

On Finding Pareto-Optimal Solutions Through Dimensionality Reduction for Certain Large-Dimensional Multi-Objective Optimization Problems

Kalyanmoy Deb and Dhish Kumar Saxena
Kanpur Genetic Algorithms Laboratory (KanGAL)
Indian Institute of Technology Kanpur
Kanpur, PIN 208016, India
{deb,dksaxena}@iitk.ac.in,
<http://www.iitk.ac.in/kangal>

KanGAL Report Number 2005011

Abstract

Many real-world applications of multi-objective optimization involve a large number (10 or more) of objectives. Existing evolutionary multi-objective optimization (EMO) methods are applied only to problems having smaller number of objectives (about five or so) for the task of finding a well-representative set of Pareto-optimal solutions, in a single simulation run. Having adequately shown this task, EMO researchers/practitioners must now investigate if these methodologies can really be used for a large number of objectives. The major impediments in handling large number of objectives relate to stagnation of search process, increased dimensionality of Pareto-optimal front, large computational cost, and difficulty in visualization of the objective space. These difficulties, do and would continue to persist, in M -objective optimization problems, having an M -dimensional Pareto-optimal front. In this paper, we propose an EMO procedure for solving those large-objective (M) problems, which degenerate to possess a lower-dimensional Pareto-optimal front (lower than M). Such problems may often be observed in practice, as ones with similarities in optimal solutions for different objectives – a matter which may not be obvious from the problem description. The proposed method is a principal component analysis (PCA) based EMO procedure, which progresses iteratively from the interior of the search space towards the Pareto-optimal region by adaptively finding the correct lower-dimensional interactions. The efficacy of the procedure is demonstrated by solving up to 30-objective optimization problems.

Keywords: Redundancy in objective functions, principal component analysis, multi-objective optimization, large-objective problems.

1 Introduction

A multi-objective optimization task involving multiple conflicting objectives ideally demands finding a multi-dimensional Pareto-optimal front. Although the classical methods have dealt with finding one preferred solution with the help of a decision-maker [20], evolutionary multi-objective optimization (EMO) methods have been attempted to find a representative set of solutions in the Pareto-optimal front [6]. The latter approach is somewhat new and mostly applied to two or three objective problems. Further, it is debatable, if it is worth utilizing EMO methods to solve a large number of conflicting objectives (such as 10 or more objectives) for finding a representative set of Pareto-optimal solutions, for various practical reasons. First, visualization of a large-dimensional

front is certainly difficult. Second, an exponentially large number of solutions would be necessary to represent a large-dimensional front, thereby making the solution procedures computationally expensive. Third, it would certainly be tedious for the decision-makers to analyze so many solutions, to finally be able to chose a particular region, for picking up solution(s). These are some of the reasons why EMO applications have also been confined to a limited number of objectives.

However, most practical optimization problems involve a large number of objectives. While formulating an optimization problem, designers and decision-makers prefer to put every performance index related to the problem as an objective, rather than as a constraint, thereby totalling a large number of objectives. Often it is not, obvious or intuitive, if any two given objectives are non-conflicting. Furthermore, in some problems, even though, apparently there may exist a conflicting scenario between objectives, for two randomly picked feasible solutions, there may not exist any conflict between the same objectives, for two solutions picked from near the Pareto-optimal front. That is, although a conflict exists elsewhere, some objectives may behave in a non-conflicting manner, near the Pareto-optimal region. In such cases, the Pareto-optimal front will be of dimension lower than the number of objectives. It is in these problems, that, there may still exist a benefit of using an EMO to find a well-represented set of Pareto-optimal solutions, although the search space is formed with a large number of objectives.

In this paper, we address solving such problems by using a principal component based approach coupled with an EMO – the elitist non-dominated sorting GA or NSGA-II [8]. The proposed procedure iteratively identifies redundant objectives from the solutions obtained by NSGA-II and eliminates them from further consideration. On a number of test problems involving as many as 30 objectives, the proposed methodology is able to identify the correct number of objectives and correct combination of objectives in most problems. The results of this study are interesting and should encourage more such techniques of dimensionality reduction, in the context of multi-objective optimization.

2 EMO for Many Objectives

Figure 1 shows the number of citations employing a given number of objective functions [5]. The

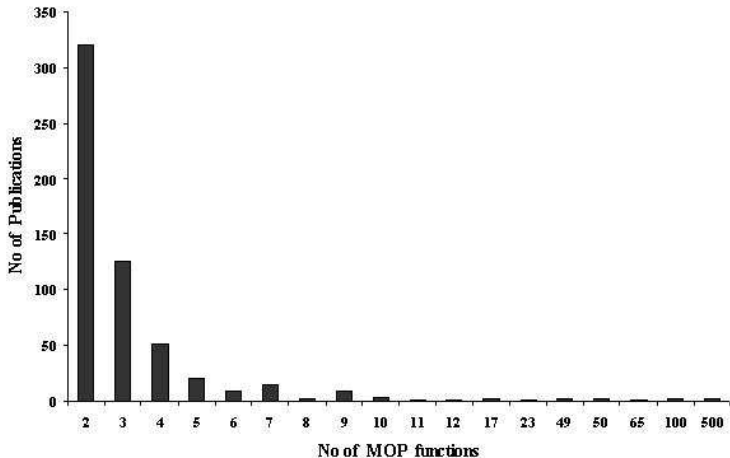


Figure 1: EMO citations by number of objectives (till year 2001), taken from [5].

overwhelming majority use only two objective functions, most probably for the ease of their solution principles. Some use three to nine objectives, and only a few tried beyond 10 objectives. The studies using more than 10 objectives often employed a single-objective optimization method by converting the multi-objective optimization problem into a single objective. In this approach, both methods

of using a linear combination of weights and the ϵ -constraint method [20] were used. Notably, the highest number of conceptually different-implemented objective functions are seven and nine, by Fonseca and Fleming [12, 13] and by Sandgren [21], respectively.

The sparse and fragmentary research efforts for a large number of objectives are indicative of the potential difficulties, a number of which we discuss in the following section.

3 Difficulties with Many Objectives

When the number of objectives are many, a number of difficulties arise for an efficient use of a population-based optimization procedure for finding a representative set of Pareto-optimal solutions:

1. For an obvious reason, if the dimensionality of the objective space increases, then in general, the dimensionality of the Pareto-optimal frontier increases.
2. Due to above reason, there is a greater probability of having any two arbitrary solutions to be non-dominated to each other, as simply now there are many objectives in which a trade-off (one is better in one objective but worse in any other objective) can occur. While dealing with a finite-sized population-based approach, the proportion of non-dominated solutions in the population increase. Since EMO algorithms provide more emphasis to the non-dominated solutions, a large proportion of the old population gets emphasized, thereby not leaving much room for new solutions to be included in the population. This, in effect, reduces the selection pressure for the better solutions in the population and the search process slows down.
3. Due to the reason stated in item 1 above, an exponential number of points will be necessary to represent the Pareto-optimal frontier, as an increase by one objective, in general, will cause the dimension of the Pareto-optimal front to increase by one. Thus, if N points are needed for adequately representing a one-dimensional Pareto-optimal front, $O(N^M)$ points will necessary to represent an M -dimensional Pareto-optimal front (which is an M -dimensional hyper-surface).
4. Last but not the least, there lies a great difficulty in visualizing a Pareto-optimal frontier which is more than three-dimensional. Finding a higher-dimensional Pareto-optimal surface is one important matter, but visualizing it for a proper decision-making is another equally important matter. Although sophisticated methods, such as the use of decision maps [19] or geodesic maps [23] do exist, they all require huge volume of points. Hence, more research effort must be spent in this direction, if many-objective problems are to be solved, for finding a representative set of Pareto-optimal solutions in a routine manner.

To illustrate the difficulties in solving many-objective optimization problems using a standard EMO procedure, here we choose the well-known DTLZ5 problem having M objectives with a small change in its formulation, as shown below [10]:

$$\left. \begin{array}{l}
 \text{Min } f_1(\mathbf{x}) = (1 + 100g(\mathbf{x}_M)) \cos(\theta_1) \cos(\theta_2) \cdots \cos(\theta_{M-2}) \cos(\theta_{M-1}), \\
 \text{Min } f_2(\mathbf{x}) = (1 + 100g(\mathbf{x}_M)) \cos(\theta_1) \cos(\theta_2) \cdots \cos(\theta_{M-2}) \sin(\theta_{M-1}), \\
 \text{Min } f_3(\mathbf{x}) = (1 + 100g(\mathbf{x}_M)) \cos(\theta_1) \cos(\theta_2) \cdots \sin(\theta_{M-2}), \\
 \vdots \\
 \text{Min } f_{M-1}(\mathbf{x}) = (1 + 100g(\mathbf{x}_M)) \cos(\theta_1) \sin(\theta_2), \\
 \text{Min } f_M(\mathbf{x}) = (1 + 100g(\mathbf{x}_M)) \sin(\theta_1), \\
 \text{where } g(\mathbf{x}_M) = \sum_{x_i \in \mathbf{x}_M} (x_i - 0.5)^2, \\
 \theta_i = \begin{cases} \frac{\pi}{2} x_i, & \text{for } i = 1, \dots, (I - 1), \\ \frac{\pi}{4(1+g(\mathbf{x}_M))} (1 + 2g(\mathbf{x}_M)x_i), & \text{for } i = I, \dots, (M - 1), \end{cases} \\
 0 \leq x_i \leq 1, \quad \text{for } i = 1, 2, \dots, n.
 \end{array} \right\} \quad (1)$$

Here we choose $k = |\mathbf{x}_M| = 10$, so that total number of variables is $n = M + k - 1$. We call this problem DTLZ5(I, M), where I denotes the dimensionality of the Pareto-optimal surface and M is the number of objectives in the problem. The variables x_1 till x_{I-1} take independent values and are responsible for the dimensionality of the front, whereas other variables (x_I till x_{M-1}) take a fixed value of $\pi/4$ for the Pareto-optimal points. A nice aspect of this test problem is that by simply setting I to an integer between two and M , the dimensionality (I) of the Pareto-optimal front can be changed. For $I = 2$, a minimum of two objectives (f_M and any other objective) will be enough to represent the correct Pareto-optimal front. Other properties of this problem is that the Pareto-optimal front is non-convex and follows the relationship: $\sum_{i=1}^M (f_i^*)^2 = 1$.

We employ the elitist non-dominated sorting GA or NSGA-II [8] to solve DTLZ5(2,10) and DTLZ5(3,10) problems here. We use a population of size 1,000 and NSGA-II is run for 5,000 generations to provide a reasonable computational effort. The SBX crossover with a probability of 0.9 and index 5 and polynomial mutation with a probability of $1/n$ and index of 50, are used here. Figure 2 plots the number of solutions found close to the Pareto-optimal front (having $g(\mathbf{x}_M) \leq 0.01$) with generations, for DTLZ5(2,10). Five runs are taken from different initial

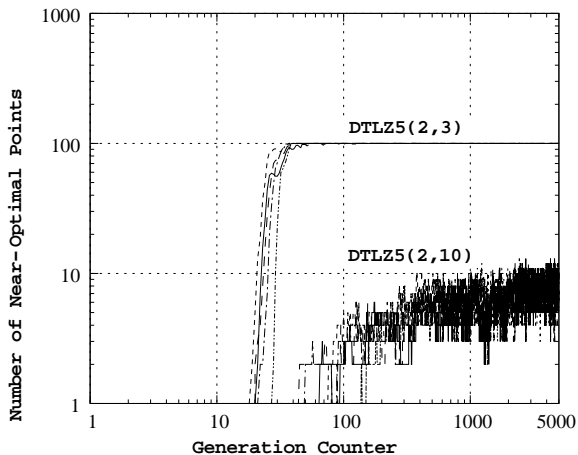


Figure 2: Only about 1% population members come close to the Pareto-optimal front in DTLZ5(2,10) with original NSGA-II.

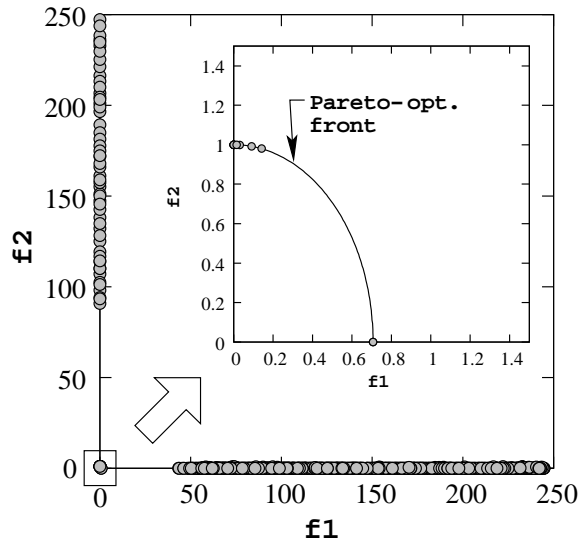


Figure 3: Population members at the end of 5,000 generations are still not close to the Pareto-optimal front for DTLZ5(2,10).

populations and results are plotted. For a comparison, the same quantity is calculated for the three-objective DTLZ5(2,3) problem with a population of size 100 and is plotted on the figure. For 10-objective DTLZ5 problem, only about 10 out of 1,000 population members (about 1%) could come close to the Pareto-optimal front at the end of 5,000 generations. On the other hand, NSGA-II with a population of size 100, run for 200 generations, is enough to bring almost 100% population members near the Pareto-optimal front for the three-objective DTLZ5 problem. The final population obtained with DTLZ5(2,10) is shown in Figure 3. The figures demonstrate the effect of dimensionality on the performance of NSGA-II, run even with a large population size, for a large number of generations.

Similar results are obtained for DTLZ5(3,10), in which the Pareto-optimal front is a three-dimensional surface. Figure 4 shows that only about 1% population members are close to the Pareto-optimal front, whereas Figure 5 shows that these 1% points are concentrated at a couple of places on or near the Pareto-optimal front at the end of 5,000 generations.

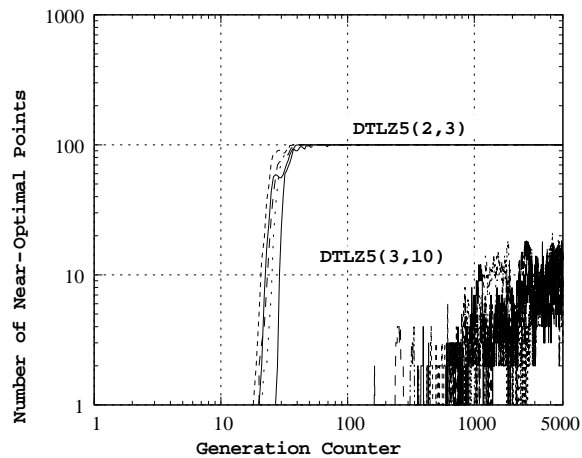


Figure 4: Only about 1% population members come close to the Pareto-optimal front in DTLZ5(3,10) with original NSGA-II.

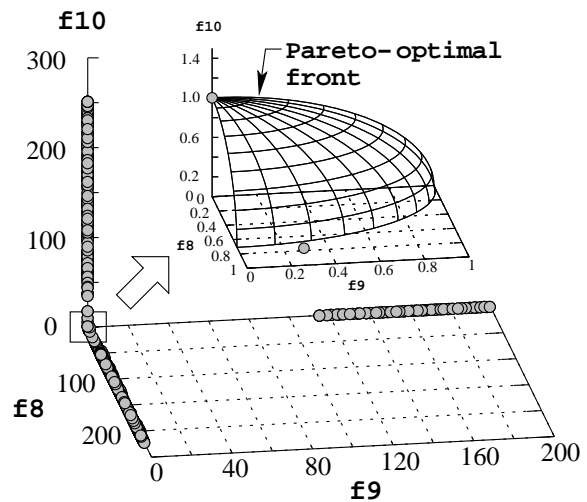


Figure 5: Population members at the end of 5,000 generations are still not close to the Pareto-optimal front for DTLZ5(3,10).

These simulation results, support our earlier argument on the difficulties associated with finding the complete Pareto-optimal front by an EMO in many-objective optimization problems. In order to solve such problems, researchers have suggested different procedures. In the following subsection, we describe a few such attempts.

3.1 Past Studies to Solve Many-Objective Problems

One simple approach followed since the early days of solving multi-objective optimization problems is a scalarization technique, in which all objectives are converted into a single composite objective by the weighted sum approach or the Tchebyshev method or the ϵ -constraint method or others [20]. With these procedures, the total number of objectives is not that big a matter, as in any case, all objectives are somehow combined to form a single objective. To generate a number of Pareto-optimal solutions, such a scalarization process may be repeated with different weight-vectors or ϵ -vectors [6]. As can be argued [22], such a multi-optimization strategy causes each optimization to be executed independent to each other, thereby losing the parallel search ability often desired in solving complex optimization problems, for an efficient and judicious use of computational resources.

In the context of EMO methodologies in handling a large number of objectives, Khare et al. [18] investigated the efficacy of three EMOs – NSGA-II, SPEA2 and PESA for their scalability to number of objectives. These algorithms were tested on two to eight-objective test problems for three performance criteria – convergence to the efficient frontier, diversity of obtained solutions and computational time. It was the first time that these EMO methodologies which work very well on fewer objectives (two or three-objective problems) showed vulnerability to problems having a relatively large number of objectives. Moreover, no one algorithm was found to outperform the other two, according to all three performance measures.

Realizing that it gets increasingly difficult to find multiple Pareto-optimal solutions for a large number of objectives, researchers have taken a different path. Although, a well-represented set of Pareto-optimal solutions provides a good idea of the optimal trade-off in a problem, thereby easing and helping in the deciding on a particular region in the trade-off frontier more reliably, the ultimate goal of these approaches is to concentrate on a specific solution or region on the Pareto-optimal front. In other words, such optimization approaches have to involve a decision-maker at some stage. For a large number of objectives, it may be better to involve a decision-maker right in the

beginning of the optimization process and instead of finding the optimal solutions corresponding to a specific weight-vector or ϵ -vector, an EMO methodology can be used to find a set of solutions in the preferred region on the Pareto-optimal frontier. This beats the dimensionality problem described earlier by not finding points on the complete high-dimensional Pareto-optimal frontier and also providing the decision-maker with a set of solutions in a region of interest to her or him. There exist a number of directions of performing this task:

1. Branke [4] and later Branke and Deb [7] suggested guided-domination principle and biased sharing approach for finding a preferred set of solutions on the Pareto-optimal front. By simply specifying the extreme pair-wise trade-off information about objectives or by providing a relative weightage of different objectives, a set of solutions near the preferred region on the Pareto-optimal front were possible to obtain. A recent study [2] suggests a projection-based technique for arriving at a biased distribution of Pareto-optimal solutions.
2. Farina and Amato [11] defined the domination principles in a different and interesting manner. They refer a solution $\mathbf{x}^{(1)}$ to be dominant over another solution $\mathbf{x}^{(2)}$, if $\mathbf{x}^{(1)}$ is better than $\mathbf{x}^{(2)}$ in more number of objectives. If thought carefully, the Pareto-optimal solutions corresponding to this definition will only be a portion of the true Pareto-optimal front, thereby reducing the number and dimensionality of target solutions. Preference information in such a technique can be added by giving more weightage for betterment to the objectives of interests to the DM. They also extended the idea and a fuzzy-dominance relationship was introduced. Although the dominance relationship was defined with a simple count on the number of better objectives, a fuzzy membership function was used in deciding if a solution will dominate another solution or not.

Although the above approaches involve some DM-dependent information, there are some DM-independent issues which can be utilized in almost all problems to reduce the size the dimensionality of the Pareto-optimal frontier. In some problems, there exists a ‘knee’ solution, from which a transfer to a neighbouring trade-off solution demands a large sacrifice in at least one objective to make a small gain in another objective. Such solutions are of utmost important to a DM and a recent study [3] have suggested an EMO-based methodology to find only the knee points in a problem, instead of the complete Pareto-optimal frontier. Similarly, another study [9] dealt with finding the robust frontier, the solutions of which are less sensitive to parameter fluctuations. Rather than finding the true Pareto-optimal frontier, in such cases a robust frontier is sought.

Thus, for handling a large number of objectives, the existing EMO methodologies are inadequately tested. Most studies have restricted the search to find only a preferred region on the Pareto-optimal front. Hence, an important question still remains for the EMO researchers to answer: Are EMO methodologies computationally viable in finding a well-representative set of points on the complete Pareto-optimal frontier? In this paper, we address this important question and attempt to provide a solution methodology for handling certain many-objective optimization problems.

4 Proposed Methodology

It is clear from the simulation results performed on 10-objective problems in Section 3 that the original NSGA-II is unable to come close to the Pareto-optimal front, even with a comparatively large ($5(10^6)$) number of function evaluations, on two problems having a Pareto-optimal front, of dimension lower than that of objective space. Although the target is a two or a three-dimensional frontier, NSGA-II procedure of a setting a direction for convergence through non-dominated sorting of its population members and maintaining a wide diversity of solutions through crowding distance sorting was unable to handle a 10-dimensional objective space. This provides some indication of the

‘curse of dimensionality’ in EMO methodologies for finding a representative set of Pareto-optimal points, for large-objective problems.

However, there may exist many-objective problems in practice, which have redundant objectives, that is, although the problem has M objectives, the Pareto-optimal front involves a much lower dimensional interactions. Such a reduction in the dimensionality may happen gradually from the region of random solutions towards the Pareto-optimal region or the entire search space may have such a structure. The later case implies that there exist some objectives in the problem formulation which are non-conflicting to each other. Often, such information about the nature of variation of objective values are not intuitive to a designer/DM. However, the former type of problems may be more likely to exist in practice. In this case, the non-conflicting nature of objectives does not exist for all solutions in the search space. For randomly picked solutions the objectives may be all conflicting and there is an M -dimensional interaction, however for solutions close to the Pareto-optimal front (special solutions), the optimality conditions of some objectives are similar and they behave in a non-conflicting manner. The DTLZ5 problem constructed earlier allows a way to simulate this second scenario with a flexibility of controlling the reduction in dimension.

When such a structure or a more general structure facilitating a dimensionality-reduction exists in a problem and when a relatively low-dimensional front (having 2 to 5-objective interactions) becomes the target Pareto-optimal front (although the problem may have a large number of objectives, such as 10 or more), it may still be beneficial to use an EMO to find the true Pareto-optimal front. In Section 3, we have shown that the original NSGA-II is unable to find the true two or three-dimensional Pareto-optimal front in a 10-objective problem in a reasonable number of function evaluations.

In this section, we suggest a PCA based NSGA-II procedure for finding the true Pareto-optimal front for such problems facilitating a reduction in objectives.

4.1 Past Research on Handling Redundant Objectives

Given the fact that not many studies have been performed in the total realm of high-dimensional multi-objective problems, it is only natural, not to expect many studies on the determination of redundancy in many-objective problems, in particular. One of the earlier studies by Gal and Leberling [14] defined an objective as redundant if it is linearly dependent to other objectives. In another study, Gal and Hanne [15] termed an objective ‘non-essential’, if dropping it does not affect the set of efficient solutions. However, a significant study in this area has been done by Agrell [1], who argued that the ultimate justification for multi-criterion models is in the inherent conflicts in decisions and that in absence of a conflict, the ranking of criterion is just a matter of scaling. He also proposed a probabilistic method based on correlation and tested it for a non-linear problem.

Jensen [17] suggested a procedure of introducing additional objectives one by one, as and when a multi-objective optimization run gets stuck to a set of solutions. Although the purpose of this study was not to solve a large number of objectives, the use of many objectives introduced new diversity in the population so that an evolutionary algorithm can proceed near to the true optimum without getting stuck anywhere in the search space. Another recent study [16] used a correlation matrix to identify the redundant objectives, if any, in a four-objective optimization problem. Later, the NSGA-II procedure is used with the reduced set of three objectives to find the resulting Pareto-optimal front.

5 Proposed PCA-NSGA-II Procedure

It is important to realize, that given an M -objective problem, if the Pareto-optimal front is less than M -dimensional, then some of the objectives are *redundant*. We have targeted here the determination of such lower-dimensional interactions among objectives to determine the true Pareto-optimal front.

In this regard, we use the principal component analysis (PCA) procedure coupled with the NSGA-II method. The proposed - combined procedure works iteratively from the interior of the search space and iteratively moves towards the Pareto-optimal region and adaptively attempts to find the correct lower-dimensional interactions. Before we discuss the PCA-NSGA-II procedure, let us make a brief summary of the PCA analysis procedure.

5.1 A Brief Note on Principal Component Analysis (PCA)

Principal component analysis is a statistical tool for multi-variate analysis. It reduces the dimensionality of a data set with a large number of interrelated variables, retaining as much variation of the data set, as possible. This reduction in dimension is achieved by transformation of the original variables to a new set of variables, referred as *principal components*. These components are *uncorrelated* and *ordered*. The later is an important property, since it is this, which justifies as to why first few principal components can be considered to still retain most of the variation in the data set. The computation of principal components is usually posed as an eigenvalue-eigenvector problem, whose formulation, solution and interpretation of results, form the basis of the proposed algorithm.

To perform a PCA analysis, first the given data set must be in the *standardized* form, which means that the centroid of the whole data set be zero. This can be achieved by subtracting the mean from each measurement. The initial data set is represented in the form of a matrix $\mathbf{D}=(D_1, D_2, \dots, D_M)^T$, where D_i is the i -th measurement. Each row of this matrix represents a particular measurement, while each column corresponds to a time sample (or an experimental trial). The number of measurements is the *dimension* of the data set. Mathematically, each data sample is a vector in M -dimensional space, where M is the number of measurements. The matrix formed by the standardized data is stored in the matrix $\mathbf{X}=(X_1, X_2, \dots, X_M)^T$. One way to identify redundant objectives would be to calculate the covariance variance (\mathbf{V}) or the correlation matrix (\mathbf{R}):

$$\text{Covariance Matrix } (\mathbf{V}): \quad V_{ij} = \frac{X_i X_j^T}{M - 1}, \quad (2)$$

$$\text{Correlation Matrix } (\mathbf{R}): \quad R_{ij} = \frac{V_{ij}}{\sqrt{V_{ii} \cdot V_{jj}}}. \quad (3)$$

Principal components are nothing but the eigenvectors of either of these two matrices. The usage of the correlation matrix is advantageous when measurements are in different units and is most suitable here. The eigenvector corresponding to the largest eigenvalue is referred as the first principal component, one corresponding to the second largest eigenvalue is called the second principal component and so on.

We now discuss how we could utilize the principal components for reducing the dimension of the data set. The first principal component (vector) signifies a hyper-line in the M -dimensional space which goes through the centroid and also minimizes the square of the distance of each point to the line. Thus, in some sense, the line is as close to all data as possible. Equivalently, the line goes through the maximum variation in the data. Furthermore, the second principal component (vector) again passes through the centroid and the maximum variation in the data, but with an additional constraint. It must be completely uncorrelated (at right angle or "orthogonal") to the first PCA axis. The advantage of working these PCA axes is that these axes can now be used to define planar sections through the data set. If one makes a slice through the cloud of data points using the plane defined by the first two PCA axes and projects all of the data points onto this plane, then it becomes a two-dimensional representation of the data retaining the maximum variation, contained in the multi-variate data set. As more and more principal components are considered, data with lesser variance are revealed.

5.2 PCA Analysis for Multi-Objective Optimization

In the context of a M -objective optimization problem being solved using N population members, the initial data matrix will be of size $M \times N$. This matrix is then converted into the standardized form and the correlation matrix is computed. To illustrate this procedure, we consider $N = 10$ solutions computed for a three-objective ($M = 3$) problem DTLZ5(2,3). The correlation matrix \mathbf{R} obtained for the data set is shown in Table 1. It can be observed from this matrix that the first and third objectives are negatively correlated, that is, they are in conflict to each other. The same is true for the second and third objectives as well. Hence, while the first and second objectives are non-conflicting to each other, each of them is in conflict with the third objective. Thus, in this simple case it can be concluded from this matrix that one of the first or second objective is redundant in this problem. However, for a large number of objectives and to a more complex

Table 1: A PCA analysis of 10 data points used for DTLZ5(2,3) is shown for illustration.

Correlation Matrix		
1.0000	0.9997	-0.9182
0.9997	1.0000	-0.9190
-0.9182	-0.9190	1.0000
Eigenvalues		
2.8918	0.0003	0.1078
Eigenvalues by proportion		
0.9639	0.0001	0.0359
Rank=1	Rank=3	Rank=2
Eigenvectors		
0.5828	0.7055	0.4033
0.5830	-0.7087	0.3973
-0.5661	-0.0035	0.8243
PCA1	PCA3	PCA2

problem, such a clear analysis may not be possible from a $M \times M$ matrix of real numbers. In the following paragraph, we suggest a PCA based procedure to reduce our attention to a smaller number of objectives. Finally we return to the correlation matrix to cross-check the conflicting nature of the objectives and if possible reduce the dimensionality of the problem further.

5.2.1 Eigenvalue Analysis for Dimensionality Reduction

For the example problem shown in Table 1, we compute the eigenvalues of the correlation matrix \mathbf{R} . The eigenvalues are also shown ranked in the decreasing order of their magnitudes. Corresponding eigenvectors are also shown in the table. The first principal component (eigenvector $(0.5828, 0.5830, -0.5661)^T$, corresponding to the largest eigenvalue) is designated as ‘PCA1’ in Table 1. The first component of this vector denotes the contribution of first objective function towards this vector, the second element denotes the contribution of the second objective, and so on. For a three-objective problem, the three contributions could be treated as the direction cosines defining a directed-ray in the objective space. These values confirm the relative correlations and the nature of conflicts among the objectives.

As discussed above, the elements of the principal component denotes the relative contribution of each objective. A positive value denotes an increase in objective value moving along this principal component (axes) and a negative value denotes a decrease. Thus, if we consider the objectives corresponding to the most-positive and most-negative elements of this vector, they denote the

objectives which have the maximum contribution for an increase or a decrease in the principal component. Thus, by picking the most-negative and the most-positive elements from a principal component, we can obtain the two most important conflicting objectives. In the above example, f_2 and f_3 are observed to be the two most critically conflicting objectives.

5.2.2 Effect of Multiple Principal Components

In the above manner, each principal component can be analyzed for the two main objectives causing a conflict and the information about the overall conflicting objectives can be collected. But here, we suggest a procedure which starts with analyzing the the first principal component and then proceed to analyzing the second principal component and so on, till all the significant components are considered. For this purpose, we pre-define a threshold cut (TC) and when the cumulative contribution of all previously principal components exceeds TC, we do not analyze any more principal components. This will not bring in less important (in terms of a conflict) objectives. For this purpose, we consider the relative contribution of the corresponding eigenvalue of each principal component. Table 1 computes the proportion of contribution of each principal component. If a TC equal to 95% is used, only the first principal component will be analyzed, as PCA1 contributes 96.39% of all principal components. Thus, we do not consider any further PCAs and declare the second and third objectives as important objectives to this problem. This way, the first objective is determined to be redundant and an EMO procedure can be applied to solve the two-objective problems (f_2 and f_3), instead of all three objectives.

To emphasize the contributions of the eigenvectors, we use the matrix $\mathbf{R}\mathbf{R}^T$, instead of \mathbf{R} to find the eigenvalues and eigenvectors. This does not change the eigenvectors of the original correlation matrix \mathbf{R} , but the eigenvalues get squared and more emphasized.

Let us now discuss the importance of choosing an appropriate threshold cut (TC) parameter. If a too high (close to 100%) TC is used, many redundant objectives may be chosen by the above analysis, thereby defeating the purpose of doing the PCA analysis. On the other hand, if too small a value is chosen, important objectives may be ignored, thereby causing an error in the whole study. However, we emphasize that to make a reliable study a high TC value (around 95% or so) may be better. This will allow a lesser number of objectives to be redundant in each iteration of the PCA analysis, reducing the risk of losing out, an otherwise important objective. Thereafter, if an iterative scheme of using a combined PCA and NSGA-II procedure (described in the next subsection) is adopted, a reliable methodology can be obtained. The choice of TC can also be based on the relative magnitudes of the eigenvalues. If the reduction in two consecutive eigenvalues is more than a predefined proportion, no further principal component may be considered. This decision can also be based on the choice of the decision-maker (DM). If the DM wishes to include certain preferred objectives in the EMO procedure, the principal component analysis can be continued till all such preferred objectives are chosen. Many such possibilities exist and are worth experimenting with, but here we simply choose a fixed TC of 0.95 in all simulations and do not consider analyzing any further principal components when the cumulative contribution exceeds 0.95. All objectives gathered in this manner are optimized by an EMO procedure.

To make the dimensionality-reduction procedure effective and applicable to various scenarios, we suggest the following additional procedure. For the first principal component, we choose both objectives contributing the most positive and most negative objectives. For subsequent principal components, we first check if the corresponding eigenvalue is greater than 0.1 or not. If not, we only choose the objective corresponding to the highest absolute element in the eigenvector. If yes and also if the cumulative contribution of eigenvalues is less than TC, we consider various cases. If all elements of the eigenvector are positive, we only choose the objective corresponding to the highest element. If all elements of the eigenvector are negative, we choose all objectives. Otherwise, if the value of the highest positive element (p) is less than the absolute value of the most-negative element (n), we consider two different scenarios. If $p \geq 0.9|n|$, we choose two

objectives corresponding to p and n . On the other hand, if $p < 0.9|n|$, we only choose the objective corresponding to n . Similarly, if $p > |n|$, then we consider two other scenarios. If $p \geq 0.8|n|$, we choose both objectives corresponding to p and n . On the other hand, if $p < 0.8|n|$, we only choose the objective corresponding to p .

5.2.3 Final Reduction Using the Correlation Matrix

Hopefully, the above procedure identifies most of the redundant objectives dictated by the data set. To consider if more reduction in the number of objectives is possible, we then return to a *reduced* correlation matrix (only columns and rows corresponding to non-redundant objectives) and investigate if there still exists a set of objectives having identical positive or negative correlation coefficients with other objectives and having a positive correlation among themselves. This will suggest that any one member from such group would be enough to establish the conflicting relationships with the remaining objectives. In such a case, we retain the one which was chosen the earliest (corresponding to the larger eigenvalue) by the PCA analysis. Other objectives from the set are not considered further.

5.3 Overall PCA-NSGA-II Procedure

We now ready to present the overall PCA-NSGA-II procedure.

Step 1: Set an iteration counter $t = 0$ and initial set of objectives $\mathbb{I}_0 = \{1, 2, \dots, M\}$.

Step 2: Initialize a random population for all objectives in the set \mathbb{I}_t , run an EMO, and obtain a population P_t .

Step 3: Perform a PCA analysis on P_t using \mathbb{I}_t to choose a reduced set of objectives \mathbb{I}_{t+1} using the predefined TC. Steps of the PCA analysis are as follows:

1. Compute the correlation matrix using equation 3.
2. Compute eigenvalues and eigenvectors and choose non-redundant objectives using the procedure discussed in Sections 5.2.1 and 5.2.2.
3. Reduce the number of objectives further, if possible, by using the correlation coefficients of the non-redundant objectives found in item 2 above, using the procedure discussed in Section 5.2.3.

Step 4: If $\mathbb{I}_{t+1} = \mathbb{I}_t$, stop and declare the obtained front. Else set $t = t + 1$ and go to Step 2.

Thus, starting with all M objectives, the above procedure iteratively finds a reduced set of objectives, by analyzing the obtained non-dominated solutions by an EMO procedure. When no further objective reduction is possible, the procedure stops and declares the final set of objectives and corresponding non-dominated solutions.

We realize that the above procedure of dimensionality reduction has not much of a meaning for those problems which have an exactly M -dimensional Pareto-optimal front. In such a scenarios, it is expected that the proposed algorithm will discover all objectives to be important in the very first iteration, thereby not performing any dimensionality reduction. However, even in this case, we highlight that the proposed procedure will establish a relative order of importance of the objectives, by the PCA analysis, which may provide additional information to a decision-maker.

6 Simulation Results

We now show the simulation results obtained with the iterative PCA-NSGA-II procedure described above, on a number of test problems having a varying number of objectives. For this purpose, we

first use DTLZ5(I,M) problems. In all problems, we use a population of size of 800 and is run for 1,000 generations with the following parameters: crossover probability of 0.9 and index of 5 and mutation probability of 0.1 and index 50.

6.1 Problem DTLZ5(2,10)

This problem involves 10 objectives, but eight of them are redundant. The Pareto-optimal front can be found with f_{10} and any other objective function, though with different scales. At the end of the first iteration, eigenvalue analysis suggested 6 objectives - f_1, f_2, f_5, f_8, f_9 and f_{10} , to be non-redundant (critical). These six objectives constitute the *reduced* correlation matrix, using which, the final reduction in objectives is attempted (as discussed in Section 5.2.3). Identical correlations of f_5 and f_8 , with the rest four, of the six objectives, suggest possible elimination of any one of these. However, as f_5 was chosen from the consideration of PCA1-the first principal component (for brevity, the details are not shown, though), it deems fit to be retained, while f_8 , to be eliminated. The correlation matrix for all objectives is shown below:

	$\cdot f_1$	$\cdot f_2$	f_3	f_4	$\cdot f_5$	f_6	f_7	$\cdot f_8$	$\cdot f_9$	$\cdot f_{10}$
$\cdot f_1$	1.000	-0.153	0.175	0.194	<u>0.205</u>	0.205	0.191	<u>0.191</u>	0.074	-0.292
$\cdot f_2$	-0.153	1.000	0.155	0.256	<u>0.266</u>	0.263	0.262	<u>0.132</u>	0.193	-0.304
f_3	0.175	0.155	1.000	0.961	0.957	0.942	0.730	0.093	-0.139	-0.342
f_4	0.194	0.256	0.961	1.000	0.993	0.981	0.782	0.141	-0.133	-0.365
$\cdot f_5$	0.205	0.266	0.957	0.993	1.000	0.990	0.785	<u>0.146</u>	-0.128	-0.361
f_6	0.205	0.263	0.942	0.981	0.990	1.000	0.785	0.152	-0.137	-0.348
f_7	0.191	0.262	0.730	0.782	0.785	0.785	1.000	0.096	-0.183	-0.385
$\cdot f_8$	0.191	0.132	0.093	0.141	<u>0.146</u>	0.152	0.096	1.000	-0.135	-0.439
$\cdot f_9$	0.074	0.193	-0.139	-0.133	<u>-0.128</u>	-0.137	-0.183	<u>-0.135</u>	1.000	-0.499
$\cdot f_{10}$	-0.292	-0.304	-0.342	-0.365	<u>-0.361</u>	-0.348	-0.385	<u>-0.439</u>	-0.499	1.000

In all the correlation matrices, shown here, the functions adjudged non-redundant by eigenvalue analysis are symbolized with a preceding dot. Further, the column coefficients of those functions are marked underlined, which are potentially reducible using correlation analysis. In effect, at the end of the first iteration, f_5 stands as representative to f_3, f_4, f_6 and f_7 (eliminated by eigenvalue analysis) and also f_8 (eliminated by correlation analysis). Figure 6 shows the final NSGA-II population on a f_9 - f_{10} plot. Although most solutions are far away from the true Pareto-optimal front, they seem to lie on the two axes.

Thereafter, the second iteration was executed with the five objectives - f_1, f_2, f_5, f_9 and f_{10} , and eigenvalue analysis performed on final population, eliminated two objectives - f_1 and f_2 . When all 10 objectives were considered in the first iteration, the redundancy of these two objectives could not be established due to the presence of too many objectives, but when in the second iteration, five objectives were considered and optimized, these two stood eliminated. Though, in second iteration, correlation analysis played no role in elimination of f_1 and f_2 , the corresponding correlation matrix with all five objectives, is shown below, to highlight some interesting facts.

	f_1	f_2	$\cdot f_5$	$\cdot f_9$	$\cdot f_{10}$
f_1	<u>1.000</u>	<u>0.538</u>	-0.233	<u>0.622</u>	-0.405
f_2	<u>0.538</u>	<u>1.000</u>	-0.338	<u>0.681</u>	-0.383
$\cdot f_5$	<u>-0.233</u>	<u>-0.338</u>	1.000	<u>-0.410</u>	-0.337
$\cdot f_9$	<u>0.622</u>	<u>0.681</u>	-0.410	<u>1.000</u>	-0.605
$\cdot f_{10}$	<u>-0.405</u>	<u>-0.383</u>	-0.337	<u>-0.605</u>	1.000

Consider a situation, if all the five objectives were declared non-redundant by eigenvalue analysis and it were upto the correlation matrix analysis to eliminate some objectives. It can be observed that objectives f_1, f_2 and f_9 are positively correlated amongst each other, while each is negatively correlated to the other two objectives - f_5 and f_{10} . Thus, only one of them could have been retained. Since f_9 appeared first among them from the consideration of the principal components (for brevity, the details are not shown), f_9 would have been retained. This highlights the consonance between

the eigenvalue and the correlation matrix analysis. It can also be seen that no further reduction is possible, using correlation analysis, amongst objectives - f_5 , f_9 and f_{10} . Figure 7 shows the population on a f_9 - f_{10} plot. Once again, the population members seem to lie on the objective axes.

Further, the third iteration was executed with the three objectives - f_5 , f_9 and f_{10} , and eigenvalue analysis performed on final population, suggested all three to be non-redundant. However, identical correlations of f_5 and f_9 , with f_{10} , as shown below, suggest possible elimination of any one of these. As f_9 was chosen from the consideration of PCA1-the first principal component (for brevity, the details are not shown), it deems fit to be retained, while f_5 , to be eliminated. Now, all population members are placed on the true Pareto-optimal front, as shown in Figure 8.

	$\cdot f_5$	$\cdot f_9$	$\cdot f_{10}$
$\cdot f_5$	<u>1.000</u>	<u>1.000</u>	-0.916
$\cdot f_9$	<u>1.000</u>	<u>1.000</u>	-0.916
$\cdot f_{10}$	<u>-0.916</u>	<u>-0.916</u>	1.000

Table 2: Four iterations are needed to find the appropriate interaction among objectives on DTLZ5(2,10).

10-obj	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}
Iteration 1	f_1	f_2			f_5				f_9	f_{10}
Iteration 2					f_5				f_9	f_{10}
Iteration 3									f_9	f_{10}
Iteration 4									f_9	f_{10}

Table 2 summarizes the proceeding of the proposed procedure on DTLZ5(2,10). The fourth iteration confirms that no further reduction in the number of objectives is possible for this problem. In DTLZ5(2,M) problems, the Pareto-optimal front can be formed with M -th and any one of the other $(M - 1)$ objectives. For the Pareto-optimal solutions, $\theta_i = \pi/4$ for all $i = 2, \dots, 9$ and $x_i = 0.5$ for $i = 10, \dots, 19$. Thus, objective values for the Pareto-optimal solutions are single-valued: $f_{10} = \sin(\theta_1)$ and $f_j = \cos(\theta_1)/(\sqrt{2})^{10-j}$ for all $j = 1, \dots, 9$. The Pareto-optimal front can be formed by using f_{10} and any other objective, as follows:

$$f_{10}^2 + 2^{10-j} f_j^2 = 1. \quad (4)$$

For the pair f_9 and f_{10} used to form the Pareto-optimal front, the corresponding relationship is given as follows: $f_{10}^2 + 2f_9^2 = 1$. Figure 8 finds this relationship through a set of points obtained by the PCA-NSGA-II procedure.

The proposed PCA-NSGA-II procedure is able to maintain f_{10} objectives in all iterations and at the end discovers that objectives f_9 and f_{10} can form the correct Pareto-optimal frontier (Figure 8), thereby solving an otherwise difficult problem having 10 objectives. Recall that this problem was difficult to solve using a straightforward NSGA-II with even five million function evaluations (refer to Figure 2). This systematic study shows how the proposed PCA-NSGA-II procedure can be used to solve redundant, large-dimensional problems to Pareto-optimality.

6.2 Problem DTLZ5(3,10)

Next, we attempt to solve DTLZ5(3,10), which has a three-dimensional Pareto-optimal front. Table 3 shows that three iterations are needed to establish the appropriate interaction of the objectives of the Pareto-optimal front. The first iteration suggests five (including the 10-th objective) of the original 10 objectives are important and the remaining five objectives can be declared as redundant.

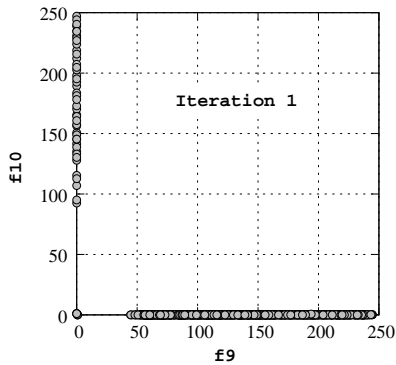


Figure 6: Population obtained after the first iteration of PCA-NSGA-II on DTLZ5(2,10).

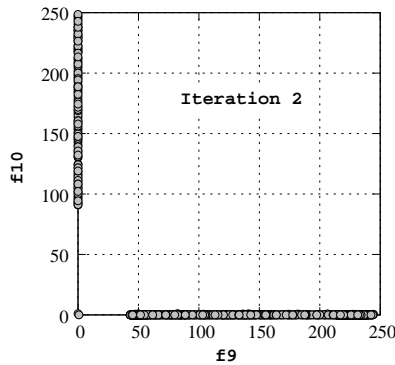


Figure 7: Population obtained after the second iteration of PCA-NSGA-II on DTLZ5(2,10).

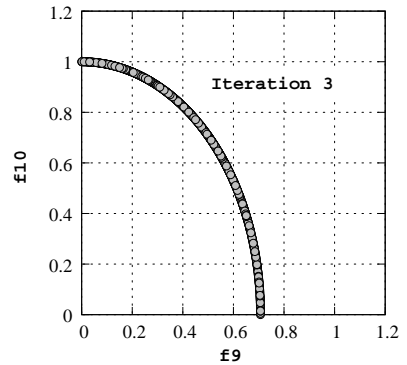


Figure 8: Population obtained after the third iteration of PCA-NSGA-II on DTLZ5(2,10).

Table 3: Three iterations are needed to find the appropriate interaction among objectives on DTLZ5(3,10).

10-obj	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}
Iteration 1	f_1						f_7	f_8	f_9	f_{10}
Iteration 2								f_8	f_9	f_{10}
Iteration 3								f_8	f_9	f_{10}

For brevity, we do not show the correlation matrices here. By using the principal component analysis, the second iteration reduces the number of objectives by two and the third iteration confirms that all three objectives (including the 10-th objective) are needed to keep an adequate level of interactions among the objectives. Figures 9, 10 and 11 show the populations obtained at the end of each iteration on f_8 - f_9 - f_{10} space. Once again, the population at the end of first iteration are mostly lined up along the objective axes with some points outside the axes lines. However, the population at the end of second iteration obtained using a five-objective evolutionary optimization puts all solutions on the axes, thereby allowing a PCA to find the correct interactions of the three objectives. Figure 11 shows that the population members lie well-distributed on the true three-dimensional Pareto-optimal front (shown with a patched surface). An analysis of the DTLZ5 problem formulation will reveal that if f_8 , f_9 and f_{10} are used to construct the three-dimensional Pareto-optimal front, the relationship is as follows: $f_{10}^2 + f_9^2 + 2f_8^2 = 1$. Figure 10 finds this relationship among these three objectives.

This problem was also difficult in solving using the original NSGA-II with five million function evaluations, as was shown in Figure 5. The PCA-NSGA-II took an overall $3 \times (800 \times 1000)$ or 2,400,000 evaluations. By choosing a smaller population size and a smaller number of generations for termination in the case of fewer objectives, the overall function evaluations can be reduced further.

6.3 Other DTLZ5(2,M) Problems

Now, we show the simulation results of the proposed method to a number of DTLZ5(I,M) problems. In all cases, identical parameter setting to that used in the previous subsections are used. Tables 4 and 5 show the proceedings of the proposed procedure on DTLZ5(2,5) and DTLZ5(3,5) problems.

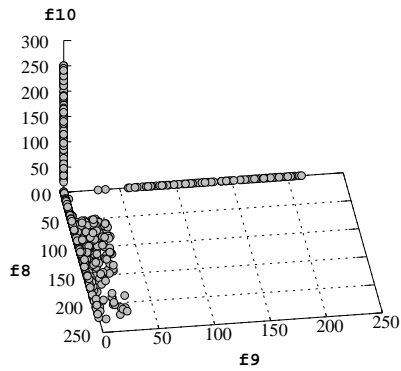


Figure 9: Population obtained after the first iteration of PCA-NSGA-II on DTLZ5(3,10).

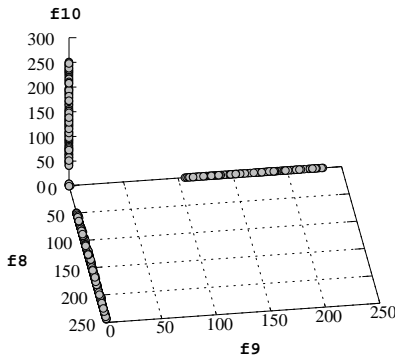


Figure 10: Population obtained after the second iteration of PCA-NSGA-II on DTLZ5(3,10).

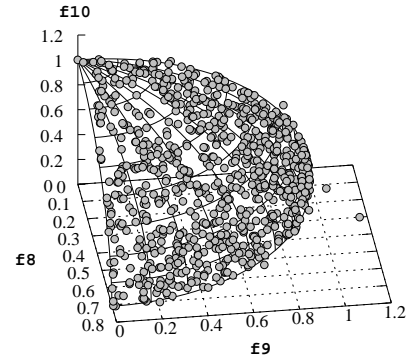


Figure 11: Population obtained after the third iteration of PCA-NSGA-II on DTLZ5(3,10).

In both cases, an appropriate final interaction among objectives is determined by the proposed approach. To investigate the performance of the proposed procedure on larger number of objectives, we apply it to 20 and 30-objective problems. Tables 6 and 7 show the proceedings of the proposed procedure.

Table 4: PCA-NSGA-II iterations on DTLZ5(2,5).

5-obj	f_1	f_2	f_3	f_4	f_5
Iteration 1				f_4	f_5
Iteration 2				f_4	f_5

Table 5: PCA-NSGA-II iterations on DTLZ5(3,5).

5-obj	f_1	f_2	f_3	f_4	f_5
Iteration 1			f_3	f_4	f_5
Iteration 2			f_3	f_4	f_5

For DTLZ5(2,20), although two objectives (f_{20} and any other) are enough to form the correct Pareto-optimal front, three objectives f_{17} , f_{19} and f_{20} are found to be important using our proposed procedure. Figure 12 shows that the presence of objective f_{17} makes some additional points to appear in the final non-dominated front. However, the points on the true Pareto-optimal front satisfying $f_{20}^2 + 2f_{19}^2 = 1$ is also found by the proposed procedure. On the other hand, the 30-objective DTLZ5(2,30) problem is correctly solved by finding objectives f_{30} and f_{29} at the end of the fifth iteration, as shown in Table 7. It is important to highlight that, given M objectives, if all combinations of objectives (with varying numbers and pairs) are to be tested to find the correct Pareto-optimal front, as many as $\sum_{i=0}^M \binom{M}{i}$ or equivalently 2^M combinations would need to be checked. Here, in general, $\binom{M}{i}$ represents a case, of independent optimization of all possible pairs involving i -objectives. It means that about a million (2^{20}) and about one billion (2^{30}) combinations would need to be checked for 20 and 30-objective DTLZ5 problems, respectively. The proposed procedure uses a fraction of these computations to make a good estimate of the Pareto-optimal front.

6.4 Problems DTLZ5(5,10)

We now consider two problems having a five-dimensional Pareto-optimal front constructed using 10 and 20 objectives, respectively. We use identical NSGA-II parameter values, as that used earlier. Table 8 shows the proceedings of the proposed procedure. At the end of the first iteration, four of the 10 objectives are found to be redundant. Thereafter, in the second iteration one more objective

Table 6: PCA-NSGA-II iterations on DTLZ5(2,20).

Iteration 1	f_1	f_7	f_{15}	f_{16}	f_{17}	f_{18}	f_{19}	f_{20}
Iteration 2	f_1	f_7			f_{17}	f_{18}	f_{19}	f_{20}
Iteration 3	f_1				f_{17}		f_{19}	f_{20}
Iteration 4					f_{17}		f_{19}	f_{20}
Iteration 5					f_{17}		f_{19}	f_{20}

Table 7: PCA-NSGA-II iterations on DTLZ5(2,30).

Iteration 1	f_{24}	f_{25}	f_{27}	f_{28}	f_{29}	f_{30}
Iteration 2		f_{25}	f_{27}		f_{29}	f_{30}
Iteration 3			f_{27}		f_{29}	f_{30}
Iteration 4					f_{29}	f_{30}
Iteration 5					f_{29}	f_{30}

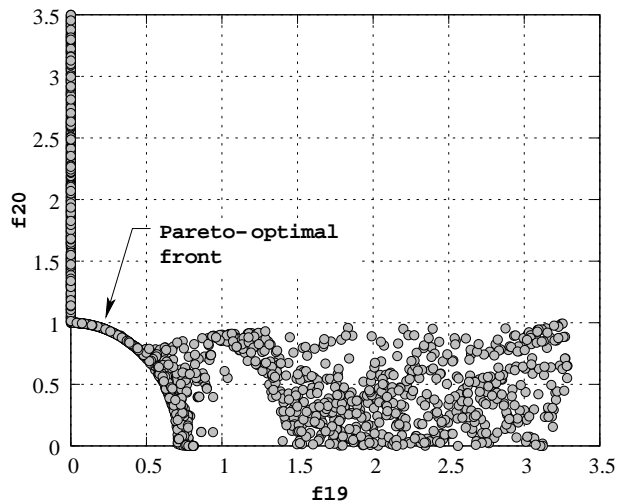


Figure 12: Population obtained after the fifth iteration of PCA-NSGA-II on DTLZ5(2,20).

is declared as redundant, thereby leaving five of the 10 objectives adequate for finding the Pareto-optimal front. It is also interesting to note that a hierarchy of important objectives are maintained (from f_{10} to f_7). However, instead of finding f_6 as the fifth objective, the procedure discovered f_1 as an important one. When the NSGA-II is executed with these five objectives, it could not quite converge on to the true Pareto-optimal front. The reason is given below. Equation 1 reveals that objectives f_7 till f_{10} involve variables x_1, x_2, x_3, x_4 , and x_{10} - x_{19} , whereas objective f_1 involves all 19 variables. Thus, the variables x_5 to x_9 gets decided only by minimization of f_1 alone. To make a small value of this objective, the θ_i parameter for all these variables need to be set close to $\pi/2$, so that $\cos(\theta_i) \approx 0$. This is possible in a situation where the five variables - x_5 to x_9 have a value of one and the function g takes on a large value (implying variables - x_{10} to x_{19} being close to either zero or one). As the goal of minimization of f_1 , turns in conflict with the condition necessary for the true Pareto-optimal front, that is, $g = 0$, the true front cannot be achieved with f_1 , present as an objective. Table 9 shows that for 20-objective DTLZ5 problem having a five-dimensional Pareto-optimal front, the proposed method discovers five objectives to be adequate to represent

Table 8: PCA-NSGA-II iterations on DTLZ5(5,10).

Iteration 1	f_1	f_6	f_7	f_8	f_9	f_{10}
Iteration 2	f_1		f_7	f_8	f_9	f_{10}
Iteration 3	f_1		f_7	f_8	f_9	f_{10}

Table 9: PCA-NSGA-II iterations on DTLZ5(5,20).

Iteration 1	f_1	f_{17}	f_{18}	f_{19}	f_{20}
Iteration 2	f_1	f_{17}	f_{18}	f_{19}	f_{20}

the desired front. But instead of finding f_{16} , it finds f_1 as an important objective along with four correct objectives (f_{17} to f_{20}), like in the case of DTLZ5(5,10).

The proposed method is not free from such errors. Determination of redundant objectives from the original 10 objectives through a PCA to a finite set of points iteratively seems to have a limitation. Finding the correct combination of five objectives from 10 objectives is a difficult task, as only one of $\binom{10}{5}$ or 252 combinations will produce the correct Pareto-optimal front. Our proposed methodology is able to find four of the correct five objectives. It is important to note that no information about the exact number of correct objectives required to solve the problem efficiently was used in the optimization process. Thus, if all combination of objectives are to be considered, the task for this problem is to identify one of the 2^{10} or 1,024 different combinations to find the right objective combination to solve the problem efficiently and accurately. Such an exhaustive search procedure will be an expensive proposition, particularly for a large number of objectives.

6.5 DTLZ2 Problems

Unlike a reduced dimensionality of the Pareto-optimal frontier in test problems DTLZ5, M -objective DTLZ2(M) problems preserve the dimensionality of the Pareto-optimal front. It implies, that in DTLZ2(M) problems, the entire Pareto-optimal front must involve all M objectives. In fact, DTLZ2(M) is identical to DTLZ5(M,M). The value of k is chosen to be 10 here, so that total number of variables are $n = M + 9$. The Pareto-optimal front is made up of $f_i \in [0, 1]$ for all $i = 1, 2, \dots, M$, such that $\sum_{i=1}^M (f_i^*)^2 = 1$. We apply PCA-NSGA-II to solve DTLZ2(3) and DTLZ2(5) problems.

Table 10: PCA-NSGA-II on DTLZ2(3).

3-obj	f_1	f_2	f_3
Iteration 1	f_1	f_2	f_3

Table 11: PCA-NSGA-II on DTLZ2(5).

5-obj	f_1	f_2	f_3	f_4	f_5
Iteration 1	f_1	f_2	f_3	f_4	f_5

A population of size 800 is run for 1,000 generations and Tables 10 and 11 show the proceedings. It is interesting to note in both problems, the proposed methodology correctly discovers that there exists no redundant objective in these problems, right in the first iteration. Thus, these problems get solved trivially, as if the NSGA-II procedure was applied to the original M -objective problem to find the Pareto-optimal front. Since all M objectives are found to be important, there is no need for any further iteration.

7 Conclusions

We started the study by demonstrating that one of the evolutionary multi-objective optimization (EMO) methodology – the elitist non-dominated sorting GA or NSGA-II – is vulnerable to a large number of objectives. Although the utility of finding a representative set of solutions for the complete Pareto-optimal front in many objectives is questionable from a practical standpoint, the presence and importance of solving a large number of objectives is unquestionable. In

such large-dimensional problems, we did argue that there still lies merits in finding a representative set of solutions in the complete Pareto-optimal set, if such problems degenerate to possess a lower-dimensional Pareto-optimal frontier. In many problems, decision-makers are often unable or reluctant to reduce a set of objectives desired to be solved to a single objective. In many such cases, although the objective space is large-dimensional, the resulting Pareto-optimal front may be inherently low-dimensional.

This paper has addressed solving such problems and suggested a principal component analysis (PCA) based NSGA-II procedure which iteratively eliminates redundant objectives. The proposed procedure is able to identify the correct number of objectives and correct objective combinations necessary to find the true Pareto-optimal front, in most cases. When the dimension of the desired frontier is two or three, the proposed PCA-NSGA-II procedure could identify the Pareto-optimal front for objectives as many as 30. On the other hand, the proposed method has incorrectly identified one of the five desired objectives in a 10-objective problem having a five-dimensional Pareto-optimal front. The performance of the proposed methodology is seen, not to be much affected, by an increase in the number of the objectives. Hence, if the task involves finding a large-dimensional Pareto-optimal front, the proposed methodology begins to show some vulnerability. Having a statistical analysis procedure to determine the redundancy of objectives, will largely depend on the data on which such an analysis is performed. For large-dimensional problems, some such errors may happen. While the PCA filtering strategy may need refinement to make the proposed method more efficient, multiple runs for identification of redundant objectives would make the study, more reliable. Instead of using the quick-and-dirty crowding distance computation scheme used in NSGA-II, a better scheme involving a clustering approach can be used. This way, a better distribution of solutions is expected, which may in turn allow a better and more reliable PCA for identifying salient objectives. Other more direct statistical methods can be adopted to establish redundant objectives by manipulating the correlation matrix. Nevertheless, this paper amply demonstrates the importance of dimensionality reduction in solving some structured problems using an evolutionary multi-objective optimization technique. Hopefully, this study will encourage more such, for devising more reliable and efficient methods of dimensionality reduction and eventually facilitate solutions to large-dimensional multi-objective optimization problems.

Acknowledgements

Authors acknowledge the support provided by STMicroelectronics, Agrate and Noida for performing this study.

References

- [1] P. J. Agrell. On redundancy in MCDM. *European Journal of Operational Research*, 98(3):571–586, 1997.
- [2] J. Branke and K. Deb. Integrating user preferences into evolutionary multi-objective optimization. In Y. Jin, editor, *Knowledge Incorporation in Evolutionary Computation*, pages 461–477. Hiedelberg, Germany: Springer, 2004.
- [3] J. Branke, K. Deb, H. Dierolf, and M. Osswald. Finding knees in multi-objective optimization. In *Parallel Problem Solving from Nature (PPSN-VIII)*, pages 722–731. Heidelberg, Germany: Springer, 2004. LNCS 3242.
- [4] J. Branke, T. Kaußler, and H. Schmeck. Guidance in evolutionary multi-objective optimization. *Advances in Engineering Software*, 32:499–507, 2001.
- [5] C. A. C. Coello, D. A. VanVeldhuizen, and G. Lamont. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Boston, MA: Kluwer Academic Publishers, 2002.

- [6] K. Deb. *Multi-objective optimization using evolutionary algorithms*. Chichester, UK: Wiley, 2001.
- [7] K. Deb. Multi-objective evolutionary algorithms: Introducing bias among pareto-optimal solutions. In A. Ghosh and S. Tsutsui, editors, *Advances in Evolutionary Computing: Theory and Applications*, pages 263–292. London: Springer-Verlag, 2003.
- [8] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan. A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
- [9] K. Deb and H. Gupta. Searching for robust Pareto-optimal solutions in multi-objective optimization. In *Proceedings of the Third Evolutionary Multi-Criteria Optimization (EMO-05) Conference (Also Lecture Notes on Computer Science 3410)*, pages 150–164, 2005.
- [10] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable test problems for evolutionary multi-objective optimization. In A. Abraham, L. Jain, and R. Goldberg, editors, *Evolutionary Multiobjective Optimization*, pages 105–145. London: Springer-Verlag, 2005.
- [11] M. Farina and P. Amato. A fuzzy definition of “optimality” for many-criteria decision-making and optimization problems. *submitted to IEEE Trans. on Sys. Man and Cybern.*, 2002.
- [12] C. M. Fonseca and P. J. Fleming. Genetic algorithms for multiobjective optimization: Formulation, discussion, and generalization. In *Proceedings of the Fifth International Conference on Genetic Algorithms*, pages 416–423, 1993.
- [13] C. M. Fonseca and P. J. Fleming. Multiobjective optimization and multiple constraint handling with evolutionary algorithms—Part II: Application example. *IEEE Transactions on Systems, Man, and Cybernetics: Part A: Systems and Humans*, 28(1):38–47, 1998.
- [14] T. Gal and H. Leberling. Redundant objective functions in linear vector maximum problems and their determination. *European Journal of Operational Research*, 1:176–184, 1977.
- [15] T. Gal and T. Hanne. Consequences of dropping nonessential objectives for the application of mcdm methods. *European Journal of Operational Research*, 119:373–378, 1999.
- [16] T. Goel, R. Vaidyanathan, R. T. Haftka, N. Queipo, W. Shyy, and K. Tucker. Response surface approximation of pareto-optimal front in multi-objective optimization. Department of Mechanical and Aerospace Engineering, University of Florida, USA, 2004.
- [17] Mikkel T. Jensen. Guiding single-objective optimization using multi-objective methods. In *EvoWorkshops*, pages 268–279, 2003.
- [18] V. Khare, X. Yao, and K. Deb. Performance scaling of multi-objective evolutionary algorithms. In *Proceedings of the Second Evolutionary Multi-Criterion Optimization (EMO-03) Conference (LNCS 2632)*, pages 376–390, 2003.
- [19] A. V. Lotov, V. A. Bushenkov, and G. K. Kamenev. *Interactive decision maps*. Boston, MA: Kluwer, 2004.
- [20] K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer, Boston, 1999.
- [21] E. Sandgren, E. Jensen, and J. Welton. Topological design of structural components using genetic optimization methods. In *Proceedings of the Winter Annual Meeting of the American Society of Mechanical Engineers*, pages 31–43, 1990.
- [22] P. Shukla and K. Deb. Comparing classical generating methods with an evolutionary multi-objective optimization method. In *Proceedings of the Third International Conference on Evolutionary Multi-Criterion Optimization (EMO-2005)*, pages 311–325, 2005. Lecture Notes on Computer Science 3410.
- [23] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(22):2319–2323, 2000.