 Program) of China (Grant No. 2009AA12Z210), the National Basic Research Program (973 Program) of China (Grant No. 2006CB705700), the Program for New Century Excellent Talents in University, and the Program for Cheung Kong Scholars and Innovative Research Team in University (Grant No. IRT0645).

All authors are with the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China, Institute of Intelligent Information Processing, Xidian University, Xi'an 710071, China.

Dongdong Yang is the corresponding author. Phone: +86-029-88209786; fax: +86-029-88201023; e-mail: ddyang@mail.xidian.edu.cn.

concluding remarks are presented.

II. RELATED BACKGROUND

A. Multiobjective Optimization

In this study, the following continuous MOPs are considered [10] [11].

$$\min \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x}))^T \quad (1)$$

subject to $\mathbf{x} \in \Omega$ Where \mathbf{x} is a decision variable vector and Ω is a continuous search space. $\mathbf{F} : \mathbf{x} \rightarrow R^m$ is the map of decision variable space to m real valued objective space. The objectives in a MOP usually conflict each other and no single solution can optimize all the objectives at the same time. The Pareto front/set are set of all the optimal tradeoff solutions in the decision/objective space.

There is a vector $\mathbf{u} = (u_1, u_2, \dots, u_m)$, which is said to dominate $\mathbf{v} = (v_1, v_2, \dots, v_m)$ (denoted by $\mathbf{u} \prec \mathbf{v}$), if and only if \mathbf{u} is partially less than \mathbf{v} , which is equal to the following expression:

$$\forall i \in \{1, \dots, m\}, u_i \leq v_i \wedge \exists j \in \{1, \dots, m\} : u_j < v_j \quad (2)$$

We say that a vector of decision vector $\mathbf{x}^* \in \Omega$ is Pareto optimal or nondominated solution if there does not exist another $\mathbf{x} \in \Omega$ such that $\mathbf{x} \prec \mathbf{x}^*$. All the Pareto optimal solutions in the decision space are made up of the Pareto optimal set.

$$\mathbf{P}^* = \{\mathbf{x}^* \in \Omega / \neg \exists \mathbf{x} \in \Omega, \mathbf{x} \prec \mathbf{x}^*\} \quad (3)$$

The corresponding image of the Pareto-optimal set in the objective space is called the Pareto-optimal front, which could be denoted as

$$\mathbf{PF} = \left\{ \mathbf{F}(\mathbf{x}^*) = (f_1(\mathbf{x}^*), f_2(\mathbf{x}^*), \dots, f_k(\mathbf{x}^*))^T / \mathbf{x}^* \in \mathbf{P}^* \right\} \quad (4)$$

All multiobjective optimization algorithms are to find solutions as close to the Pareto-optimal front as possible and to make them as diverse as possible in the obtained nondominated front.

B. Dimensionality Reduction Algorithms

It can be induced from the Karush-Kuhn-Tucker condition that the Pareto set of a continuous MOP defines a piecewise continuous $(m-1)$ -dimensional manifold in the decision space, where m is the number of objectives [1]. That is to say, the Pareto set of a continuous biobjective optimization problem is a piecewise continuous curve in n -dimensional decision space, and so on. Therefore, the individuals could be modeled by manifold learning algorithms since they submit to manifold distribution.

The classical techniques for dimensionality reduction, including principal component analysis (PCA) [12] and multidimensional scaling (MDS) [13], are simple to implement, efficiently computable, and guaranteed to discover the true structure of data lying on or near a linear subspace of the high-dimensional space. In PCA, one performs an orthogonal transformation to the basis of correlation eigenvectors and projects onto the subspace

spanned by those eigenvectors corresponding to the largest eigenvalues. Classical MDS finds an embedding that preserves the inter-points distances, equivalent to PCA when those distances are Euclidean. However, such algorithms often fail when nonlinear structure cannot be regarded as a perturbation from a linear approximation.

More important, several approaches have been devised to address the nonlinear dimensionality reduction, such as LLE [9] and ISOMAP [14]. LLE attempts to preserve the local geometry of the data; essentially, they seek to map nearby points on the manifold to nearby points in the low-dimensional representation. Isomap attempts to preserve geometry at all scales. In this study, LLE is employed to build the statistical model of promising individuals because of its optimization not involving local minima, simply to implement and representational capacity.

LLE algorithm has been succeeded in identifying the underlying structure of the complicated manifold. In this study, the individuals and their neighbors lie on or close to a locally linear patch of the manifold. The local geometry of these patches can be characterized by linear coefficients and all the linear coefficients could be used to reconstruct each data in low dimensional manifold space. Reconstruction errors are measured by following cost function:

$$\theta(W) = \sum_{i=1}^N \left| X_i - \sum_{j=1}^k W_{ij} X_j \right|^2 \quad (5)$$

where X is the N individuals with dimensionality n , k is the number of nearest neighbors of each individuals and W_{ij} is the reconstruction weights, which reflect intrinsic geometric properties of the data. Therefore, the high dimensional data could be mapped to a low dimensional vector Y by the reconstruction weights. This could be done by finding Y to minimize the embedding cost function:

$$\Phi(Y) = \sum_{i=1}^N \left| Y_i - \sum_{j=1}^k W_{ij} Y_j \right|^2 \quad (6)$$

The basic LLE algorithm can be summarized into the following framework and more details of this algorithm could be found in [9].

Algorithm 1: Local Linear Embedding

Input:

\mathbf{X} (N real vectors with D dimensionality)
 d (dimension of embedding manifold)
 K (number of neighbors of each vector)

Output:

\mathbf{Y} (embedding manifold vectors)

Step1: Find the neighbors of each data, X_i .

Step2: Compute the weights W_{ij} that best reconstruct each data point X_i from its neighbors, minimizing the cost equation (5).

Step3: Compute the vectors Y_i by the reconstructed weights W_{ij} , minimizing the equation (6).

III. HYBRID MULTIOBJECTIVE EDA COMBINING LLE AND SICA

A. Sparse Individuals Clonal Algorithm (SICA)

In this study, we introduced a sparse individual clonal algorithm that pays more attention to the less-crowded regions of the current tradeoff front. As the literature [8], we can select only minority isolated nondominated individuals in population. The selected individuals are then cloned proportionally to their crowding-distance values [15] before heuristic search. By latter experimental study, we can obtain that the scheme is beneficial to HMEDA. The locally linear embedding based EDA aims at global statistical information, while SICA aims at searching the local sparse regions of the current tradeoff front. Therefore, hybridizing the two methods can be implemented to maintain a balance between exploration and exploitation.

The crowding-distance is proposed in [15] to measure the density of solutions surrounding a particular solution in the population. In each objective direction, the solutions are first sorted in the ascending order of objective value. Thereafter, for each solution an objective-wise crowding distance is assigned equal to the difference between normalized objective values of the neighboring solutions. The final crowding distance of a solution is equal to the sum of the crowding distances from all objectives.

The proportional cloning is proposed in [8], and it could be denoted as follows:

$$d_i = \left\lceil N_1 \times \frac{r_i}{\sum_{i=1}^e r_i} \right\rceil \quad (7)$$

where N_1 is the number of individuals generated by SICA, $d_i, i=1,2,\dots,e$ is the cloning number assigned to i th individual. By the normalized crowding distances of current nondominated solutions, half ones with large values are selected to form an active set, $\mathbf{A} = (a_1, a_2, \dots, a_e)$. e is the size of \mathbf{A} . r_i is the normalized crowding distance for i th solution in the active set. The cloning operator \mathbf{C} on active set could be described as followings.

$$\begin{aligned} \mathbf{C}(a_1, a_2, \dots, a_e) &= \{\mathbf{C}(a_1), \mathbf{C}(a_2), \dots, \mathbf{C}(a_e)\} \\ &= \left\{ \underbrace{\{a_1, a_1, \dots, a_1\}}_{d_1}, \underbrace{\{a_2, a_2, \dots, a_2\}}_{d_2}, \dots, \underbrace{\{a_e, a_e, \dots, a_e\}}_{d_e} \right\} \end{aligned} \quad (8)$$

The procedure of SICA is described as follows.

Algorithm 2: Spare Individual Clone Algorithm

Input:

\mathbf{D}_t (nondominated individuals at current population)

N_1 (the number of individuals generated by SICA)

Output:

\mathbf{LD}_t (Cloned individuals)

Step1: Normalize crowding distance: find the maximum and minimum crowding distance of \mathbf{D}_t and divide all the solutions' crowding distance by the difference of maximum and minimum.

Step2: Selection: Sorting the current solutions in \mathbf{D}_t by their normalized crowding distances. Select half solutions with large the normalized crowding distances and the selected solutions form \mathbf{A} .

Step3: Proportional Clone: Clone each solution by equation (8) and All the cloned solutions form \mathbf{LD}_t .

Step4: Recombination and mutation: Perform simulated binary crossover on solutions randomly selected from \mathbf{LD}_t and \mathbf{A} . Implement polynomial mutation on \mathbf{LD}_t .

B. The Local Linear Embedding based Model

Several EDAs have been proposed for MOPs. Khan et al. [15] proposed multiobjective Bayesian optimization algorithm (mBOA) and multiobjective hierarchical Bayesian optimization algorithm (mhBOA) by combining the model building and model sampling procedures of Bayesian optimization algorithm (BOA) [17] and hierarchical Bayesian optimization algorithm (hBOA) [18] with the selection procedure of NSGA-II. Laumanns and Ocenasek [19] combined BOA with the selection and replacement procedures of SPEA2. Ahn [20] combined real-coded BOA with selection procedure of NSGA-II with a sharing intensity measure and a modified crowding distance scheme. Zhang et al. proposed regularity model-based multiobjective EDA and obtained good performance [1].

In this study, by the regularity that Pareto set of a continuous MOP defines a piecewise continuous $(m-1)$ -dimensional manifold in the decision space, the individuals and their neighbors lie on or close to a locally linear patch of the manifold. The local geometry of these patches can be characterized by linear coefficients and all the linear coefficients could be used to reconstruct each data in low dimensional manifold space. Therefore, once the compact representation of individuals' variability is found by equation (5), we could sample the manifold distribution evenly and reconstruct the individuals in original high dimensional space. The following model could be used to describe the reconstructed individuals ξ .

$$\xi = \zeta + \varepsilon \quad (9)$$

where ζ is the individuals by reconstructing the uniformly sampled individuals in low dimensional manifold space, and ε is an n -dimensional zero-mean noise vector.

Once the embedding vectors \mathbf{Y} with the size of N and dimensionality of $(m-1)$ are found, we can build the statistical model in manifold space as follows:

$$\mathbf{Z} = \mathbf{OS}(\mathbf{a}, \mathbf{b}), \quad \mathbf{a} = (a_1, a_2, \dots, a_{m-1}), \quad \mathbf{b} = (b_1, b_2, \dots, b_{m-1}) \quad (10)$$

$$a_i = \min_{j \in N} (Y_j^i), b_i = \max_{j \in N} (Y_j^i) \quad i = 1, \dots, (m-1), \quad j = 1, \dots, N$$

where $\mathbf{OS}(\mathbf{a}, \mathbf{b})$ denotes orthogonal sampling in the space, which is spanned by the vector \mathbf{a} and \mathbf{b} . For the limited space of the paper, the orthogonal sample method is not described here. Please refer to the literature [9] for more details. Afterward, these solutions \mathbf{Z} are denoted by their neighbor solutions in \mathbf{Y} by equation (5), and the reconstruction weights \mathbf{RW} could be obtained accordingly. Since the one-to-one correspondence between original solutions \mathbf{X} and embedding ones \mathbf{Y} , the neighbor relations in \mathbf{Y} could be transferred to original solutions \mathbf{X} . By the reconstruction weights, offspring could be reconstructed by the followings.

$$\xi_j = \left\{ \xi_j / \xi_j = \sum_{i=1}^k \mathbf{RW}_{ji} \bar{X}_i + N(0, \sigma_j I), j = 1, 2, \dots, N \right\} \quad (11)$$

$$\sigma_j = \frac{1}{k - (m-1)} \sum_{i=m}^k \lambda_i^j$$

where λ_i^j is the i th largest eigenvalue of covariance matrix of the reconstruction solutions of solutions j . $N(0, \sigma_j I)$ is the Gaussian distribution with the mean of zero and the standard deviation of σ_j . The main loop of HMEDA is as follows.

Algorithm 3: Hybrid Multiobjective Estimation of Distribution Algorithm

Input:

G_{\max} (maximum number of generations)
 N (size of Population)
 N_E (size of subpopulation generated by EDA)
 N_I (size of subpopulation generated by SICA)

Output:

$\mathbf{D}_{G_{\max}+1}$ (final Approximate Pareto-optimal Sets)

Step1: Initialization: Generate an initial population \mathbf{B}_0 with the size N . Set $t=0$.

Step2: Update Dominant Population: Identify dominant solutions in \mathbf{B}_{t+1} ; Copy all the dominant solutions to form the temporary dominant population \mathbf{TD}_{t+1} ; If the size of \mathbf{TD}_{t+1} (denoted by TN) is larger than N , calculate the crowding distance values of all individuals in \mathbf{TD}_{t+1} and sort them in descending order of crowding distance, delete the last $TN-N$ individuals in \mathbf{TD}_{t+1} .

Step3: Termination: If $t \geq G_{\max}$ is satisfied, export \mathbf{TD}_{t+1} as the output of the algorithm, Stop; Otherwise, $t=t+1$.

Step4: Spare individual clone algorithm: Let $\mathbf{D}_t = \mathbf{TD}_{t+1}$. Algorithm 2 is employed and parts of new offspring are generated.

Step5: Local linear Embedding: Let $\mathbf{X} = \mathbf{B}_t$, $d=m-1$. Algorithm 1 is adopted to find the embedding vectors \mathbf{Y} . Each solution of \mathbf{X} is mapped to that of \mathbf{Y} in manifold space.

Step6: Build Statistical Model in Manifold Space: By the equation (10), N solutions are used to find the vector \mathbf{a} and \mathbf{b} . N_E solutions \mathbf{Z} are orthogonally sampled by the model.

Step7: Compute the Reconstruction Weights: Find the neighbors of \mathbf{Z} by the solutions in \mathbf{Y} . Compute the weights \mathbf{RW}_{ij} that best denote each data point \mathbf{Z}_i from its neighbors, $Y_i = \{y_i^1, y_i^2, \dots, y_i^k\}$, minimizing the cost equation (5).

Step8: Reconstruction: By Step 5, each \mathbf{Y}_i maps to a solutions in \mathbf{X}_i . By the reconstruction weights, the solution ξ_i is reconstruction in equation (11).

Step9: Hybridization: Hybridize the solutions generated by LLE and SICA. New combined solutions $\mathbf{O}_t = \xi_i \cap \mathbf{LD}_t$. Evaluate \mathbf{O}_t .

Step10: Selection by Nondominated sorting: Nondominated sorting based selection is used to choose solutions with size N from the combined \mathbf{O}_t and \mathbf{B}_t . \mathbf{B}_{t+1} is updated by these selected solutions; go to Step2.

C. The Characteristics of HMEDA

It can be seen that HMEDA uses two kinds of optimization methodologies. The EDA is based on a well-known manifold learning algorithm, local linear embedding. It explicitly extracts globally information from the current population and builds an orthogonal sample model in the manifold space. The EDA is expected to find the common intrinsic characteristics of current population, however, the independent behavior of promising individuals may be lost. In order to offset the shortcoming of EDA, a sparse individual clonal algorithm that pays more attention to the less-crowded regions of the current tradeoff front is introduced to the EDA. Some of the offspring are generated by LLE, while the rest ones are from SICA. Therefore, a tradeoff between the exploration ability of the EDA and the exploitation ability of SICA is maintained, which is often crucial to the success of the search and optimization process.

D. Computational Complexity of HMEDA

In this section, we will present the computational complexity of HMEDA. Assuming the size of population is N , and m is the number of objectives. The size of active population is e , the clone population size is N_I , and the maximum number of nearest neighbors is k . The time complexity of one generation for the algorithm can be

calculated as follows:

In **Algorithm 3**, the worst time complexity of LLE is $\mathcal{O}(mN^2)$; computing nearest neighbors scales as $\mathcal{O}(mN^2)$ in the worst case; the worst time complexity of computing the reconstruction weights is $\mathcal{O}(nNK^3)$; the time complexity of nondominated sorting is $\mathcal{O}(mN^2)$.

In **Algorithm 2**, the time complexity for identifying nondominated individuals in population is $\mathcal{O}(N^2)$; the worst time complexity for nondominated spare individual selection is $\mathcal{O}(N_l \log(N_l))$; the time complexity for cloning, recombination and mutation operation are $\mathcal{O}(N_l)$.

According to the operational rules of the symbol \mathcal{O} , the worst time complexity of one generation for HMEDA can be simplified as $\mathcal{O}(mN^2)$.

IV. EXPERIMENTAL STUDY

In order to validate the performance of the HMEDA, here we give the experimental results on eight test instances with or without variable linkages. The results will be compared with a fast and elitist multi-objective genetic algorithm (NSGA-II) [15], an immune inspired multiobjective algorithm (MISA) [8], and a regularity model-based multiobjective estimation distribution algorithm (RM-MEDA) [1]. Besides, inverted generation distance (IGD) [21] is adopted for measuring both the diversity and convergence of an approximation to the true Pareto optimal front. A binary quality metric, the Coverage of two sets [22] is employed to compare obtained solutions by different algorithms.

The eight problems are defined in the Table 2. The first two problems (F1 and F2) were developed by Zitzler, Deb, and Thiele [23], of which there are not linkage mappings among variables. The next three problems are from [1], and there are linear or near linear mappings among variables. Last three problems are modified by introducing complicated nonlinear mappings in the form of *sin()*, *circle*, and *S-curve*. Therefore, the last three problems are more complicated.

In this study, all the simulations run at a personal computer with P-IV 3.0G CPU and 2G RAM. By reference [1] and [15], the maximal number of function evaluations of the eight test instances is shown in Table 2. The experimental settings are as follows. The value of population N_E is 60 and the value of k is a tenth part of N_E when the neighbors of each data are computed. In SICA, the clonal scale N_l is 40. The indexes of simulated binary crossover and polynomial mutation are 15 and 20 respectively. The crossover probability of $pc=1$ and a mutation probability of $pm=1/n$ (where n is the number of decision variables for real-coded GAs). Indexes of different algorithms are presented in Table 1.

TABLE 1. INDEXES OF ALGORITHMS

Index	1	2	3	4
Algorithms	MISA	HMEDA	NSGA-II	RM-MEDA

In the following experiment, 30 independent runs are performed on each test problems. The statistical results of the IGD metric are shown by running performance in Fig 1. The Coverage of two sets is shown by box plots [24] in Fig 2.

Fig 1 shows that, for F1 and F2, NSGA-II obtained the best IGD-metric measure, and HMEDA obtained the next-best IGD-metric measure. F1 and F2 are MOPs without linkage mapping on variables. Therefore, the estimation distribution models are hard to discover the variable relations since there are not variable linkages at all. Therefore, HMEDA and RM-MEDA did not perform better than NSGA-II on these problems, while they performed better than MISA. With respect to F3 and F4, they are variants of ZDT1 and ZDT2 by introduced the linear variable linkages [1]. By Fig 1, we can obtain that RM-MEDA and HMEDA were better than NSGA-II and MISA on F3 and F4. The reason of this result is that NSGA-II and MISA have no efficient mechanism for using the regularity that the Pareto set of a continuous MOP submits to low manifold distribution.

Furthermore, in Fig 1, the experimental results of F5, F6, F7, F8 show that HMEDA could approximate the Pareto front very well while the other three algorithms perform a little worse. F6, F7, and F8 are variants of ZDT1 by introduced complicated nonlinear mappings among variables. It may be obtained that hybridization of LLE based EDA and SICA is beneficial to the optimization process.

Fig 2 shows the box plots of HMEDA against MISA, NSGA-II, and RM-MEDA on the eight problems in the Coverage of two sets respectively. The comparison between HMEDA and MISA shows that most of the values of $C(2, 1)$ (note that different Arabic numerals denote different algorithms in Table 1) are greater than these of $C(1, 2)$, i.e. it seems that almost all the MISA's solutions are weakly dominated by HMEDA's ones over 30 independent runs. By literature [1], if $0 < C(A, B) < 1$ and $0 < C(B, A) < 1$, it shows that A and B are incomparable. But if $C(A, B) \approx 1$ and $C(B, A) \approx 0$ indicates that A almost dominates B completely. Therefore, by Fig 2, we can get that solutions obtained by HMEDA dominate almost all the solutions obtained by MISA. The main limitation of MISA may be its binary representation. Since the test problems that we are dealing with have continuous spaces, real encoding should be preferred to avoid problems related to Hamming cliffs and to achieve arbitrary precision in the optimal solution [8].

TABLE 2. TEST INSTANCES

instance	variable	objectives	evaluation
F1	$[0,1]^n, n=30$	$f_1(\mathbf{x}) = x_1 \quad f_2(\mathbf{x}) = g(\mathbf{x}) \left[1 - \sqrt{f_1(\mathbf{x})/g(\mathbf{x})} \right]$ $g(\mathbf{x}) = 1 + 9 \left(\sum_{i=2}^n x_i \right) / (n-1)$	20,000
F2	$[0,1]^n, n=30$	$f_1(\mathbf{x}) = x_1 \quad f_2(\mathbf{x}) = g(\mathbf{x}) \left[1 - \sqrt{f_1(\mathbf{x})/g(\mathbf{x})} - f_1(\mathbf{x})/g(\mathbf{x}) \sin(10\pi x_1) \right]$ $g(\mathbf{x}) = 1 + 9 \left(\sum_{i=2}^n x_i \right) / (n-1)$	30,000
F3	$[0,1] \times [0,10]^n, n=29$	$f_1(\mathbf{x}) = x_1 \quad f_2(\mathbf{x}) = g(\mathbf{x}) \left[1 - \sqrt{f_1(\mathbf{x})/g(\mathbf{x})} \right]$ $g(\mathbf{x}) = 1 + 9 \left(\sum_{i=2}^n (x_i - x_1)^2 \right) / (n-1)$	25,000
F4	$[0,1] \times [0,10]^n, n=29$	$f_1(\mathbf{x}) = x_1 \quad f_2(\mathbf{x}) = g(\mathbf{x}) \left[1 - (f_1(\mathbf{x})/g(\mathbf{x}))^2 \right]$ $g(\mathbf{x}) = 1 + 9 \left(\sum_{i=2}^n (x_i - x_1)^2 \right) / (n-1)$	25,000
F5	$[0,1] \times [0,100]^n, n=29$	$f_1(\mathbf{x}) = x_1 \quad f_2(\mathbf{x}) = g(\mathbf{x}) \left[1 - \sqrt{f_1(\mathbf{x})/g(\mathbf{x})} \right]$ $g(\mathbf{x}) = 0.00025 \sum_{i=2}^n (x_i^2 - x_1)^2 - \prod_{i=2}^n \cos((x_i^2)/\sqrt{i-1}) + 2$	50,000
F6	$[0,1]^n, n=30$	$f_1(\mathbf{x}) = x_1 \quad f_2(\mathbf{x}) = g(\mathbf{x}) \left[1 - \sqrt{f_1(\mathbf{x})/g(\mathbf{x})} \right]$ $g(\mathbf{x}) = 1 + 9 \left(\sum_{i=2}^n (100 \sin(\pi x_i) - x_1)^2 \right) / (n-1)$	50,000
F7	$[0,1]^n, n=30$	$f_1(\mathbf{x}) = x_1 \quad f_2(\mathbf{x}) = g(\mathbf{x}) \left[1 - \sqrt{f_1(\mathbf{x})/g(\mathbf{x})} \right]$ $g(\mathbf{x}) = 1 + 10 \left(\left(x_2 - \frac{1}{2} \right)^2 + \left(x_1 - \frac{1}{2} \right)^2 - \frac{1}{4} \right) + \frac{9 \left(\sum_{i=3}^n (100x_i - x_1)^2 \right)}{n-1}$	100,000
F8	$[-1,1] \times [-1,3] \times [0,1]^n, n=28$	$f_1(\mathbf{x}) = x_1 \quad f_2(\mathbf{x}) = g(\mathbf{x}) \left[1 - \sqrt{f_1(\mathbf{x})/g(\mathbf{x})} \right]$ $g(\mathbf{x}) = 1 + 10 \left(\left(x_{2,1:\frac{N}{2}} \right)^2 + \left(x_{1,1:\frac{N}{2}} \right)^2 - 1 \right) + 10 \left(\left(x_{2,1:\frac{N}{2}} - 2 \right)^2 + \left(x_{1,1:\frac{N}{2}} \right)^2 - 1 \right) + 9 \left(\sum_{i=3}^n (100x_i - x_3)^2 \right) / (n-1)$	100,000

From Fig 2, we also can see the comparison between HMEDA and NSGA-II in terms of the Coverage of the two sets. The results show that NSGA-II did better than HMEDA on F1 and F2. However, HMEDA did better than NSGA-II on F3, F4, F5, F6, F7, and F8. Especially for F5, F7, and F8, almost all the solutions obtained by NSGA-II weakly dominated by HMEDA's solutions. Fig 2 also shows the

comparison between HMEDA and RM-MEDA by the Coverage of two sets. The results show that the majority solutions obtained by HMEDA are weakly dominated by the solutions obtained by RM-MEDA on F3 and F4. However, the majority solutions obtained by RM-MEDA are weakly dominated by the solutions obtained by HMEDA on F1, F2, F5, F6, F7, and F8.

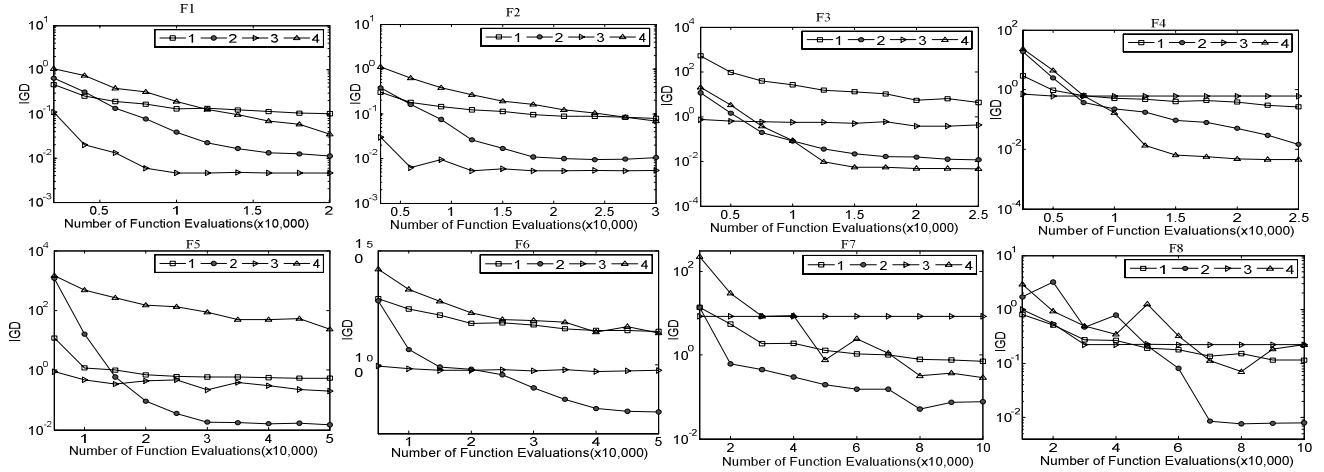


Fig. 1 Statistical mean values of inverted generation distance for the eight problems obtained by MISA, HMEDA, NSGA-II, and RM-MEDA respectively versus number of function evaluations over 30 independent runs. A logarithmic (base 10) scale is used for the Y-axis.

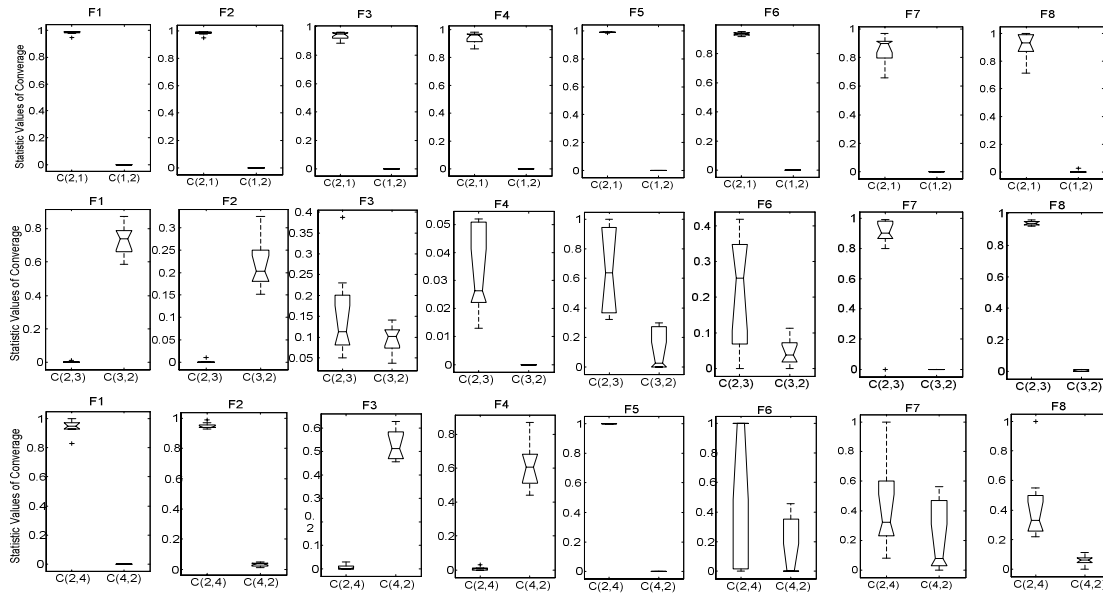


Fig. 2 Statistical values of the Coverage of the two sets obtained by MISA (1), HMEDA (2), NSGA-II(3) and RM-MEDA(4) in solving the eight problems respectively. Box plots are used to illustrate the distribution of these samples. In a notched box plot the notches represent a robust estimate of the uncertainty about the medians for box-to-box comparison. Symbol '+' denote for outliers.

In order to find the appropriate tradeoff between the EDA and SICA, we have investigated the different proportions of EDA to SICA in Fig 3. It could be obtained that the performance of HMEDA without SICA (denoted by 10/0) and HMEDA without EDA (denoted by 0/10) are not better than that of HMEDA.

The average values of the performance of HMEDA with the proportions of EDA to SICA from 1/9 to 9/1 in Fig 3 are similar. All of them are lower than those of HMEDA at 0/10 and 10/0. By Fig 3, it seems that HMEDA with the proportions of EDA to SICA at 5/5 and 6/4 performs a little better.

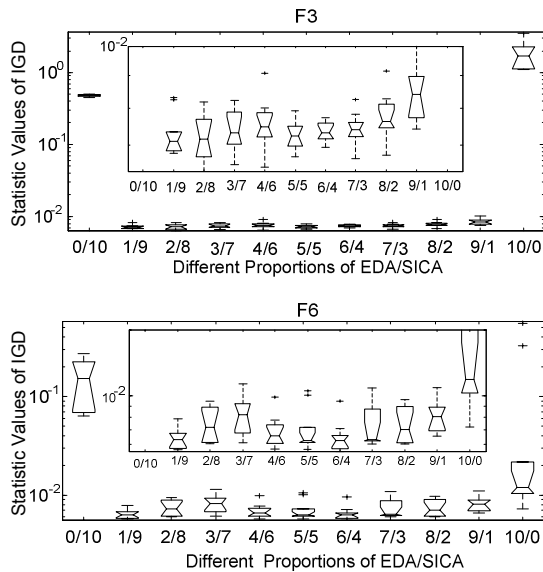


Fig. 3 Statistical average values of the IGD-metric for F3 and F6 obtained by HMEDA with different proportions of EDA to SICA over 30 independent runs respectively.

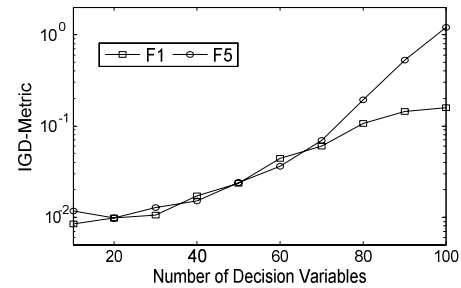


Fig. 4 The average value of IGD-metric obtained by HMEDA over 30 independent runs in F1 and F5 versus the different number of decision variables.

Fig 4 shows the scalability of HMEDA on F1 and F5 by increasing number of decision variables. We can see that the performance of HMEDA did not decline dramatically. Note that the range of X-axis is [0, 100] and the range of Y-axis is [0.01, 1]. We can estimate that the average values of IGD-metric rise linearly with the number of decision variables.

Overall considering the experimental results, we can see that hybridization of LLE based EDA and an immune inspired algorithm is beneficial to the optimization process. The tradeoff between the exploration ability of EDA and the exploitation ability of SICA is successfully maintained in HMEDA, which is an example to devise hybrid algorithm.

V. CONCLUDING REMARKS

In this study, we have introduced a novel hybrid multiobjective estimation of distribution algorithm combining LLE and sparse individual clonal algorithm. The EDA extracted global statistical information from current population to build an orthogonal sample model in the manifold space. The EDA is expected to find the common intrinsic characteristics of current population. Besides, an

immune inspired algorithm is introduced and combined with the model. The immune algorithm selected only the minority of nondominated individuals at sparse regions, which realized the enhanced local search ability of EDA. HMEDA was tested by eight multiobjective problems with or without variable linkages. RM-MEDA, NSGA-II, and MISA are used in comparison. The results show that HMEDA get competitive results at solving these multiobjective problems. Finally the tradeoff scheme is investigated by different proportions of these two methodologies.

REFERENCES

- [1] Qingfu Zhang, Aimin Zhou, and Yaochu Jin. RM-MEDA: A regularity model based multi-objective estimation of distribution algorithm. *IEEE Transactions on Evolutionary Computation*, 12(1): 41-63, 2008.
- [2] Coello Coello CA. Evolutionary multi-objective optimization: a historical view of the field, *IEEE Computational Intelligence Magazine*, 2006, Vol. 1, No. 1, pages 28-36.
- [3] Ishibuchi H, Tsukamoto N, Nojima Y. Evolutionary many-objective optimization: a short-review. In *Proceedings of the 2008 Congress on Evolutionary Computation, CEC' 08*, pages 2424-2431, Hong Kong, June 1-6, 2008.
- [4] Bosman P. A. N and Thierens D. Expanding from discrete to continuous EDAs: The IEDA, In *Proceedings of Parallel Problem Solving from Nature, PPSN-VI*, pp. 767-776, 2000.
- [5] Nicosia, G., Cutello, V., Bentley, P. J. and Timmis, J. (eds.). *Artificial Immune Systems, Proceedings of The Third International Conference on Artificial Immune Systems, ICARIS 2004*, Catania, Italy, September 13-16. Springer-Verlag, Lecture Notes in Computer Science, Vol. 3239.
- [6] Jacob, C., Pilat, M. L., Bentley, P. J. and Timmis, J. (eds.). *Artificial Immune Systems: Proceedings of The Fourth International Conference on Artificial Immune Systems, ICARIS 2005*, Banff, Alberta, Canada, August 14-17, Springer-Verlag, Lecture Notes in Computer Science, Vol. 3627.
- [7] Maoguo Gong, Licheng Jiao, Haifeng Du, and Liefeng Bo. Multi-objective immune algorithm with Pareto optimal neighbor-based selection. *Evolutionary Computation*, 16(2), 2008.
- [8] Coello Coello C. A. and Nareli Cruz Cortés, Solving multiobjective optimization problems using an artificial immune system, *Genetic Programming and Evolvable Machines*, Vol. 6, No. 2, pages. 163--190, June 2005.
- [9] Roweis S and Saul L. Nonlinear dimensionality reduction by locally linear embedding. Volume 290, pages 2323-2326, *Science* 2000.
- [10] Deb K. *Multi-objective optimization using evolutionary algorithms*. John Wiley & Sons, Chichester, UK, 2001.
- [11] Coello Coello C A, Van Veldhuizen D A, and Lamont G B. *Evolutionary algorithms for solving multi-objective problems*. Kluwer Academic Publishers, New York, USA, 2002.
- [12] Jolliff I. T. *Principal component analysis*, Springer-Verlag, New York, 1989.
- [13] Cox T., and Cox M. *Multidimensional scaling*, Chapman&Hall, London, 1994.
- [14] Tenenbaum J. B, Silva V. de, and Langford J. C. A global geometric framework for nonlinear dimensionality reduction. Volume 290, pages 2318-2323, *Science* 2000.
- [15] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2): 182-197, 2002.
- [16] Khan N. Bayesian optimization algorithms for multiobjective and hierarchically difficult problems. Master's thesis, University of Illinois at Urbana- Champaign, Urbana, IL. 2003.
- [17] Pelikan M, Goldberg D. E, and Cant-Paz E. BOA: the Bayesian optimization algorithm. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-99)*, W. Banzhaf et al., Eds. San Francisco, CA: Morgan Kaufmann, 1999, pp. 525-532.
- [18] Pelikan M, Goldberg D. E, and Tsutsui S. Hierarchical Bayesian optimization algorithm: toward a new generation of evolutionary algorithms. *SICE Annual Conference in Fukui*, August 4-6, 2003, Fukui University, Japan, pp.2738-2743.
- [19] Laumanns M, Ocenasek J. Bayesian optimization algorithm for multiobjective optimization. In *Conference on Parallel Problem Solving from Nature (PPSN VII)*, volume 2439 of LNCS, pages 298-307. Springer, 2002.
- [20] Ahn C. W. Theory, design, and application of efficient genetic and evolutionary algorithms. Doctoral dissertation, Gwangju Institute of Science and Technology, Korea, 2005.
- [21] Sierra M. R., and Coello Coello, C. A. A study of fitness inheritance and approximation techniques for multiobjective particle swarm optimization. In *proceeding of congress of evolution computation (2005)*, Edinburgh, U.K., 2005, pp. 65-72.
- [22] Zitzler E and Thiele L. Multi-objective optimization using evolutionary algorithms--A comparative case study. In *Parallel Problem Solving from Nature-PPSN V*, volume 1498 of Lecture Notes in Computer Science, pages: 292-301, 1998.
- [23] Zitzler E, Deb K, and Thiele L. Comparison of multi-objective evolutionary algorithms: empirical results. *Evolutionary Computation*, 8(2): 173-195, 2002.
- [24] McGill R, Tukey J. W, and Larsen W. A. Variations of boxplots. *The American Statistician*, Vol. 32, pages: 12-16, 1978.