2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

A Novel Algorithm for Many-Objective Dimension Reductions: Pareto-PCA-NSGA-II

Ronghua Shang[a], Member, IEEE, Kun Zhang[a], Licheng Jiao[a], Senior Member, IEEE, Wei Fang, Xiangrong Zhang[a], Member, IEEE, and Xiaolin Tian[a]

a. Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China, International Research Center for Intelligent Perception and Computation, Xidian University, Xi'an 710071, China;
b.Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education), Department of Computer Science and Technology,Jiangnan University, No. 1800, Lihu Avenue, Wuxi, Jiangsu 214122, China
rhshang@mail.xidian.edu.cn

***Abstract*-Many-objective problem has more than 3 objectives. Because of the extraordinary difficulty of acquiring their Pareto optimal solutions directly, traditional methods will be out of operation for such problems. In recent years, many researchers have turned their attention to the study of this area. They are interested in two areas: acquiring some part of Pareto front which is useful to the researchers (Preferred Solutions) and reducing redundant objectives. In this paper, we combine two dimension reduction methods: the method based on Pareto optimal solution analysis and the method based on correlation analysis, to form a novel algorithm for dimension reduction. Firstly, the Pareto optimal solutions are acquired through NSGA-II. Then the objectives who contribute little to the number of non-dominated solutions are removed. At last, the dimension of objectives is reduced further according to their contribution to the principal component in PCA analysis. In this way, we can acquire the right non-redundant objectives with low time complexity. Simulation results show that the proposed algorithm can effectively reduce redundant objectives and keep the non-redundant objectives with low time.**

***K*eywords-Pareto optimal solution, PCA, many-objective optimization, multi-objective optimization, dimension reduction, DTLZ5 (I, M).**

I. INTRODUCTION

In recent years, Evolutionary Computation (EC) has become of intriguing interest. Objective optimization is one of the main applications of EC. A problem is considered as single-objective optimization problem if its objective is only one; and it is considered as multi-objective optimization problem if its objectives are more than one. In multi-objective optimization area, many-objective problem means its objectives being more than three [1]. The general model of a multi-objective optimization problem [2] is shown below:

$$\begin{aligned} &\min\ \boldsymbol{F}(\boldsymbol{x}) = \left(f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_m(\boldsymbol{x})\right) \\ &s.t.\ \ g_i(\boldsymbol{x}) \le 0, \quad\ \ i = 1,2,\cdots,q \\ &\qquad h_i(\boldsymbol{x}) = 0, \quad i = q+1, q+2, \cdots, k \\ &\qquad \boldsymbol{l} < \boldsymbol{x} < \boldsymbol{u}, \quad \boldsymbol{x} = (x_1, x_2, \cdots, x_n) \\ &\qquad \boldsymbol{l} = (l_1, l_2, \cdots, l_n), \quad \boldsymbol{u} = (u_1, u_2, \cdots, u_n) \end{aligned} \tag{1}$$

This work was partially supported by the National Natural Science Foundation of China, under Grants 61371201, 61203303 and 61272279, the Fundamental Research Funds for the Central Universities, under Grant K5051302028, the Fund for Foreign Scholars in University Research and Teaching Programs (the 111 Project) under Grant B07048.

Where, $\boldsymbol{F}(\boldsymbol{x})$ indicates the objective vector; m is the dimension of $\boldsymbol{F}(\boldsymbol{x})$; $\boldsymbol{x}$ represents the independent variable vector, which has a value between the upper bound $\boldsymbol{l}$ and the low bound $\boldsymbol{u}$; n is the dimension of $\boldsymbol{x}$; $g_i(\boldsymbol{x})$ and $h_i(\boldsymbol{x})$ represent the constraints of the optimization problem, including the equality and inequality constraints. As a general rule, an optimization problem involves several inequality constraints (q) and equality constraints (k-q). If both q and k are equal to zero, the problem becomes an unconstrained multi-objective optimization problem; otherwise, the problem is called constrained multi-objective problem.

II. RELATED WORKS

In the past few years, some classical multiobjective optimization algorithms based on evolutionary algorithm have proposed, such as NSGA-II [3], MOEA/D [4], PESA-II [5], SPEA2 [6] and so on. Nevertheless, when the objectives are more than 3, the many-objective optimization problems will involve some new features, which make the traditional classical algorithms poor effect, or even out of action. To solve such difficulties caused by the increase of objectives, we should first have some knowledge about the new features, and then search for some effective methods to solve these difficulties.

Along with the number of objectives increasing, the dimension of objective space grows rapidly. Any two individuals in the population are easier to be inter-dominated in the evolutionary process [4]. As a result, most individuals in the population become non-dominated in the early stage of evolution, making the selection strategy weaken its pressure to the Pareto front, or even lose its pressure [7]. This pressure is one of the main reasons that EAs can converge to the Pareto optimal solutions.

A simple method to solve the loss of selection pressure is to enlarge the size of population and increase the number of generations. A large population can make the probability that the new better solutions appear higher than a small population, while a large number of generation give algorithms more times to acquire better solutions. However, there have been some papers [8] pointed out that, when the number of objectives increases by one, the population should also double itself to avoid losing its selection pressure. As a result, accompanied by the increase of the objectives, the population size expands exponentially [7].

Along with the increase of objectives and population size, the computational complexity grows rapidly. When objectives are too many, the time EAs consume in obtaining the Pareto optimal solutions is often intolerable for a desktop computer. So an algorithm with low computational complexity must be found to reduce the time cost. This has been one of the difficulties we have faced with in many-objective optimization area in recent years.

The increase of the dimension of objective space will also cause difficulty in the visualization of Pareto front. When the objectives are more than 3, we could not show the high dimension Pareto front in Cartesian Coordinates system, which is different from multi-objective optimization problem with only 2 or 3 objectives. Lotov [9] and Tenenbaum [10] proposed methods for the visualization of high dimensional Pareto front, called decision map [9] and geodesic map [10], respectively. The analysis in reference [8] confirm that two methods need a huge amount of data, which makes them limited in application and still need much research. However we find that if we remove the non-conflictive objectives while keeping the inter-confliction among objectives, the dimension of objective space could be reduced. This is really helpful for the visualization of the final solutions, especially when the objectives can be reduced to 3 or 2.

For traditional multi-objective optimization problem, researchers have proposed many test problems to inspect the effect of new algorithms. These test problems involve all kinds of features that Pareto front has, such as multimodal, deceptive multimodal, the bias, degenerated front, disconnected front, the concavity or convexity, and so on[11]. Moreover most of these test problems have the characteristic of scalability, that is, the objectives and the variables are scalable. As a result, these test problems can be applied to inspect many-objective optimization algorithms. For instance, Deb et al [12] proposed DTLZ test problem and Huband et al [11] proposed WFG test problem. There is a description about other common test problem in [11]. But few of these test problems have the degraded Pareto front. This is one of the features that make the objectives of these test problems can be reduced. Consequently, the test problems that can really be applied to test many-objective optimization algorithms, especially dimension reduction algorithms, are still not enough. For this reason, Deb et al improved the DTLZ5 test problem, and proposed DTLZ5 (I, M) test problem suit, which is specially used to test the effect of dimension reduction algorithms in many-objective optimization.

In recent years, the research about many-objective optimization problems, especially those with a large number of objectives, is still inadequate. But on the other hand, what is promising is that there is more and more research in this area, and some algorithms have been proposed by researchers over the past few years. These algorithms resolve the many-objective optimization problems from the following aspects:

1. In many-objective optimization problems, the users are not interested in all the Pareto optimal solutions, but only a local part of them. The rest parts may somehow cause interference to the final decision for users. Therefore, when dealing with such problems, users do not need to resolve all the Pareto optimal solutions, but only the ones the users interested in. Such method is called multi-objective optimization based on priori-knowledge. This method usually assumes that there are no redundant objectives [13] and may somehow lose diversity and shrinks the distribution of Pareto front. Because the priori-knowledge makes the individuals in the population converging to a special part of the Pareto front and reducing the non-dominated solutions, the selection pressure and the convergence to Pareto front are improved.

According to the way the priori-knowledge is considered, the priori-knowledge is divided into priori information, interactive information and posteriori information, which are described in detail in [1] and [14], while the posteriori information is the most common choice for the researchers [14]. Corne et al [15] effectively analyzed the characteristics of the Pareto optimal solutions, which offers us guidance about how to choose priori-knowledge and which part of the Pareto solutions is useful. Jaszkiewicz et al [16] proposed a complex relational model of local preference, and based on which a kind of immune clone algorithm solving preference multi-objective optimization problems was proposed [1].

2. In practical application, the assumption about non-redundant objectives doesn't stand in most of common conditions. Zitzler [17] did a detailed analysis about the redundancy among the objectives in many-objective optimization. The main characteristic of the redundant objectives being removable is that there are correlations among some objectives in the internal of the Pareto optimal solutions and the values of these objectives increase or decrease simultaneously (or the trend is similar). Be such objectives removed, the Pareto optimal solutions could be acquired without influence, while the time needed could be saved. For this reason, this kind of optimization problems can be solved by computing the Pareto optimal solutions after the objectives being reduced. Through such Pareto optimal solutions, we can rebuild the whole Pareto front for the primitive problem.

Many dimension reduction algorithms proposed in recent years mainly focus on two aspects: analyzing the non-dominated solutions or analyzing the correlations among objectives. Zitzler et al [17] have proposed a dimension reduction algorithm which is based on the dominance relationship among the Pareto optimal solutions. Meanwhile, Coello Coello et al [18] have proposed a dimension reduction algorithm using feature selection technique. This kind of algorithm is also called dimension reduction method based on the features. Deb et al [8] have proposed a dimension reduction algorithm based on PCA analysis, which uses the linear correlations among the objectives to reduce the dimension of objective space. This algorithm is further developed in [13]. However, some papers point out that [11] a few of key points are enough to characterize the real dimension of objective space. A good method is the introduction of priori-knowledge, which can help us to acquire the key points among Pareto optimal solutions. Singh [19] has proposed the conception of "corner solution" and designed an algorithm of dimension reduction using this conception.

3. In most of the multi-objective optimization algorithms, researchers use the concept of "Pareto optimal", which was proposed by Edgeworth [20] and later improved by Pareto [21], to select and rank individuals in the population [22]. This method shows good effect when the objectives are few, but low efficiency with amount of time cost when the objectives are many. Therefore, researchers have proposed several improved concepts to instead "Pareto optimal". For instance, the conception of "E-optimal" has been proposed in reference [22], and a fast algorithm has formed based on this concept. Some classic concepts of the representation of optimization have been summarized in reference [14].

Although researchers have proposed lots of new thinking about dimension reduction, a single method either cannot reduce the dimension thoroughly or cost a long period of time. Therefore, how to reduce the time cost while keeping the effect of dimension reduction is an important aspect worth studying. Because different algorithms tend to have their own advantages and disadvantages, the combination of two or more such algorithms may offset their weakness and achieve a better effect. In this paper, we combine the advantages of two algorithms (the algorithm which is based on the analysis of the Pareto optimal solutions [19] and the algorithm which is based on the PCA analysis of objectives [13]) to design a new algorithm. While keeping the accuracy of dimension reduction, the algorithm costs less time than the algorithms in references [8] and [13], both of which are based only on the PCA analysis of objectives.

The rest parts of this paper are organized as follows: the third part describes the proposed algorithm in detail; the fourth part presents the test problem suit which is used to verify the effect of the algorithm, meanwhile the result and its analysis will also be involved; the last part mainly summarizes the deficiencies of the proposed algorithm and the future research direction.

III. PARETO-PCA-NSGA-II ALGORITHM

In this section, we introduce two dimension reduction methods used in the proposed algorithm, which are based on the analysis of Pareto optimal solution and the PCA analysis of objectives respectively, and the whole algorithm proposed in this paper, then we give the process of the proposed algorithm. We firstly acquire the Pareto optimal solutions using the NSGA-II algorithm, and then we reduce the objectives through the above two dimension reduction methods orderly and iteratively until no objective can be reduced. The analysis of the proposed algorithm will be shown as follows.

A. The dimension reduction method based on analysis of Pareto optimal solutions

1) The dimension reduction method based on "corner solutions"

We know that there is a correspondence between Pareto optimal solutions and Pareto front, which is described by the objective functions. In order to acquire the maximum of the whole Pareto front, we have to keep the most of the Pareto optimal solutions when reducing the objectives. The dimension reduction method based on the analysis of Pareto optimal solutions takes the relative decrement of the Pareto optimal solutions as the criterion whether the corresponding objective should be removed. According to whether the criterion of the reduction is to minimize the objectives with a certain decrement of Pareto optimal solutions or to maximize the Pareto optimal solution with a certain number of objectives, Singh et al [19] proposed δ-moss problem which takes the former criterion and k-moss problem which takes the latter criterion. Because δ-moss problem does not need to know the number of objectives, this kind of problem is commonly used.

Let N_F and N_{F-f} respectively represent the number of Pareto optimal solutions before and after one objective is removed, then the ratio of the two variables can be determined by the following formula:

$$R_f = \frac{N_{F-f}}{N_F} \tag{2}$$

When the value of R_f is bigger than a certain threshold which is artificially settled, the removal of the corresponding objective may has little effect on the Pareto optimal solutions, which means the corresponding objective should be removed. On the other hand, if the value is less than the threshold, then the effect is great, and the objective should not be removed.

In [19], Singh proposed the concept of "corner solution" and designed a dimension reduction algorithm based on this concept. "Corner solutions" are some special points among the Pareto optimal solutions. If the solutions converge to one special point when one or more objectives take the minimum values, we call this point "corner solution". The dimension reduction algorithm based on the corner solution takes these corner solutions as the representation of Pareto optimal solutions, and realizes very low time complexity.

2) The dimension reduction method based on uniform Pareto optimal solutions

In spite of its low time complexity, the dimension reduction algorithm based on corner solutions [19] cannot guarantee the thoroughness of dimension reduction. Meanwhile, in the process of acquiring the corner solutions, the shapes of Pareto front may have some influence on the result, which means some interference point may appear to impact the final result of dimension reduction.

The proposed algorithm takes advantage of the main principal of the method in reference [19]. As the difference between the two methods in this paper and reference [19], for ease of the PCA analysis after the analysis of uniform Pareto optimal solutions, we use the uniformly distributed Pareto optimal solutions on the Pareto front instead of the corner solutions in reference [19]. There is no doubt that this will increase the computational complexity and the time cost, but the final result will not be influenced by the shape of Pareto front, thereby accuracy of dimension reduction is enhanced. In addition, in order to minimize the number of objectives, the proposed algorithm carries the analysis of Pareto optimal solutions iteratively until the result of the analysis doesn't change, which is different from the one iteration of corner solutions analysis [19]. In this way, the dimension of objective space can be reduced thoroughly.

In the analysis of uniform Pareto optimal solutions, the first step is to acquire the Pareto optimal solutions which are uniformly distributed on the Pareto front. NSGA-II [3] is adopted to obtain such an approximation of Pareto optimal solutions. And then, we use the method introduced above to estimate whether the corresponding objective is removable; if not, we should restore the objective; if yes, considering that the objective has been removed, we should do nothing. In spite of NSGA-II, some other algorithms can also acquire uniformly distributed Pareto optimal solutions, such as MOEA/D, which can used instead of NSGA-II. It can be found from the later analysis that even if the solutions cannot converge to the real Pareto Front accurately, this method still works. The process of dimension reduction method in the paper is shown in Algorithm 1.

Algorithm 1: The dimension reduction method based on uniform Pareto optimal solutions
variables：*Rpa*：Pareto non-dominated population; N_F：The number of non-dominated solution in *Rpa;* *DRpa*：The Pareto non-dominated population after one objective is removed; $N_{F\text{-}f}$：The number of non-dominated solution in *DRpa;* *C*：the threshold of the ratio of $N_{F\text{-}f}$ and *N.* Algorithm process: 1. Initialize the iteration count variable: $i=0$; 2. Acquire the *Rpa* through running the NSGA-II algorithm; 3. While $i<$the dimension of objective space 4. $N_F\leftarrow$the number of non-dominated solution in *Rpa;* 5. Remove the *ith* objective in objective space; 6. Reacquire the *DRpa* through non-dominated strategy; 7. $N_{F\text{-}f}\leftarrow$the number of non-dominated solution in *DRpa;* 8. Calculate the ratio of $N_{F\text{-}f}$ and N_F: $R_f=N_{F\text{-}f}/N_F$; 9. If $R_f<C$, then remain the situation; otherwise rebuild the *ith* objective in objective space; 10. $i=i+1$; 11. end

B. Dimension reduction process based on PCA analysis

1) PCA analysis

PCA analysis is the abbreviation of principal component analysis, which is a kind of statistical analysis tool for multivariate data. The main function of such tool is to reduce the dimension of the data through analyzing the statistical correlation, minimizing the correlation and keeping the biggest variation among the data.

When applying the PCA to EAs, we should firstly acquire the Pareto optimal solutions. Then, through the PCA analysis of these solutions, we can acquire the information about principal component and secondary components. After that, we can rank the objectives according to their contributions to the principal component and secondary components. And finally, the objectives with little contribution should be removed.

Among the statistic characteristics of the data, the covariance matrix and correlation matrix both contain the linear correlation among the data. Therefore both matrixes are often adopted by the PCA analysis. For instance, the correlation matrix is adopted in reference [8]. However, because both matrixes could only involve the linear correlation among the data and lose the nonlinear correlation, the algorithm in reference [8] has some limitations. Deb [13] used kernel matrix instead of the correlation matrix, he proposed a nonlinear dimension reduction framework.

Because the data used for PCA analysis usually locate in a specific interval, the results of PCA analysis are related to the interval. To keep the accuracy of PCA analysis, we should solve the Pareto optimal solutions accurately, or we might be led to the wrong results. To assure that the solutions are located on the Pareto front, the algorithms in references [8] and [13] take large-scale population and thousands of iterations. Nevertheless, such a process may cost huge amount of time which might be out of the capacity of a common desktop computer. The algorithm may be more practical if we can lessen the time cost of the algorithm.

2) The process of dimension reduction based on PCA analysis

The PCA in this paper still follows the general principle of PCA. That is, we firstly obtain the Pareto optimal solutions; and then carry the PCA analysis of the solutions. Because most of the redundant objectives are removed during the analysis of Pareto optimal solutions, few objectives, or even no objective is removed during the PCA analysis sometime. To assure that the redundant objectives are totally removed, we should reduce the dimension of objective space iteratively with the analysis of Pareto optimal solutions. The process of dimension reduction based on PCA analysis is shown in Algorithm2.

Algorithm2: the process of PCA analysis
variables：*Rpa*：Pareto non-dominated population; *C*: threshold of calculation of the eigenvalue; *CT*: threshold of the correlation coefficient between objectives. Algorithm process: 1. Initialize the iteration count variable: $i=0$; 2. $R\leftarrow$the correlation coefficient matrix of *Rpa;* 3. *EigValue*$\leftarrow$the eigenvalue vector of the matrix $R*R$, *EigValueVector*$\leftarrow$the eigenvector matrix of the matrix $R*R$; *EigValue*$\leftarrow$the normalized vector of *EigValue,* The vector *EigValue* and each row of the matrix *EigVector* are ordered based on the order of the value of *EigValue* elements in small to large; 4. *EigV*$\leftarrow$*EigValue(1)*, *EV*$\leftarrow$*EigVector(:,*1*)*, *ReOrder*$\leftarrow$the subscript of objectives corresponding to the maximum and minimum value of the elements in vector; 5. $i\leftarrow2$; C=0.97 6. While *EigV*<*C* 7. *EV*$\leftarrow$*EigVector(:,i)*, *EigV=EigV+EigValue(i)* 8. If *EigValue(i)*<0.1

```
    Re ←the subscript of objectives corresponding to the
    maximum value of the elements in vector EV;
      ReOrder←[ReOder; Re]
    Else
      t=sign(EV),l=length(EV);
      If sum(t)=l or sum(t)=-l
        Re ← the subscript of objectives corresponding to the
maximum value of the elements in vector EV;
        ReOrder←[ReOder; Re]
      Else
        PM←max(EV);NM←min(EV)
        If PM>|NM|&|NM|>0.8*PM || PM<|NM|&PM>0.8*|NM|
          Re←the subscript of the objectives corresponding to
PM and NM in EV
          ReOrder←[ReOder; Re]
        Else
          Re←the subscript of objectives corresponding to
max(PM,|NM|) in EV
          ReOrder←[ReOder; Re]
        End
      End
   End
End of if and while
Remove the recurring objectives in vector ReOrder;
R1←R(ReOrder,ReOrder);
a←length(ReOrder);b←length(FOv)
CT←(1-a/b)*C
for every two column of R1 , presented as the ith and jth
column (i<j)
 If the sign of the two column are exactly the same, then
    If R1(i,j)>CT
      ReOrder(j) should be removed from ReOrder
    end
 End
End of for
The objective remained in vector ReOrder are the last
non-redundant objective.
```

C. *The main framework of the algorithm*

The proposed algorithm firstly obtains the Pareto optimal solutions using the NSGA-II algorithm. And next, we use the analysis of uniform Pareto optimal solutions to reduce the dimension of objective space, which alternate with the NSGA-II iterations until the dimension of the objective space cannot be reduced. After that, the dimension of the objective space will be reduced further through PCA analysis of objectives. To assure that the dimension has been reduced adequately, the whole process will be carried out iteratively, until the dimension of the objective space cannot be reduced any more in both processes of PCA analysis and the analysis of uniform Pareto optimal solutions.

To make the furthest use of the data acquired during the NSGA-II iterations, the method of online dimension reduction [13] will be adopted. The whole process of the proposed algorithm is shown as following:

Algorithm 3: The main framework of the proposed algorithm
variables：*FOv*：objective vector; *ORc*：the threshold of Pareto analysis dimension reduction; *Epop,Epa*：the parent population in variable space ad objective space; *Rpa*：Pareto non-dominated population; *SGMax*：the number of iteration of population; *SNM*：population size. Algorithm process： 1. Initialize the vector *FOv* and the threshold *ORc*, randomly generate the Parent population *EPOP* and calculate the value of the individuals in *EPOP* which are stored in *Epa;* 2. Run the algorithm NSGA-II with the size of the population *SNM* and the maximum generation *SGMax* 3. *Rpa*←the Pareto non-dominated population acquired using the algorithm NSGA-II 4. Reduce the dimension of objective space through the analysis of the Pareto optimal solution in *Rpa*, the result of dimension reduction is stored in vector *Tov* 5. If length(*TOv*)!=length(*FOv*), then *FOv*←*TOv*, and goto step 2; otherwise goto step 6 6.Reduce the dimension of objective space through the PCA analysis of the objective in *Rpa*, the result of dimension reduction is stored in vector *Tov* 7. If length(*TOv*)!=length(*FOv*), then *FOv*←*TOv*, and goto step 2; otherwise the algorithm is ended. And the final result is stored in *FOv*.

IV. SIMULATION RESULTS

A. *DTLZ5 (I, M) test problem suit*

DTLZ5 (*I, M*) [8] test problem suit was improved from DTLZ5 test problem [12]. DTLZ5 has a degraded Pareto front, which is a good feature for dimension reduction. However, due to the fixed number of non-redundant objectives in the test problem, DTLZ5 could not adequately test the effect of the algorithms. DTLZ5 (*I, M*) test problem suit overcame the shortage of DTLZ5, Making the number of objectives *M*, the number of non-dominated objectives *I* and the number of variables *n* changeable without influence to each other. For this reason, Deb et al proposed DTLZ5 (*I, M*) to test the effect of the algorithm he proposed. Moreover, this test problem suit is also used in reference [8]. For the clarity, the common formula of DTLZ5 (*I, M*) test problem suit [8] is given as follows.

$$
\begin{aligned}
&\min\ f_1(\boldsymbol{x}) = \left(1+100g(\boldsymbol{x}_M)\right) cos(\theta_1)\cdots cos(\theta_{M-2})\, cos(\theta_{M-1})\\
&\min\ f_2(\boldsymbol{x}) = \left(1+100g(\boldsymbol{x}_M)\right) cos(\theta_1)\cdots cos(\theta_{M-2})\, sin(\theta_{M-1})\\
&\ \vdots \qquad \vdots\\
&\min\ f_M(\boldsymbol{x}) = \left(1+100g(\boldsymbol{x}_M)\right) sin(\theta_1)\\
&g(\boldsymbol{x}_M) = \textstyle\sum_{x_i\in \boldsymbol{x}_M}(x_i-0.5)^2\\
&\theta_i = \begin{cases} \dfrac{\pi}{2}x_i & i = 1,2,\cdots,I-1\\ \dfrac{\pi}{4\left(1+g(\boldsymbol{x}_M)\right)}\left(1+2g(\boldsymbol{x}_M)x_i\right) & i = I,\cdots,(M-1)\end{cases}\\
&0\le x_i\le 1 \qquad\qquad for\ i=1,2,\cdots,n
\end{aligned} \tag{3}
$$

where, $\boldsymbol{x}_M$ is the distance parameter [11], while x_1~$x_{M-1}$are the position parameters [11]. During the evolution, the location of individuals on the Pareto front and the dimension of objective space are decided by x_1~x_{M-1}, while $\boldsymbol{x}_M$ only affects the way individuals converge to the Pareto front. Both x_1~x_{M-1} and $\boldsymbol{x}_M$ form the variable space, and the values of them range in the interval [0, 1]. On the assumption that the dimension of $\boldsymbol{x}_M$ is *k*, the dimension of variable space could be decided by the fo formula (4) [8]:

$$n = M + k - 1 \tag{4}$$

In the formulas (3) and (4), *M* is the dimension of the objective space, and *I* in formula (3) is the real dimension of the Pareto front. The objective values of the Pareto optimal solutions have the following relationship [8]:

$$\textstyle\sum_{i=1}^{M}(f_i)^2 = 1 \tag{5}$$

This formula could help us to estimate whether the individuals converge to the Pareto front.

B. The relevant parameter settings

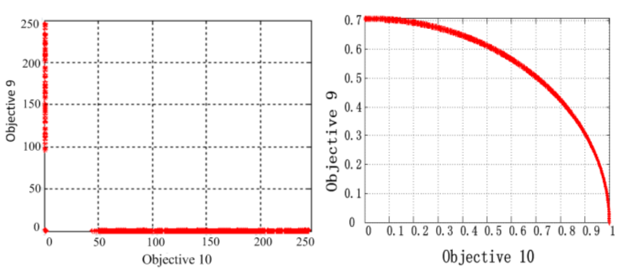
In order to facilitate the comparison of the proposed algorithm with that in reference [8], we use the same scale of population as in reference [8] which involves 800 individuals. To assure that the population has adequately evolved before the PCA analysis, we arrange 400 iterations for every NSGA-II process. Large numbers of experiments show that the ideal threshold of R_f is between 0.7 and 0.8, which in this interval has little effect on the result of dimension reduction. However, when the value of R_f is too small, the iterations tend to stop prematurely and result in the deletion of non-redundant objectives; while when the value is too large, it tends to remove few objectives before the PCA analysis, making it difficult to effectively converge to the Pareto optimal solutions. Generally we set the R_f to be 0.8.

C. The analysis of the results

Because the same test problem suit is adopted in this paper and in reference [8], and the algorithm in reference [8] has a better effect than other algorithms while being completed in the theoretical analysis, the algorithm in reference [8] is chosen to compare with the proposed algorithm. Compared with the larger number of iterations in the NSGA-II process, the time cost of the PCA analysis and the analysis of uniform Pareto optimal solutions could be ignored. Meanwhile, because both two algorithms have the same scale of population, the time complexity of two algorithms could be compared by the total iterations of the NSGA-II adopted in this paper.

1) The analysis of the simulation results

In the process of dimension reduction, before the PCA analysis, we should first make the population converge to the Pareto front through a huge number of iterations. However, the experiments tell us that, when the dimension of objective space is high, it is hard for the population to converge to the Pareto front even if a huge number of iterations are carried. For instance, we have carried an experiment using DTLZ5 (2, 10) test problem, which has ten objectives with two dimensional Pareto front. We ran NSGA-II on it using a population of 800 individuals for a thousand iterations. The simulation results are shown in figure1.



(a) (b)

Figure1: Pareto optimal solutions of DTLZ5 (2, 10)

Figure 1 (a) shows the resulting Pareto front of 1000 iterations, meanwhile figure 1 (b) represents the real Pareto front of this problem [8]. The axes represent respectively the objective f9 and f10. The comparison of two figures tells us that, even though 1000 iterations have been carried, the individuals of the population could not converge to the real Pareto front, and many of them are located in the wrong place. Moreover, in the whole process of the algorithms in references [8] and [13], the number of objectives tends to decrease. As a result, there is no need to carry so huge scale of population and such large number of iterations when the objectives are few. For this reason, using the changeable scale of population and iterations according to the number of the objectives is a good way to reduce the time cost of the algorithm. On the other hand, if we could reduce some redundant objectives before the PCA analysis, then we can enhance the selection pressure of individuals and make it easier for the population to converge to the Pareto front. The proposed algorithm performs the analysis of uniform Pareto optimal solutions to remove the redundant objectives before the PCA analysis. In order to illustrate the feasibility of this process more clearly, we give the result of the analysis of Pareto optimal solutions for DTLZ5 (2, 10) and DTLZ5 (3, 10) respectively in table 1 and table 2.

In both two tables, we ran the NSGA-II on both two problems for 1000 iterations with the population of 800 individuals, we statistic the ratio of the number of Pareto optimal solutions before and after each objective being removed after every 100 iterations. Each row of the tables represents the results of every statistic, while each column represents 10 statistical results for every objective. Each cell in tables corresponds to the value of R_f for every objective mentioned in part 2. In order to make the results of the analysis more clear, we use the underline to emphasize the objectives with low statistic ratio.

Table 1: Results of DTLZ5 (2, 10)

Iterations	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10
100	0.668	0.700	0.890	1.000	0.962	0.840	0.885	0.503	<u>0.329</u>	<u>0.016</u>
200	0.576	0.710	0.907	0.997	0.995	0.943	0.553	<u>0.350</u>	<u>0.196</u>	<u>0.089</u>
300	0.572	0.707	0.882	1.000	0.992	0.987	0.577	<u>0.310</u>	<u>0.171</u>	<u>0.060</u>
400	0.648	0.675	0.849	1.000	1.000	0.997	0.578	<u>0.351</u>	<u>0.199</u>	<u>0.066</u>
500	0.602	0.703	0.848	0.999	0.994	0.997	0.585	<u>0.358</u>	<u>0.179</u>	<u>0.058</u>
600	0.576	0.688	0.865	1.000	0.996	0.994	0.579	<u>0.313</u>	<u>0.173</u>	<u>0.060</u>
700	0.624	0.659	0.862	1.000	0.998	0.996	0.513	<u>0.349</u>	<u>0.188</u>	<u>0.062</u>
800	0.601	0.688	0.849	0.999	0.988	1.000	0.555	<u>0.381</u>	<u>0.201</u>	<u>0.064</u>
900	0.654	0.651	0.848	1.000	0.991	0.992	0.622	<u>0.393</u>	<u>0.198</u>	<u>0.064</u>
1000	0.673	0.630	0.846	1.000	0.998	0.992	0.623	<u>0.369</u>	<u>0.204</u>	<u>0.058</u>

Table 2: Results of DTLZ5 (3, 10)

Iterations	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10
100	1.000	0.999	0.999	0.999	0.998	0.996	0.996	<u>0.007</u>	<u>0.008</u>	<u>0.005</u>
200	0.999	0.999	0.996	0.996	0.994	0.992	0.990	<u>0.003</u>	<u>0.009</u>	<u>0.005</u>
300	1.000	0.999	0.998	0.998	0.996	0.995	0.995	<u>0.004</u>	<u>0.009</u>	<u>0.002</u>
400	1.000	0.999	0.998	0.997	0.997	0.996	0.995	<u>0.003</u>	<u>0.006</u>	<u>0.002</u>
500	1.000	0.999	0.997	0.995	0.995	0.994	0.994	<u>0.002</u>	<u>0.003</u>	<u>0.009</u>
600	1.000	0.996	0.996	0.994	0.993	0.991	0.990	<u>0.002</u>	<u>0.007</u>	<u>0.005</u>
700	1.000	1.000	0.999	0.997	0.994	0.993	0.991	<u>0.003</u>	<u>0.006</u>	<u>0.004</u>
800	1.000	0.999	0.999	0.999	0.997	0.997	0.995	<u>0.004</u>	<u>0.008</u>	<u>0.004</u>
900	1.000	0.999	0.997	0.996	0.995	0.991	0.988	<u>0.005</u>	<u>0.006</u>	<u>0.005</u>
1000	1.000	1.000	0.999	0.999	0.998	0.994	0.992	<u>0.003</u>	<u>0.004</u>	<u>0.007</u>

Through two tables we could find that even in the early stage of the evolution, there always are some objectives that have a small value of R_f, such as the objectives f9 and f10 in the 100th iterations and the objectives f8, f9 and f10 after 200th iterations in DTLZ5 (2, 10), and the objectives f8, f9 and f10 in DTLZ5 (3, 10). While being the reason why the algorithm in references [19] and [17] could achieve good effect, this situation universally exists in the many-objective optimization problems. This situation indicates that we could retain the non-redundant objectives that have a great contribution to the dominance relationship, even though the population hasn't converged to the Pareto front and there are redundant objectives kept. For example, in DTLZ5 (2, 10), if we set the threshold C of R_f to 0.8, only 4 objectives could be identified as redundant objectives and removed in the first statistical analysis. As a result, the redundant objectives couldn't be removed totally during the analysis of uniform Pareto optimal solutions. The PCA analysis is still in need to remove the redundant objectives left. Because the redundant objectives could not be removed at all, the final result is mainly determined by the PCA analysis. If the PCA analysis could not remove the redundant objectives totally, or mistakenly remove the non-redundant objectives, the effect of the algorithm will become bad.

2) DTLZ5 (I, 10) test problems

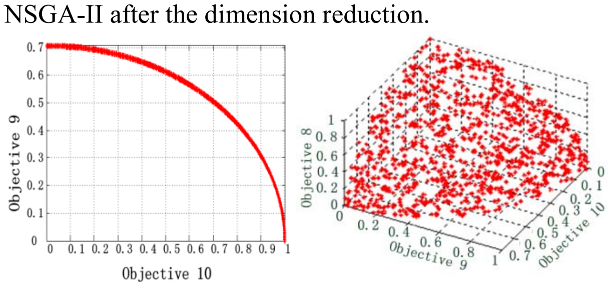
The test problems with 10 objectives are experimented. In order to clearly understand the whole process of dimension reduction and analyze the result of every sub process, the experiments in this paper keeps the results after every PCA analysis and the analysis of uniform Pareto optimal solutions. Take the problem DTLZ5 (2, 10) for instance, we describe the whole process of the algorithm as follow:

Table 3: Analysis of dimension reduction of DTLZ5(2, 10)

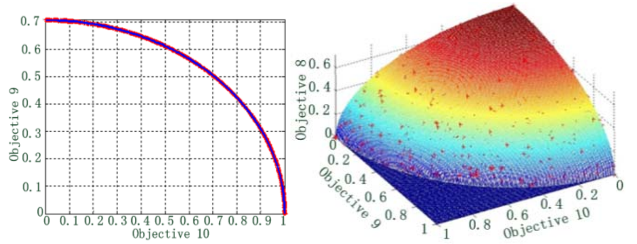
Pareto Analysis1	f1 f2 f3 f6 f7 f8 f9 f10
Pareto Analysis2	f3 f7 f2 f1 f9 f8 f10
Pareto Analysis3	f3 f2 f1 f8 f9 f10
Pareto Analysis4	f2 f1 f3 f8 f9 f10
PCA1	f10 f9
Pareto Analysis5	f9 f10
PCA2	f10 f9

As shown in table 3, the algorithm has carried 5 times analysis of uniform Pareto optimal solutions and twice PCA analysis. f4 and f5 are removed after the first Pareto analysis, after that, f6 and f7 are removed respectively after the second and third Pareto analysis. While there is no objective removed during the fourth and fifth Pareto analysis, the PCA analysis will be carried out to reduce the redundant objectives further. Since the smallest non-redundant objective set has been obtained after the first PCA analysis, the rest process is not necessary. But the situation is not always the same. The later experiments will prove that more than one PCA analysis should be carried out before acquiring the smallest non-redundant objective set. However, to assure that the redundant objectives have been removed totally, an extra PCA analysis has to be carried out.

Figure 2 (a) and figure 2 (b) gives the Pareto fronts of the problems DTLZ5 (2, 10) and DTLZ5 (3, 10) acquired by NSGA-II after the dimension reduction.



(a) (b)



(c) (d)

Figure 2: the Pareto fronts of DTLZ (I, 10) acquired by the NSGA-II after dimension reduction

Similarly, the mixture of the real Pareto fronts and the experimental results of both problems are shown respectively in figure 2 (c) and figure 2 (d). From both figures we can find that the experimental Pareto fronts have adequately converged to the real Pareto fronts, the results of the experiments and the real Pareto fronts have been coincident.

In order to compare the difference between the proposed algorithm and the algorithm in reference [8], we give the smallest non-redundant objective set and the time cost of the problems DTLZ5 (2, 10), DTLZ5 (3, 10) and DTLZ5 (5, 10) for both algorithms in table 4. The comparison of both characteristics shows great advantage of the proposed algorithm.

Table 4: The comparison of the proposed algorithm and PCA-NSGA-II in [8]

Test problems	The proposed algorithm		PCA-NSGA-II in [8]	
	result	iterations	result	iterations
DTLZ5(2,10)	f9 f10	2000	f9 f10	3000
DTLZ5(3,10)	f8 f9 f10	800	f8 f9 f10	3000
DTLZ5(5,10)	f6 f7 f8 f9 f10	800	f1 f7 f8 f9 f10	3000

Table 4 indicates that both algorithms can find the smallest non-redundant objective set. However, compared with the 3000 iterations of PCA-NSGA-II [8], the proposed algorithm performs only 2000 iterations before the smallest non-redundant objective set is identified in the problem DTLZ5 (2, 10). The time cost is shortened by about one third. In DTLZ5 (3, 10), the proposed algorithm performs only 800 iterations, which means that the time cost is shortened by about two third compared with the algorithm in reference [8]. As a result, the time complexity of the proposed algorithm is significantly decreased.

3) DTLZ5 (I, 20) test problems

In the DTLZ5 (*I*, 20) test problems, we take the situation when the parameter *I* equals to 2, 3 and 5 respectively, the results are shown in table 5.

Table5: The comparison of the proposed algorithm and PCA-NSGA-II in [8]

Test problems	The proposed algorithm		PCA-NSGA-II in [8]	
	results	iterations	results	iterations
DTLZ5(2,20)	f17 f20	2000	f17 f19 f20	5000
DTLZ5(3,20)	f16 f19 f20	2400	f16 f19 f20	6000
DTLZ5(5,20)	f14 f17 f18 f19 20	2800	f1 f17 f18 f19 f20	2000

We can find from the table 5 that the proposed algorithm could also find the smallest objective set with a relative low time cost when the number of objectives is as high as 20. However, because the analysis of uniform Pareto optimal solutions could not removes all the redundant objectives and the removed objectives are uncertain, the results vary among some smallest non-redundant objective sets. Figure 3 shows the restored Pareto front using the Pareto optimal solutions acquired by the NSGA-II after the dimension reduction.

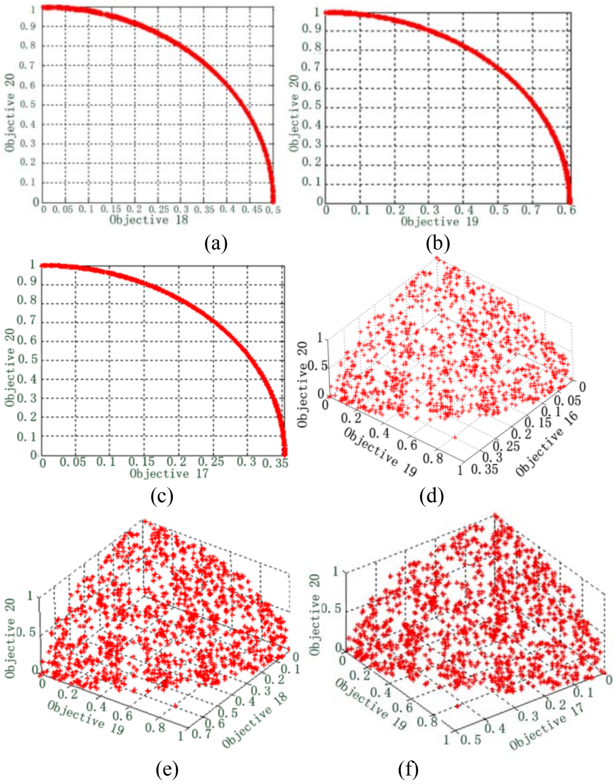


Figure 3: the Pareto fronts by different non-redundant objective set

In figure 3, the axes of figure 3 (a), (b) and (c) represent respectively the objectives [f19 f20], [f18 f20] and [f17 f20], while the Pareto fronts in these three figures corresponds to the same Pareto optimal solutions. These three figures show that any of these three objective sets could be the final result of the algorithm of dimension reduction. In the DTLZ5 (3, 20) problem in figure 3, the axes of the figure 3 (d), (e) and (f) represent respectively the objectives [f18 f19 f20], [f17 f19 f20] and [f16 f19 f20]. These three figures have the same situation as the DTLZ5 (2, 20) test problem. Figure 3 shows that the non-redundant objective sets are many in the optimization problems whose objectives have the characteristic of dimension reduction.

In the problem DTLZ5 (5, 20), we find that the time cost of the proposed algorithm exceeds that in reference [8], however, the proposed algorithm succeeds in finding the right non-redundant objectives, thus avoiding the bad convergence to the Pareto front caused by the objective f1.

V. CONCLUSION

The proposed algorithm combines two dimension reduction methods: the analysis of Pareto optimal solutions [17,19] and the PCA analysis of objectives [8,13]. The experiments are designed using the DTLZ5 (I, M) problem suit to test the performance of the proposed algorithm. The simulation results show that the proposed algorithm could acquire the smallest non-redundant objective set while significantly reducing the time complexity of algorithm. However, because NSGA-II is used to acquire the Pareto optimal solutions, the time cost is still very high. Finding a fast algorithm to instead the NSGA-II is a good way to reduce the time cost, which is also the future focus of the study.

REFERENCES

[1] RH Shang, LC Jiao, F Liu WP Ma, "A Novel Immune Clonal Algorithm for MO Problems", *IEEE Transactions on Evolutionary Computation*, 2012, 16 (1): 35-49.

[2] K Deb, "Multi-Objective Optimization Using Evolutionary Algorithms", *Chichester: JohnWiley & Sons*, 2011.

[3] K Deb, A Pratap, S Agarval, T Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II", *IEEE Trans. Evol. Comput*., 2002, 6(2):182-197.

[4] H Ishibuchi, Y Sakane, N Tsukamoto, Y Nojima. "Evolutionary Many-Objective Optimization by NSGA-II and MOEA/D with Large Populations", *IEEE International Conference on Systems, Man, and Cybernetics*, 2009: 1758-1763.

[5] DW Corne, JD Knowles, MJ Oates. "PESA-II: Region-based selection in evolutionary multiobjective optimization". *Proceedings of the Genetic and Evolutionary Computation Conference,* 2001: 283 -290

[6] E Zitzler, M Laumannns, L Thiele. "SPEA2: Improving the Strength Pareto Evolutionary Algorithm". TIK-Report 103. 2001.

[7] H Ishibuchi, N Tsukamoto, Y Nojima, "Evolutionary Many-Objective Optimization: A Short Review", *IEEE congress on Evolutionary Computation*, 2008.

[8] K Deb, D Saxena, "On Finding Pareto-Optimal Solutions Through Dimensionality Reduction for Certain Large-Dimensional Multi-Objective Optimization Problems", *Kangal report* 2005011, 2005.

[9] AV Lotov, VA Bushenkov, GK Kamenev. "Ineractive decision maps". *Boston, MA: Kluwer,* 2004.

[10] JB Tenenbaum, V de Silva, JC Langford. "A global geometric framework for nonlinear dimensionalty reduction". *Science*, 2000, 290(22):2319~2323.

[11] S Huband, P Hingston, L Barone, L While, "A review of multiobjective test problems and a scalable test problem toolkit", *IEEE Trans. Evol. Comput*., 2006, 10(5): 477-506.

[12] K Deb, L Thiele, M Laumanns, E Zitzler, "Scalable multiobjective optimization test problems", in *Proc. Congre. Evol. Comput.*, 2002, 1:825-830.

[13] K Saxena, JA Duro, A Tiwari, K Deb, QF Zhang, "Objective Reduction in Many-objective Optimization: Linear and Nonlinear Algorithms", *IEEE Transactions on Evolutionary Computation*, 2010, 1-22.

[14] CA Coello Coello. "Handling preferences in evolutionary multi-objective optimization: A survey." In: *Proc. of Congress on Evolutionary Computation. Piscataway: IEEE Service Center*, 2000, 1: 30-37.

[15] DW Corne, JD Knowles, "Techniques for highly multiobjective optimization: Some nondominated points are better than others", *in Proc. 9th Annu. Conf. GECCO*, 2007: 773-780.

[16] A Jaszkiewicz, R Slowinski, "the light beam seatch based multi-objective optimization using evolutionary algorithms", *Technical Report, No.2007005, Kanpur: Kanpur Genetic Algorithms Laboratory (KanGAL), Indian Insitute of Technology*, 2007.

[17] D Brockhoff, E Zitzler, "Are all Objectives Necessary? On Dimensionality Reduction in Evolutionary Multiobjective Optimization", *in Parallel Problem Solving from Nature – PPSN IX, ser. Lecture Notes in Computer Science*, 2006, 4193: 533-542

[18] AL Jaimes, CA Coello Coello, D Chakraborty, "Objective Reduction Using a Feature Selection Technique", *in Genetic and Evolutionary Computation Conference (GECCO)*, 2008: 673-680.

[19] H K Singh, A Isaacs, and T Ray, "A Pareto Corner Search Evolutionary Algorithm and Dimensionality Reduction in Many-Objective Optimization Problems", *IEEE Transactions on Evolutionary Computation*, 2011, 99: 1-18.

[20] FY Edgeworth, Mathematical Psychics, London, *UK: P. Keagan*, 1881.

[21] V Pareto. CD Politique. Lausanne, *Switzerland: F. Fouge*, 1896.

[22] K Zhou, LS Kang, CH Li, YC, MZ Liu, "Convergence Properties of E-Optimality Algorithms for Many-Objective Optimization Problems", *IEEE World Congress on Computational Intelligence (CEC 2008),* 2008: 472-477.