International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 6 (2018) pp. 3789-3809

K-means cluster algorithm-based evolutionary approach for constrained multi-objective optimization

A. A. Mousa[1,2], M. Higazy[1,3], Yousria Abo-Elnaga[4]

[1] *Department of Mathematics and Statistics, Faculty of Sciences, Taif University, Saudi Arabia.*
[2]*Department of Basic Engineering Science, Faculty of Engineering, Shebin El-Kom, Menoufia University, Egypt.*
3Department of physics and engineering mathematics, faculty of electronic engineering, Menofia University, Egypt.
[4]*Department of Basic science, Higher Technological Institute, Tenth of Ramadan City, Egypt.*

Abstract

One of the most popular approaches for generating non-dominated solutions of a multi-objective optimization problem is the evolutionary algorithms. The need of an approximation to the non-dominated set for the decision maker for selecting a final preferred solution is required in several real-life applications . Unfortunately, the cardinality of the Pareto optimal solutions' set may be very large or even infinite. On the other hand, due to the overflow of information, the decision maker (DM) may not be concerned in having an excessively large number of Pareto optimal solutions to deal with. In this paper, a new enhanced evolutionary algorithm is presented, our proposed algorithm has been enriched with modified k-means cluster scheme, On the first hand, in phase I, K-means clustering method is implemented to partition the population into a determined number of subpopulation with-dynamic-size, where distinct genetic algorithms (GAs) operators can be applied to each sub-population, instead of one GAs operator applied to the whole population. On the other hand, phase II uses K-means algorithm in order to make the algorithms practical by allowing a decision maker to control the resolution of the Pareto-set approximation by choosing an appropriate k value (no of required clusters). To prove the excellence of the proposed approach compared to state-of-the-art evolutionary algorithms, diverse numerical studies will be done using a suite of multimodal test functions taken from the literature.

Keywords: K-means cluster algorithm, evolutionary approaches, constrained optimization, multiobjective optimization

INTRODUCTION

Multiobjective Optimization Problems (MOPs) are those having more than one objective. There are a set of possible solutions and there is no unique optimum solution. these solutions are optimal in the wider sense that no other solutions in the search space are dominate them when all objectives are considered. As in [1,2], these solutions are called *Pareto optimal solutions*. Surely, MOPs appear in several areas of knowledge for example economics [3-6], machine learning [6-8] and electrical power system [9-12]. A few developed evolutionary algorithms are constructed for solving the constrained multi-objective optimization problems. In spite of the several studies for solving the constrained optimization problems; there are no enough studies concerning the procedure for handling constraints. As an instance, in [13], treating constraints as high-priority objectives was suggested by Fonseca. In [14] by Harada, a few effective constraint-handling guidelines were suggested and a Pareto descent repair method was constructed. For MOPs, because the objective vector cannot be directly assigned as the fitness function value, it is always needed a correctly constructed algorithm for fitness assignment. Using the Pareto dominance relationship, the values of the fitness functions are assigned by most of the existing MOEAs [15-16].

AL Malki et al.[17] have presented Hybrid Genetic K-Means algorithm for Identifying the Most Significant Solutions from Pareto Front. They have implemented clustering techniques to organize and classify the solutions for various problems. In [18], the Hybrid Genetic Algorithm with K-Means has been presented by Al Malki et al. to solve the Clustering Problems. They have succeeded to eliminate the empty cluster problem by using a hybrid form of the k-means algorithm and GAs. Their proposition was be proved by the results of simulation tests via various data collections. The use of K-means clustering technique in [19] enable the algorithm to implement various operators of GA to each subpopulation rather than utilizing a single GA operator for all population.

In this article, a new optimization algorithm is proposed which runs in two stages: in the first stage, the algorithm combines all principal characteristics of K-means with GA clustering method and uses it as a search engine to produce the correct Pareto optimal front. The use of K-means clustering technique enable the algorithm to implement various operators of GA to each subpopulation rather than utilizing a single GA operator for all population.

Then in the second phase, k -means cluster algorithm is adopted as reduction algorithm in order to improve the spread of the solutions found so far. Our proposed algorithm has been enriched with modified k-means cluster scheme, the use of k -means also makes the algorithms practical by allowing a decision maker to control the resolution of the Pareto-set approximation by choosing an appropriate k value. k -means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean. Finally, various kinds of multi-objective benchmark problems have been reported to stress the importance of hybridization algorithms in generating Pareto optimal sets. Simulation results with the proposed approach will be compared to those reported in the literature. The comparison will demonstrate the superiority of the proposed approach and confirms its potential to solve the multi-objective optimization problems.

The organization of this paper is as follows: Multi-objective optimization problem is presented in Section 2. Overview of GAs are presented in Section 3. Clustering Algorithms are introduced in Section 4. K-Means for clustering problems is introduced in section 5. The proposed algorithm is presented in section 6. The Simulation results are discussed in Section 7. Finally, we conclude the paper in Section8.

MULTIOBJECTIVE OPTIMIZATION (MO)

Mathematically, a general minimization problem of M objectives can be presented as [20-21]:

$$\left.\begin{array}{ll}\text{Minimize:} & \vec{f}(\vec{x}) = \left[f_i(\vec{x}),\ i = 1, 2, ..., M \right] \\ \text{subject to the constraints:} & g_j(\vec{x}) \le 0,\ j = 1, 2, ..., J.\end{array}\right\} \quad (1)$$

and $\vec{x} = [x_1, x_2, ..., x_n]$, where the dimension of the decision variable space is equal n, the i-th objective function is $f_i(\vec{x})$ and the j-th inequality constraint is $g_j(\vec{x})$. Then the main task of the MO problem is to find $\vec{x}$ that optimize $f_i(\vec{x})$. The evaluation of the solutions uses the concept of the Pareto dominance because the notion of an optimum solution in MO is different compared to the single objective optimization (SO).

Definition 1: (Pareto dominance). A vector $\vec{u} = (u_1, u_2, ..., u_M)$ is said to dominate a vector $\vec{v} = (v_1, v_2, ..., v_M)$ ($\vec{u}$ dominate $\vec{v}$ denoted by $\vec{u} \succ \vec{v}$), for a MO minimization problem, if and only if

$$\forall i \in \{i, ..., M\},\ u_i \le v_i \wedge \exists i \in \{i, ..., M\} : u_i < v_i \quad (2)$$

where M is the dimension of the objective space.

Definition 2: (Pareto optimality). A solution $\vec{u} \in U$, is called a Pareto optimal iff there is no other solution ($\vec{v} \in U$), such that $\vec{u}$ is dominated by $\vec{v}$. These solutions are called non dominated solutions. The set of all such non dominated solutions constitutes the Pareto-Optimal Set.

OVERVIEW OF THE GA

The techniques of natural selection are the primary concept of GAs. Any optimization variable, (xn), is converted into a gene as a real number or a string of bits. All variables' genes, $x_1, \ldots, x_n$, form a chromosome, that depicts all individuals. Depending on the specific problem, an array of real numbers, binary string, a list of components in a database could be considered as a chromosome. Each possible solution is represented by an individual, and the set of individuals form a population. From a population the fittest is chosen for matting. Matting is done by merging genes from distinct parents to generate a child, called a crossover. Finally, the children are added to the population and the process repeats over again, thus symbolizing an artificial Darwinian environment as illustrated in Figure 1. The optimization shall stop if and only if the population has converged or the generation has been reached the maximum number.

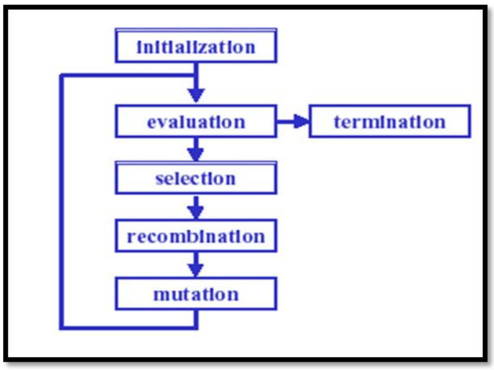

Figure 1. GAs outline for optimization problems.

CLUSTERING ALGORITHM

Clustering is an approach of partitioning of data into similar objects sets. Each set, called cluster, consists of mutually similar objects and different to objects of other clusters [22]. Clustering is a very important area of research, which has various applications in several fields such as in psychiatry [23], market research [24], archaeology [25], pattern recognition [26], medicine [27] and engineering [28].

In the literature, clustering has various proposed algorithms. Due to its simplicity and accuracy, the K-means clustering is possibly the most commonly-used clustering algorithm [29].

K-Means Clustering Technique

In 1967, Macqueen produced the K-means clustering algorithm [30]. Due to the easiness of K-means clustering algorithm, it is used in many fields. The K-means clustering algorithm separates data into k sets as it is a partitioning clustering approach [31]. The K-means clustering algorithm is more prominent because it is an intelligent algorithm that can cluster massive data rapidly and efficiently.

The basic idea of K-means algorithm is to classify the data D into k different clusters where D is the data set, k is the number of desired clusters. More precisely, the following are the main steps of the K-means clustering algorithm (Figure 2):

Step 1: Define a number of desired clusters, k.

Step 2: Choose the initial cluster centroids randomly, which represent temporary means of the clusters.

Step 3: Compute the Euclidean distance from each object to each cluster and each object is assigned to the closest cluster with the smallest square distance.

Step 4: For each cluster, the new centroid is computed, and each centroid value is now replaced by the respective cluster centroid

Step 5: Repeat steps 3 and 4 until no point changes its cluster.

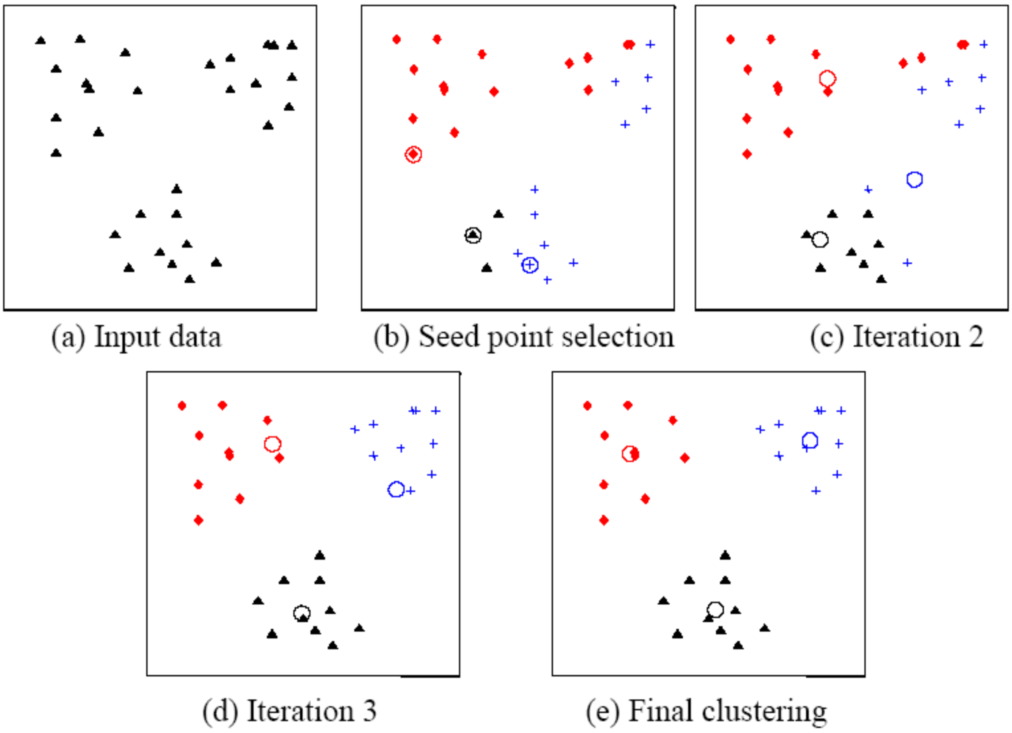

Figure (2): K-means algorithm procedure(Taken from [32]).

THE PROPOSED APPROACH

In this section, the proposed algorithm that we are dealing with is informally described. Our proposed algorithm consists of two phases. In phase I, K-means clustering technique is implemented to partition the population into a determined number of subpopulation with-dynamic-sizes. On the other hand, phase II applies K-means algorithm to make the algorithms practical by allowing the decision maker to control the precision of the Pareto-set approximation by choosing a suitable number of the needed clusters.

Phase I

Step1: Population Initialization: Two disconnected sub-populations are used by the algorithm, the individuals that initialized randomly satisfying the search space (The lower and upper bounds) form the first one, while the reference points that satisfying all constraints (feasible points) form the second one. However, we have concentrated on how elitism could be introduced to guarantee convergence to the true Pareto-optimal solutions. For that, an "archiving/selection" plan is proposed that ensures instantaneously the advance to reach the Pareto-optimal set and coverage of the entire range of the non-dominated solutions. An externally limited size archive $\boldsymbol{A}^{(t)}$ of non-dominated solutions is preserved and iteratively updated in the presence of new solutions based on the concept of ε-dominance by the proposed algorithm.

Step 2: Repair Algorithm: By repairing infeasible individuals, the main task of this algorithm is to recognize all feasible individuals from the infeasible ones. Via this algorithm, the infeasible individuals in a certain population are developed gradually till they become feasible ones. As explained in [10], the repair operation is done as follows. Let $\boldsymbol{S}$ be the feasible domain and $\omega \notin \boldsymbol{S}$ be a search point (individual). Since better reference point has better chances to be selected, the algorithm selects one of the reference points, let it be $\boldsymbol{r} \in \boldsymbol{S}$, and generates random points $\bar{Z} = \alpha\omega + (1-\alpha)r, \quad \alpha \in [0,1]$ from the segment defined between $\omega, \boldsymbol{r}$, but the segment may be extended equally [33, 34] on both sides determined by a user specified parameter $\mu \in [0,1]$.

In such a case the algorithm selects one of the reference points (Better reference point has better chances to be selected), say $\boldsymbol{r} \in \boldsymbol{S}$ and creates random points $\bar{Z} = \alpha\omega + (1-\alpha)r, \quad \alpha \in [0,1]$ from the segment defined between $\omega, \boldsymbol{r}$, but the segment may be extended equally [33, 34] on both sides determined by a user specified parameter $\mu \in [0,1]$. So, an up-to-date feasible individual is represented as:

$$\left.\begin{aligned} z_1 &= \gamma.\omega + (1-\gamma).r \\ z_2 &= (1-\gamma).\omega + \gamma.r \end{aligned}\right\}, \gamma = (1+2\mu)\delta - \mu, \delta \in [0,1] \quad (3)$$

Step3: Classifying a population according to non-domination: In this step, the objective functions of each solution are evaluated and the non-dominated sets of solutions are selected according to non-domination concept by using the following algorithm [35-36]. Figure(3) illustrate the classifying a population according to non-domination.

- Start with m = 1.
- The solutions x_m and x_n are compared for domination for all $n = 1, 2, \cdots, N_{POP}$ and $m \neq n$.
- If x_m is dominated by x_n , for any n, mark x_m as 'dominated', and it is Inefficient.
- Increase m by one and return to Step 2. If all solutions in the population are considered, go to Step 5
- The solutions which are not marked 'dominated' are non-dominated solutions
- Save the non-dominated solutions in an external archive.

Figure (3): Classifying a population according to non-domination

Step 4: Selection Stage**:** Through a weighted combination, a dynamic-weight approach [37] is used to aggregate, the multi-objective functions into a single combined fitness function. The main characteristic of dynamic-weight approach is that the weights attached to the multiple objective functions are not fixed but randomly specified. Consequently, the direction of the search is not specific. The multiple objective functions are combined into a scalar fitness solution as follows:

$$Z = w_1 \cdot f_1(x) + \ldots + w_i \cdot f_i(x) + \ldots + w_q \cdot f_q(x); \quad (4)$$

where x is an individual, Z is the combined fitness function, $f_i(x)$ is the i[th] objective function and w is a weighting-vector with $w_i \geq 0 \ \forall \ i = 1, \ldots, q;$ where $\sum_{i=1}^{q} w_i = 1$. In general, the value of each weight can be randomly determined as follows:

$$w_i = \frac{random_i}{\sum_{j=1}^{q} random_j}, \qquad i = 1, 2, \ldots, q; \qquad (5)$$

where $random_i$ and $random_j$ are non-negative random real numbers.

After formulating the combined fitness function (Z) for each chromosome by equation (4), the selection probability for each chromosome is then defined by following linear scaling function:

$$prob_i = \frac{Z_i - Z_{\min}}{\sum_{j=1}^{N_{pop}} (Z_j - Z_{\min})} \qquad (6)$$

where $prob_i$ is the selection probability of a chromosome i whose combined fitness function is Z_i and $Z_{\min}$ is the worst combined fitness in the population.

Finally, a pair of individual is chosen randomly from the current population and the best solution with better selection probability is chosen and subsequently copied in mating pool [30]. This step has to be repeated frequently, until the size of the mating pool is equivalent to the original population size.

Step 5: K-Means clustering technique: In this paper, K-means clustering technique[38] was used to split the population to K separated subpopulations with dynamic size, as shown in Figure (4). By using this method, various operators of GA can be applied to subpopulations in stead of applying a single GA operator to the whole population.

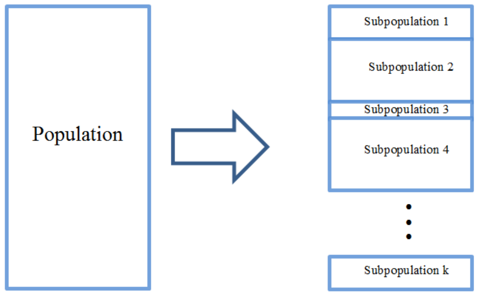

Figure 4: The splitting of the population into K separated subpopulations

Step6: Crossover operator: The crossover operator generates a new offspring by mating (combing) two chromosomes (parents). The concept behind crossover is that the new offspring may take the best characteristics from the parents, and it will be better than both parents. The Crossover occurs during evolution process according to user-definable fixed probability. In this step, three different crossover operators are used which are horizontal band crossover, uniform crossover and real part crossover which are described as follows:.

- **Horizontal band crossover :**In this operator, two horizontal crossover sites are selected randomly. The information in the horizontal region, which is determined by horizontal crossover sites, is exchanged between the two parents based on a fixed probability [39].
- **Uniform crossover:** In uniform crossover [40], random (0,1) mask is generated. In the random mask the '0' denotes bit unchanged, while '1' represents bits swapping.
- **Real part crossover:** This operator works on the real part (p_{it}) of a chromosome to cross the real variables p_{it} of the chromosomes by exchanging the information in column vectors of p_{it} matrix [41]. The new power matrices of off-springs ($p_{it}(offspring_1)$ and $p_{it}(offspring\,2)$) are created from The power matrices (p_{it}) of parent chromosomes, as follows:

$$p_{it}(offspring_1) = \left[column_1^{parent_1}, ..., (1-\alpha)column_j^{parent_1} + \alpha column_j^{parent_2}, ..column_T^{parent_1}\right]$$
$$p_{it}(offspring\,2) = \left[column_1^{parent_2}, ..., (1-\alpha)column_j^{parent_2} + \alpha column_j^{parent_1}, ..column_T^{parent_2}\right]; \quad (7)$$

where α is chosen randomly between 0 and 1, and $coulmn_j$ is the column vector in power matrix and "j" is a random positive integer in range of [1,T].

Step 7: Mutation operator: This operator is used to explore some of the points in the search space by altering few genes value in an individual randomly from its initial state. The mutation is generally occurs according to user-definable mutation probability which is usually low value. In this step, we used three different mutation methods which are one point mutation, intelligent mutation and swap window mutation which are described as follows:

- **Single point mutation:** In single point mutation, a single bit is chosen randomly from binary part (u_{it}) of offspring and then its value is changed from '0' to '1' and vice versa [40-41].
- **Intelligent mutation:** This mutation [42] searches for (10) or (01) combinations in commitment schedule, then randomly changing them to 00 or 11.
- **Swap window mutation:** Swap window mutation works on the binary part (u_{it}) of a chromosome by selecting: 1) two units at random 2) a time window of width w between 1 and T and 3) the location of the window, then exchanging the entries of the two units which are in the window [43].

At any time instant, if status of any unit changed due to mutation operator from 0 to 1, the corresponding output power of this unit will be changed from 0 to real value chosen at random from the range of $[p_i^{\min}, p_i^{\max}]$.

Step 8: Combination stage: At this step [44,45], all subpopulations are grouped once more to form a new population, as shown in Figure (5).

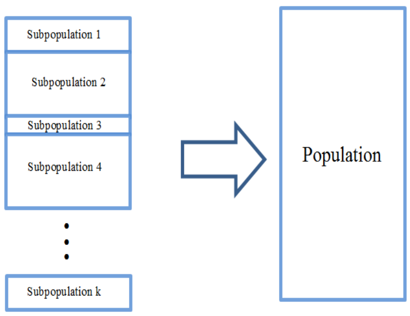

Figure 5: Commination stage

Step 9: Update the Archive of Non-dominated Solution**:** The proposed approach has an external archive of non-dominated solutions which gets updated iteratively when new solutions are found based on the concept of non-domination. The Archive is updated in each iteration by copying the solutions of current population P^t to archive V^t and applying the dominance criteria to remove all dominated solutions (i.e., each solution of P^t has three probabilities as in Algorithm 1 in Figure (6) [60].

Algorithm 2:Update the Archive

Input $(V^t, X \in P^t)$
If $\exists Y \in V^t \,|\, Y \succ X$ *then*
$V^t = V^t$
Else if $\exists Y \in V^t \wedge X \succ Y$ *then*
$V^t = V^t \cup \{X\} / \{Y\}$
Else if $\nexists Y \in V^t \,|\, Y \succ X$ *then*
$V^t = V^t \cup \{X\}$
End
Output $: V^t$

Figure 6: Update the Archive of non-dominated solutions

Phase II: K-means Algorithm

Step 10: Centres $Z_1, Z_2, ..., Z_K$ of K initial cluster are randomly chosen from the n observations $\{x_1, x_2, ..., x_n\}$.

Step 11: A point $x_1, i = 1, 2, ..., n$ is assigned to cluster $C_j, j \in \{1, 2, ..., k\}$ if and only if:

$$\|x_i - z_j\| < \|x_i - z_p\|, p = 1, 2, ..., K \;\; \& \; j \neq p \quad (8)$$

Step 12: Centres of new cluster $z_1, z_2, ..., z_K$ are computed as follows:

$$z_i^* = \frac{1}{n_i} \sum_{x_j \in C_i} x_j, \; i = 1, 2, ..., K; \quad (9)$$

where n_i is the number of elements which are belonging to cluster C_j.

Step 13: If $z_i^* = z_i, i = 1, 2, ..., K$ then the algorithm terminate, otherwise go to step 11.

After this phase, we get an initial center for all predetermined clusters.

Identifying the centroids of Pareto front

In order to identify one single point belong to the Pareto front that represents certain cluster, the following algorithm as shown in Figure (7) is presented. Figure (7) presents an algorithm for identifying the centroid for each cluster in such a way that it locate in the Pareto front while, Figure (8) shows geometrically how the identifying mechanism occurs.

INPUT $partition\_matrix\ M = \left[m_{Kj}\right],\ K = 1,...,K\ \&\ j = 1,...,n\ ,u_{Kj} = \begin{cases} 1 & if\ x_j \in C_K \\ 0 & if\ x_j \notin C_K \end{cases}$

INPUT $Z = \{z_1, z_2, ..., z_K\}$

For $all\ K \in \{1,...,K\} do$

Find $z_K^* = \{x_{Ki} \mid (x_{Ki} - z_k) = Min(x_{Ki} - z_K) \forall i = 1,2,...,n, u_{Kj} = 1\}$

End for

OUTPUT Pareo front centriod: $Z^* = (z_1^*, z_2^*, ..., z_K^*)$

Figure 7: Algorithm for identifying the Pareto front centroids, taken from [17]

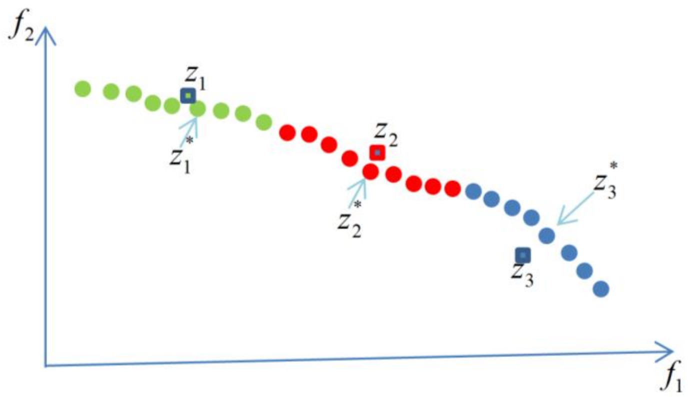

Figure 8: Algorithm for identifying the Pareto front centroids, taken from [17]

Results

The proposed algorithm is applied by various Pareto front of MOO Test Instances for the CEC09 [46]; which are containing different Pareto front characteristics to demonstrate its ability to select the most compromise set of solution. The problems cover different characteristics of MOOPs, namely convex Pareto front, nonconvex Pareto Front and discrete Pareto front and non-uniformity of solution distribution. Twenty problems from CEC2009 contain different shapes of Pareto front is selected and described in Table 1.

Table 1: MOO Test Instances for the CEC09

Problem	Problem features	Problem	Problem features
UF1	Unconstrained Problem: The two objectives to be minimized	CF1	Constrained Problem: Two objectives to be minimized
UF2		CF2	
UF3		CF3	
UF4		CF4	
UF5		CF5	
UF6		CF6	
UF7		CF7	
UF8	Unconstrained Problem: Three multiple objectives to be minimized	CF8	Constrained Problem: Three multiple objectives to be minimized
UF9		CF9	
UF10		CF10	

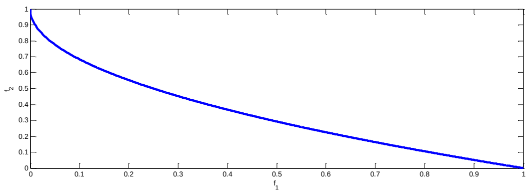

Figure 9: Pareto Front of problem UF2

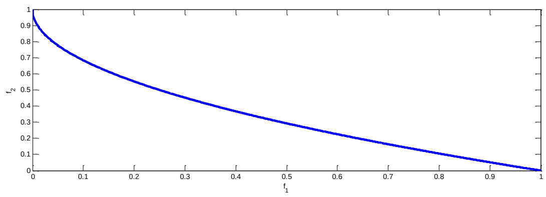

Figure 10: Pareto Front of problem UF1

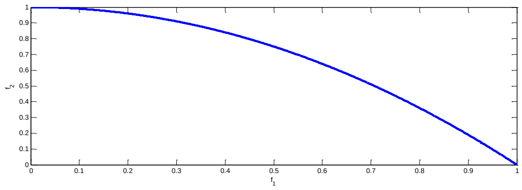

Figure 11: Pareto Front of problem UF4

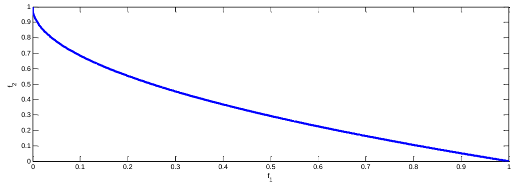

Figure 12: Pareto Front of problem UF3

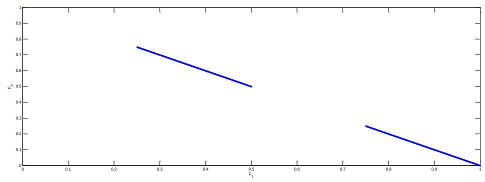

Figure 13: Pareto Front of problem UF6

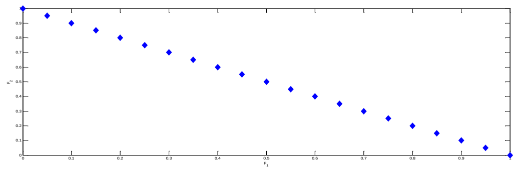

Figure 14: Pareto Front of problem UF5

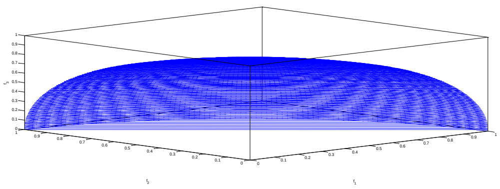

Figure 15: Pareto Front of problem UF8

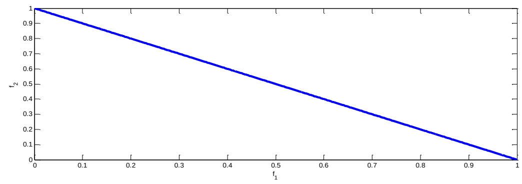

Figure 16: Pareto Front of problem UF7

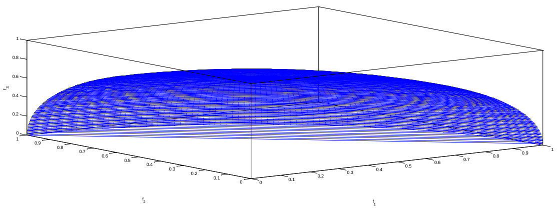

Figure 17: Pareto Front of problem UF10

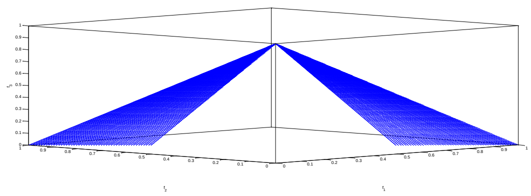

Figure 18: Pareto Front of problem UF9

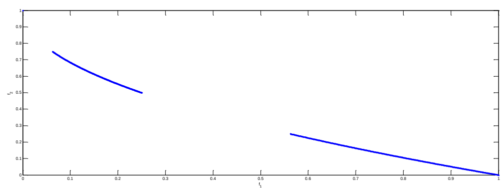

Figure 19: Pareto Front of problem CF2

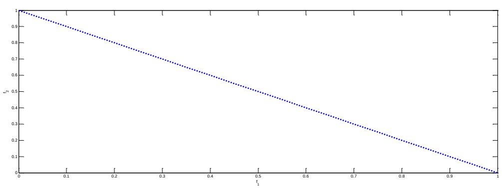

Figure 20: Pareto Front of problem CF1

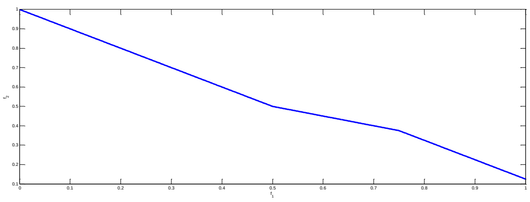

Figure 21: Pareto Front of problem CF4

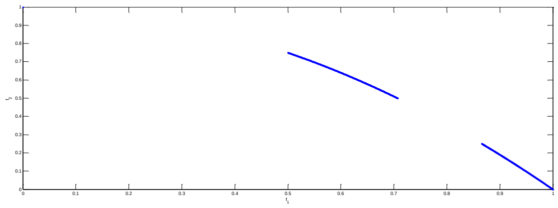

Figure 22: Pareto Front of problem CF3

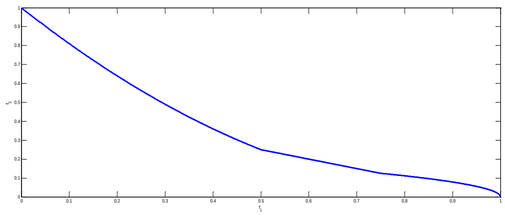

Figure 23: Pareto Front of problem CF6

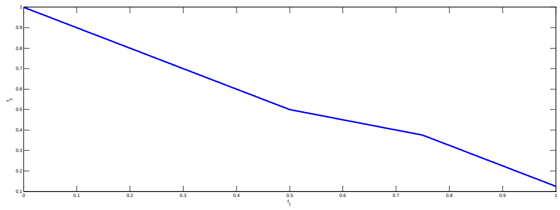

Figure 24: Pareto Front of problem CF5

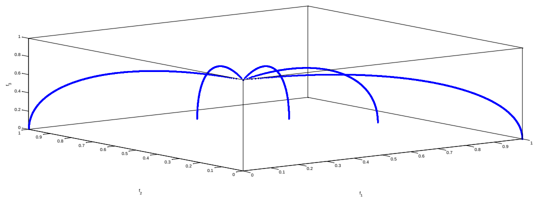

Figure 25: Pareto Front of problem CF8

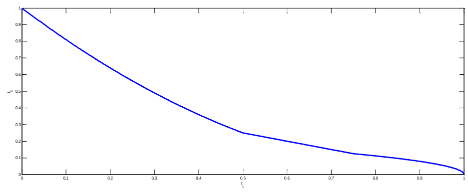

Figure 26: Pareto Front of problem CF7

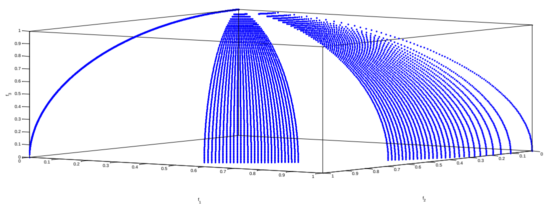

Figure 27: Pareto Front of problem CF10

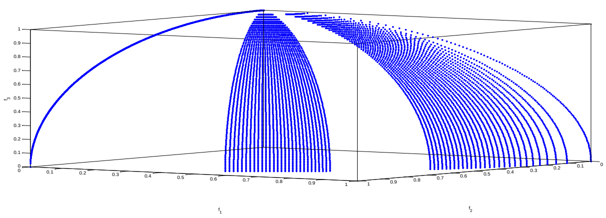

Figure 28: Pareto Front of problem CF9.

The algorithm (Phase I) was implemented for the 20 problems which are taken from CEC2009 with different Pareto front Charctertistics as in figures (9-28).

Optimization of the above-formulated objective functions using multi-objective EAs yields a set of Pareto-optimal solutions, not give a single optimal solution. But the DM in practical application needs to select a set of solutions with limited size. From the above results the selected solutions should have a diversity characteristics also it must cover the entire Pareto front domain and allows the DM to attain a reasonable representation of the Pareto-optimal front. In addition, the results guarantee that the diversity among selected solutions is achieved. Finally, the proposed algorithm has a diversity preserving mechanism to overcome the cruise of huge number of Pareto front as in figures(29-48)

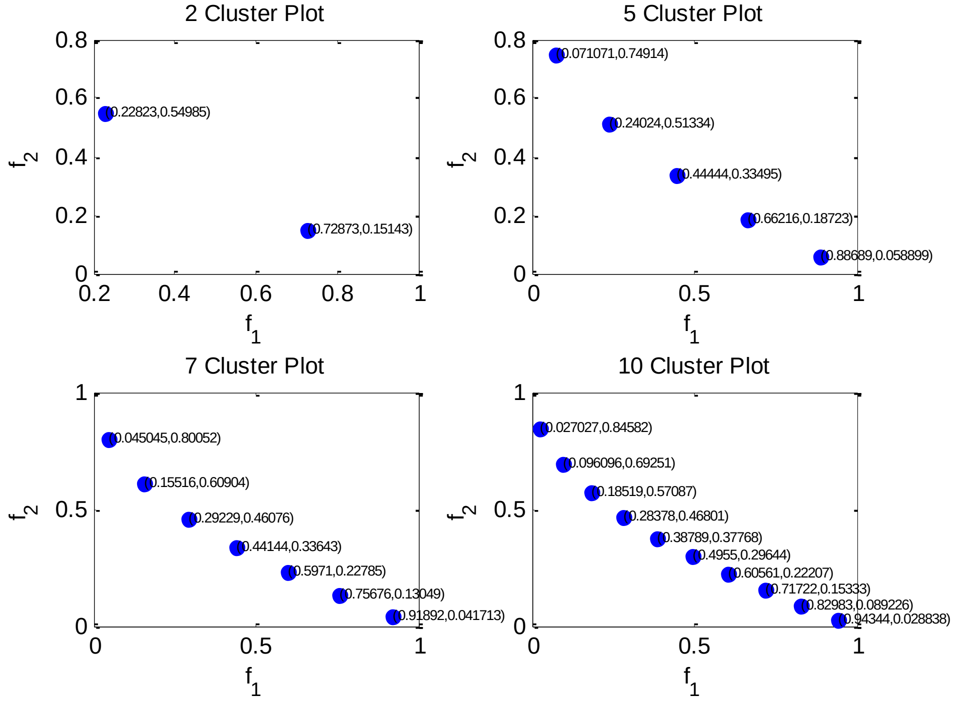

Figure 29: Pareto front centroids of problem UF1for k=2,5,7, and 10 clusters

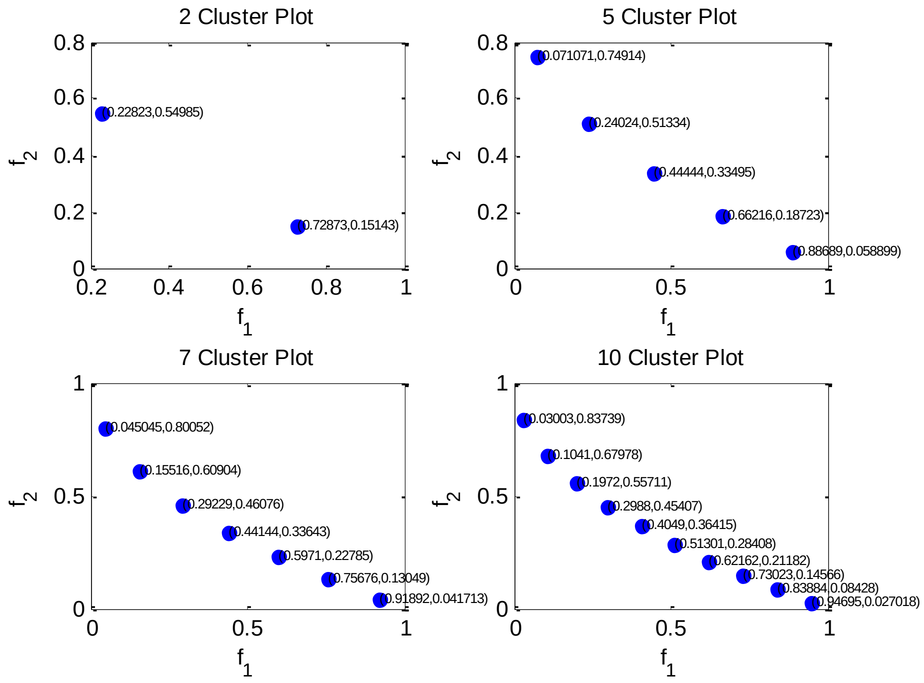

Figure 30: Pareto front centroids of problem UF2for k=2,5,7, and 10 clusters

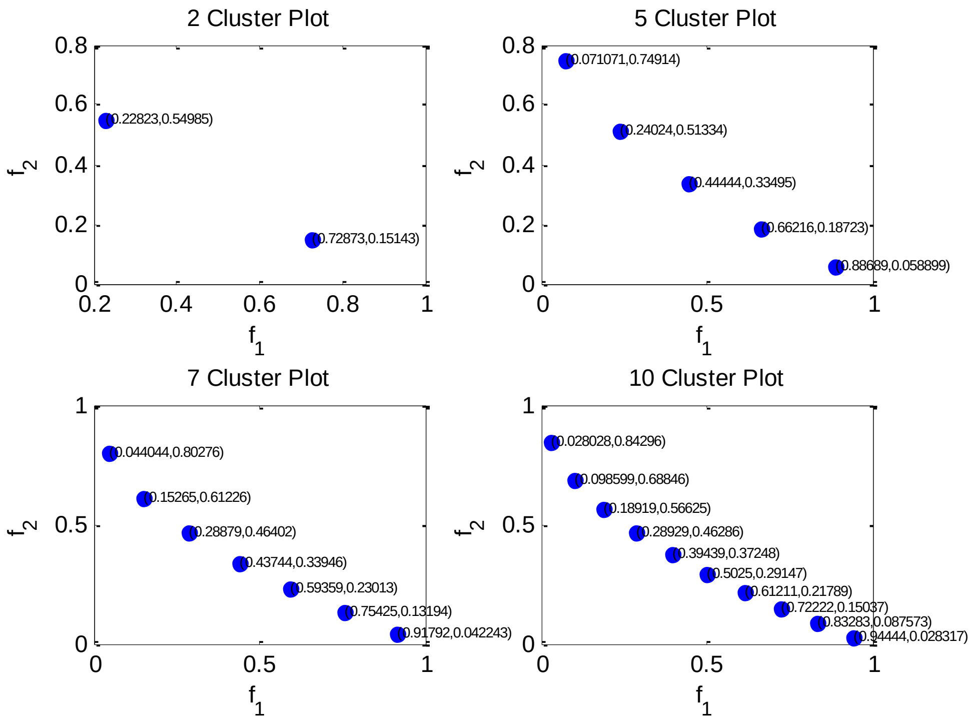

Figure 31: Pareto front centroids of problem UF3for k=2,5,7, and 10 clusters

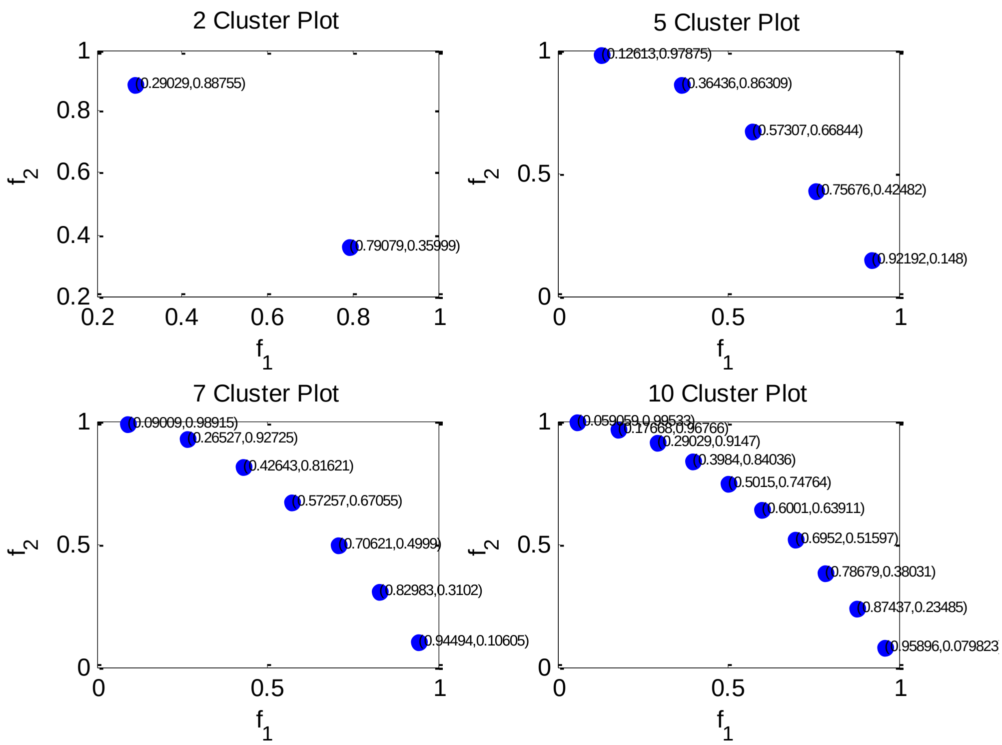

Figure 32: Pareto front centroids of problem UF4for k=2,5,7, and 10 clusters

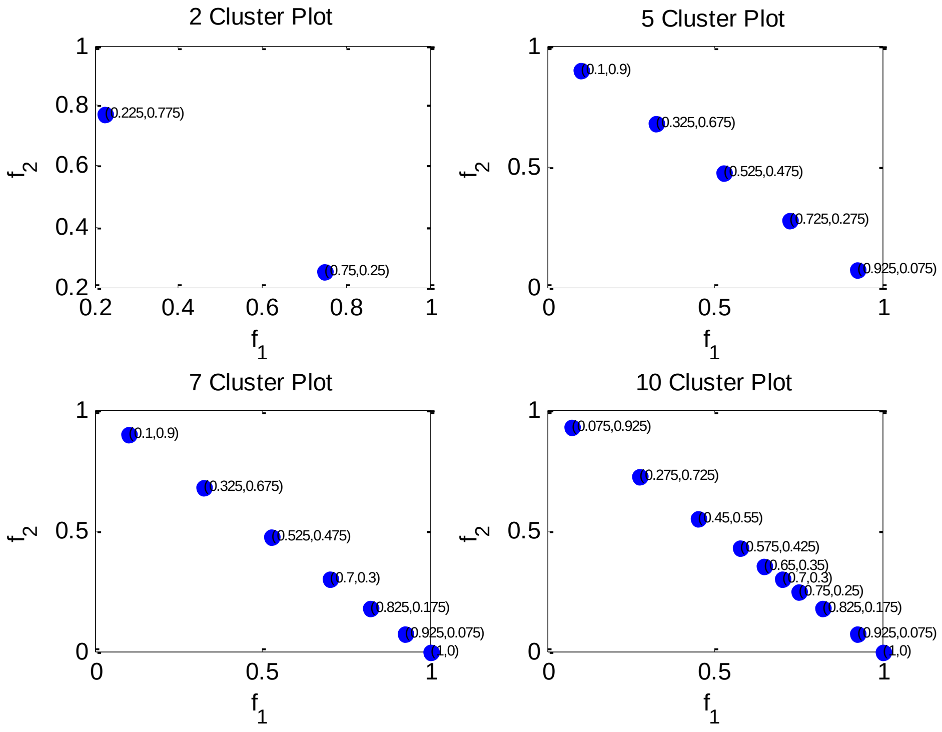

Figure 33: Pareto front centroids of problem UF5for k=2,5,7, and 10 clusters

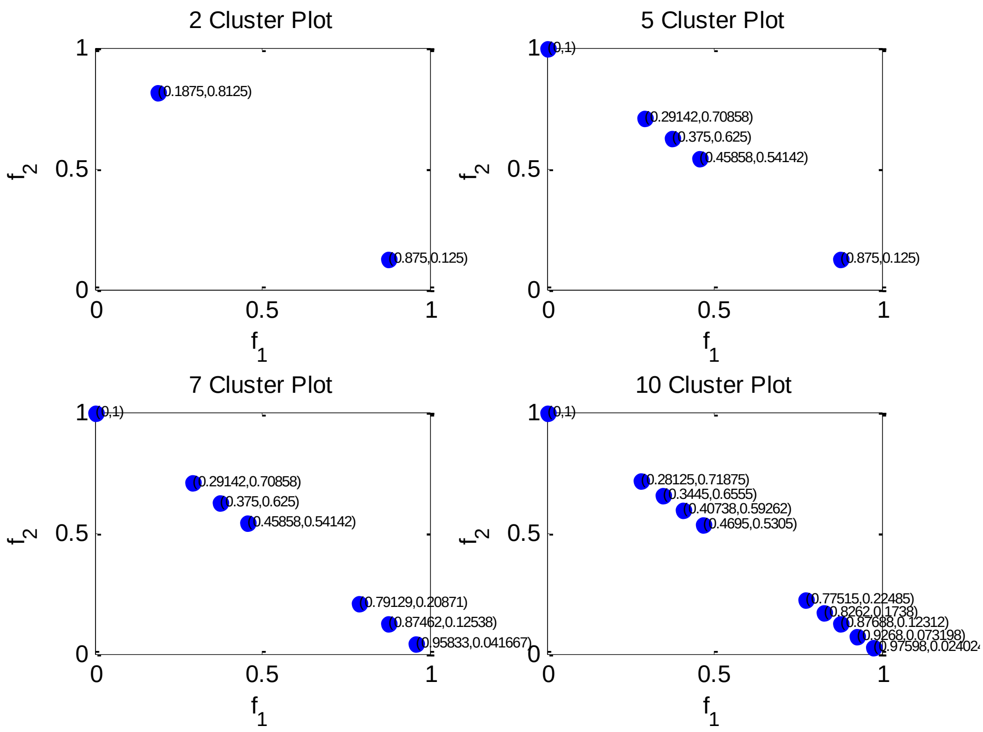

Figure 34: Pareto front centroids of problem UF6for k=2,5,7, and 10 clusters

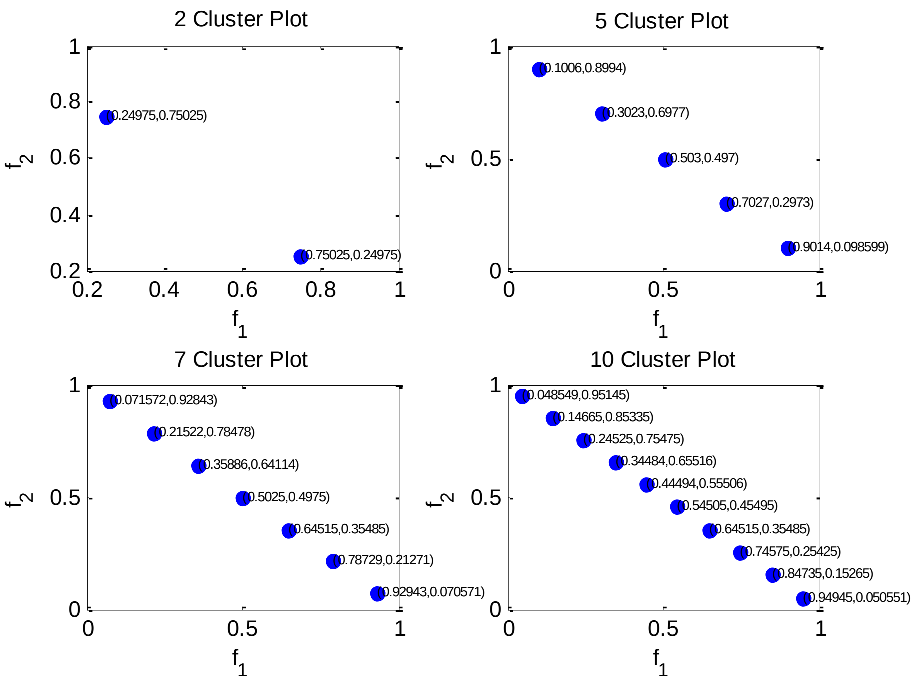

Figure 35: Pareto front centroids of problem UF7for k=2,5,7, and 10 clusters

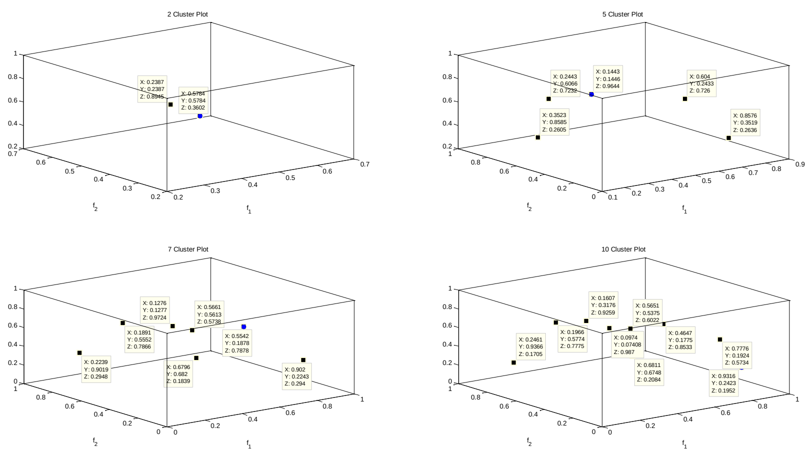

Figure 36: Pareto front centroids of problem UF8for k=2,5,7, and 10 clusters

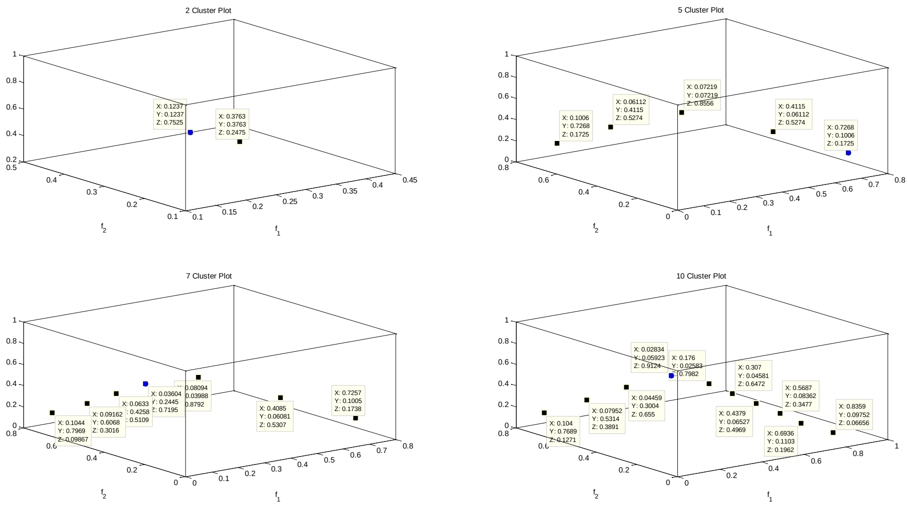

Figure 37: Pareto front centroids of problem UF9for k=2,5,7, and 10 clusters

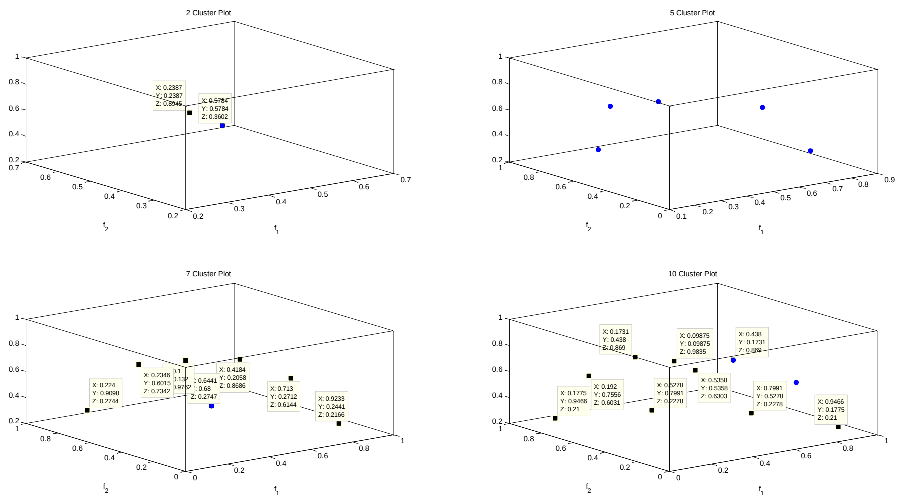

Figure 38: Pareto front centroids of problem UF10for k=2,5,7, and 10 clusters

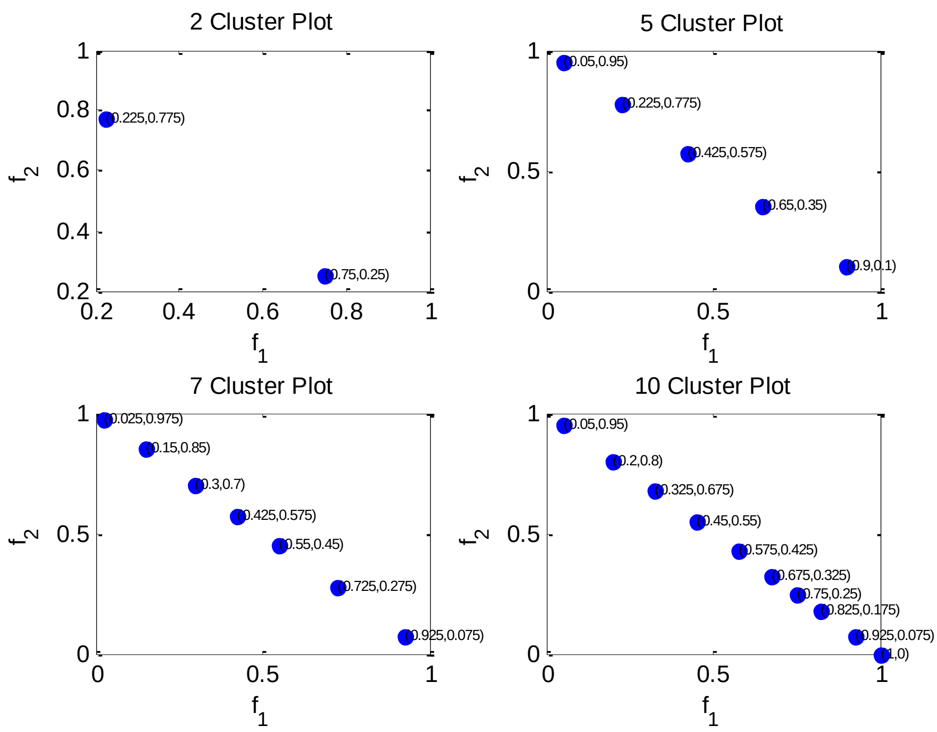

Figure 39: Pareto front centroids of problem CF1 for k=2,5,7, and 10 clusters

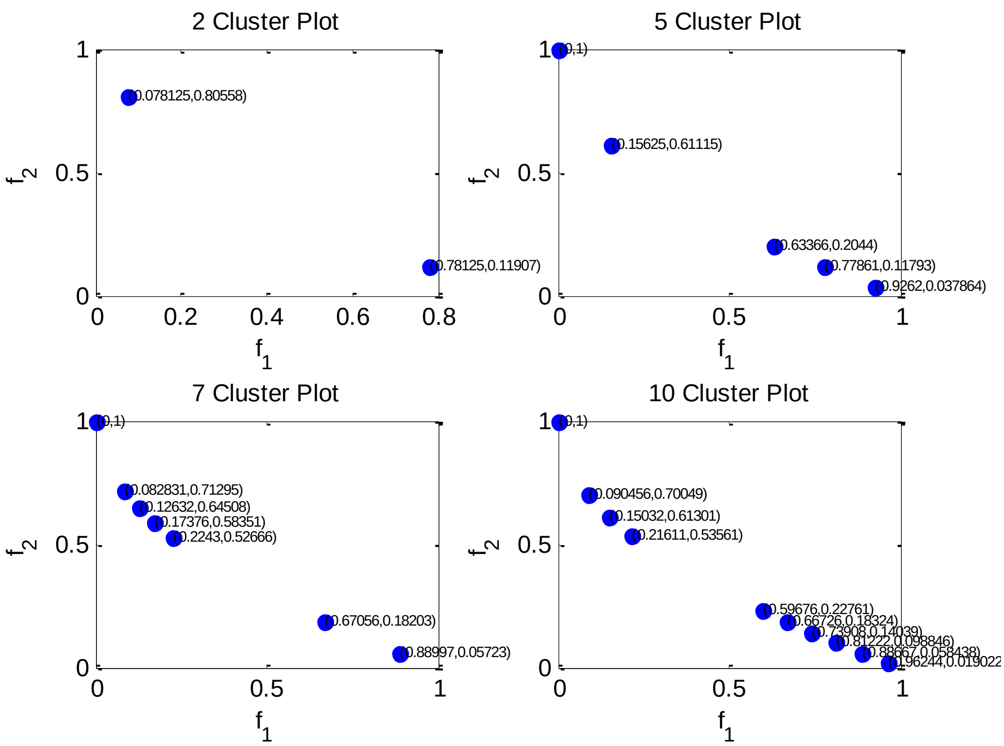

Figure 40: Pareto front centroids of problem CF2 for k=2,5,7, and 10 clusters

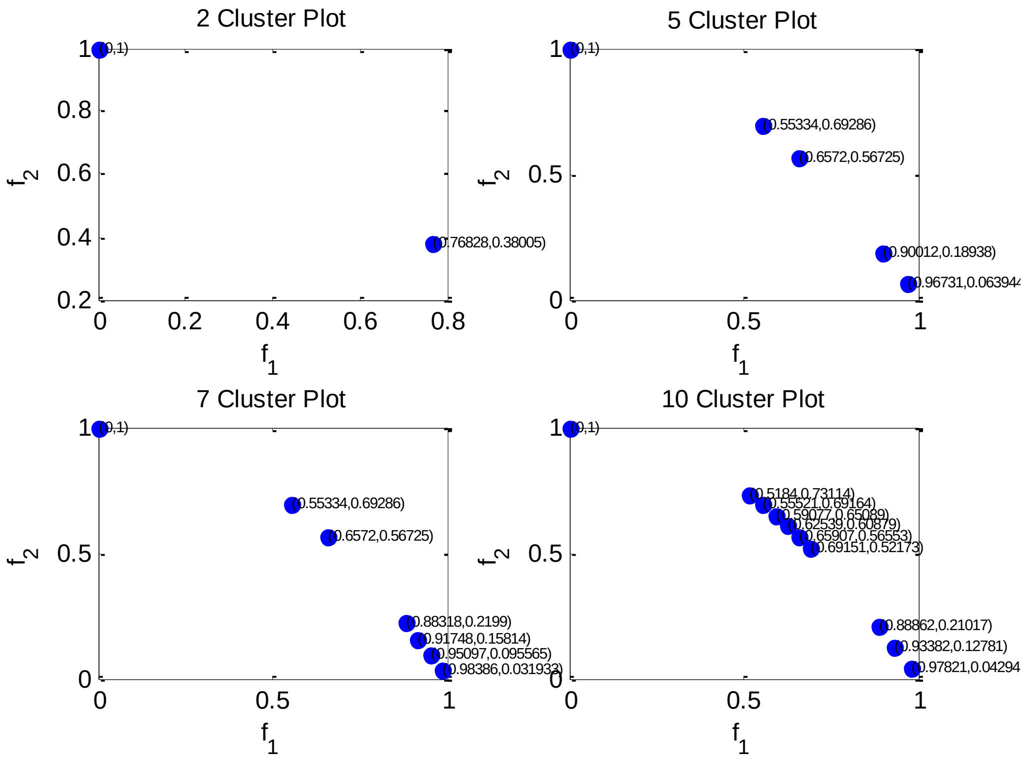

Figure 41: Pareto front centroids of problem CF3 for k=2,5,7, and 10 clusters

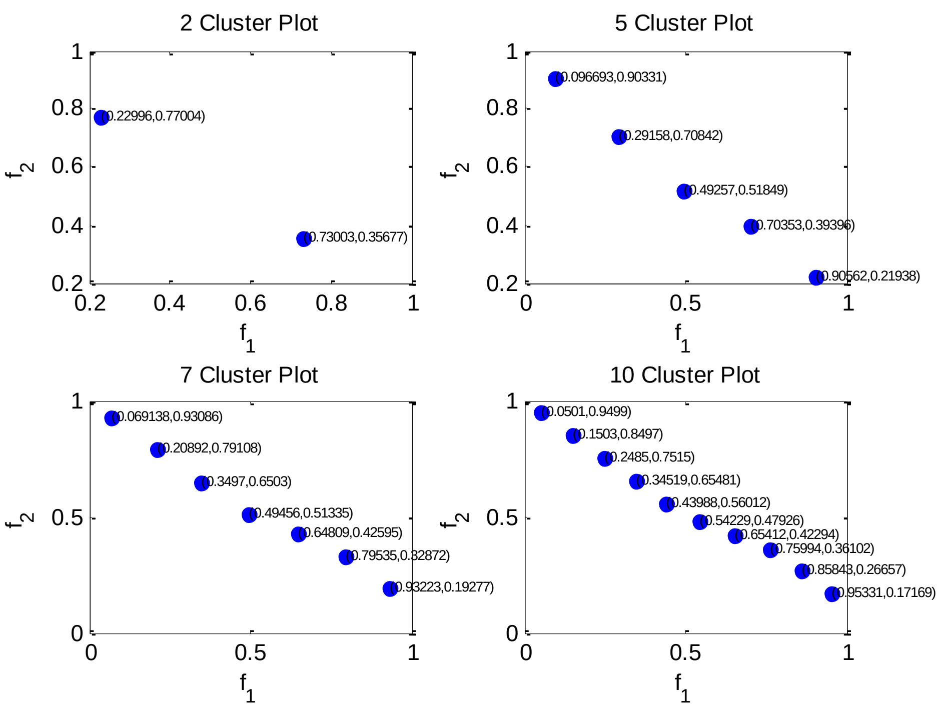

Figure 42: Pareto front centroids of problem CF4 for k=2,5,7, and 10 clusters

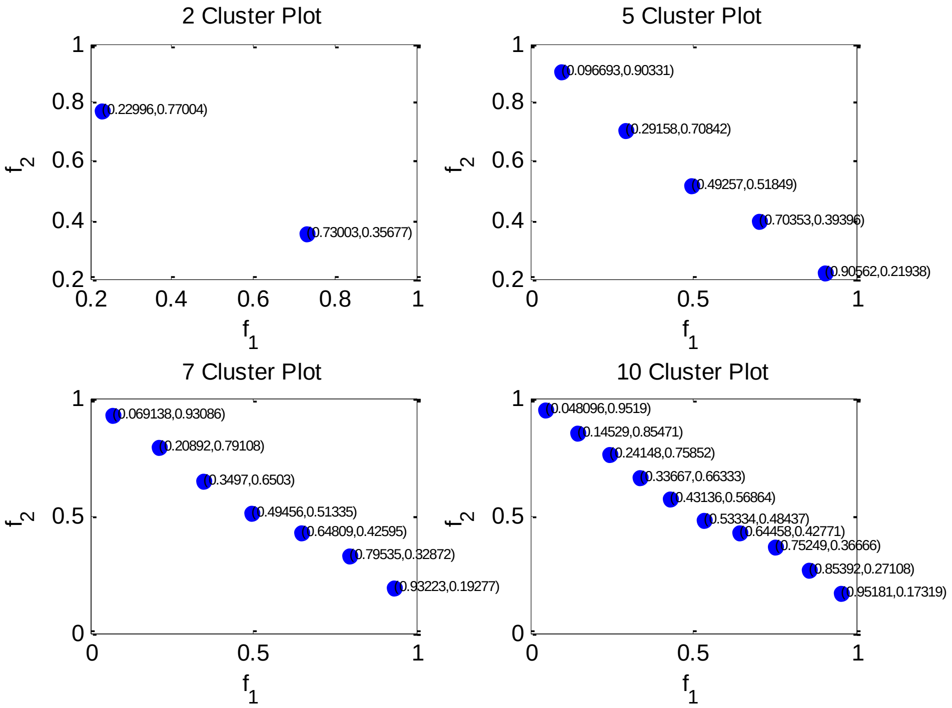

Figure 43: Pareto front centroids of problem CF5 for k=2,5,7, and 10 clusters

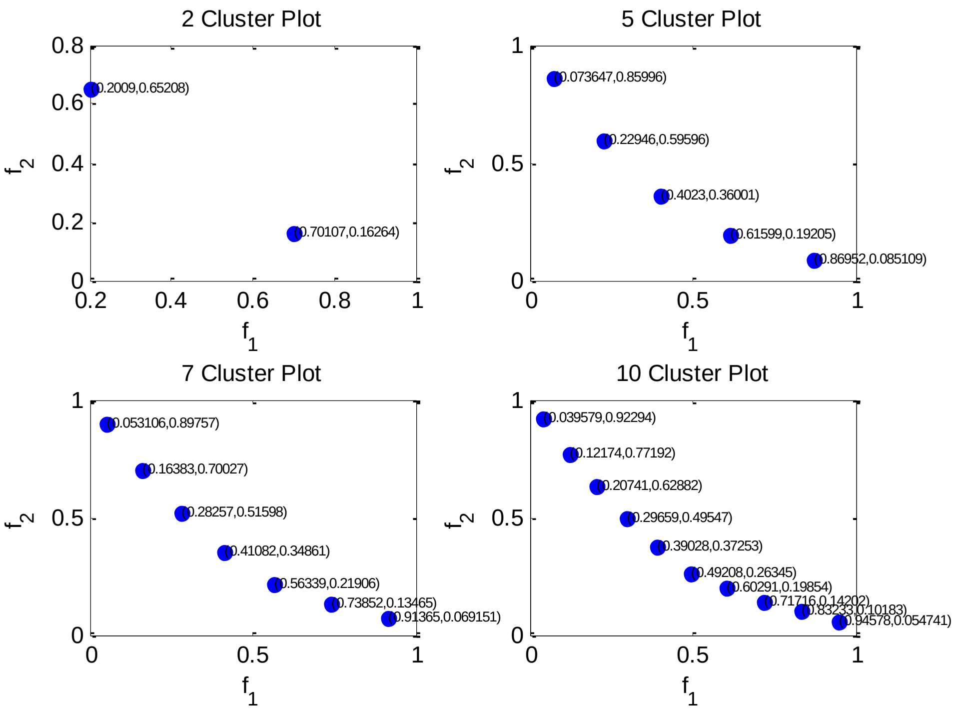

Figure 44: Pareto front centroids of problem CF6 for k=2,5,7, and 10 clusters

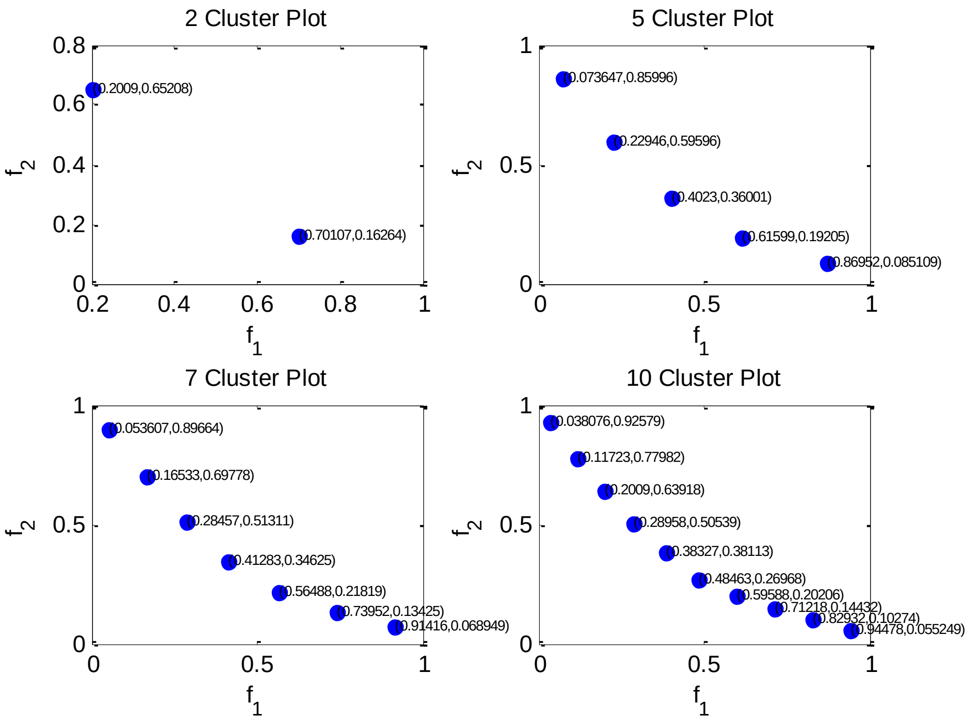

Figure 45: Pareto front centroids of problem CF7 for k=2,5,7, and 10 clusters

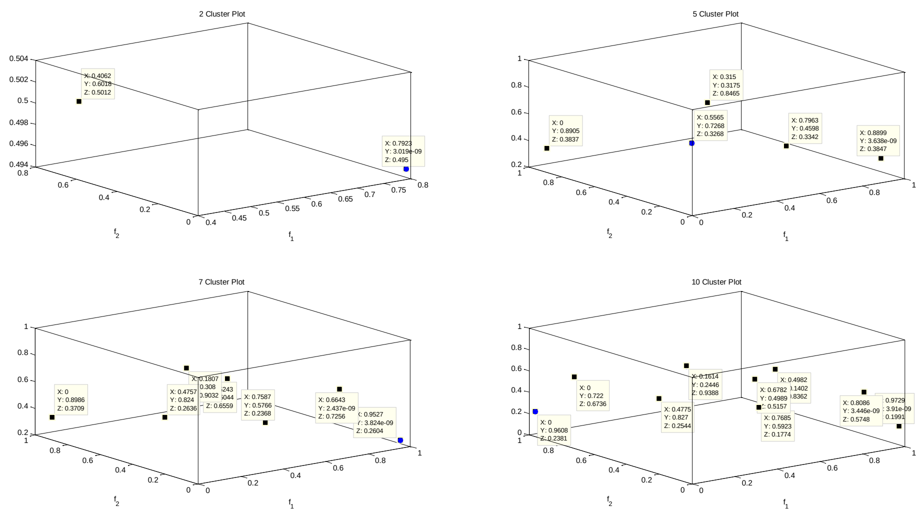

Figure 46: Pareto front centroids of problem CF8 for k=2,5,7, and 10 clusters

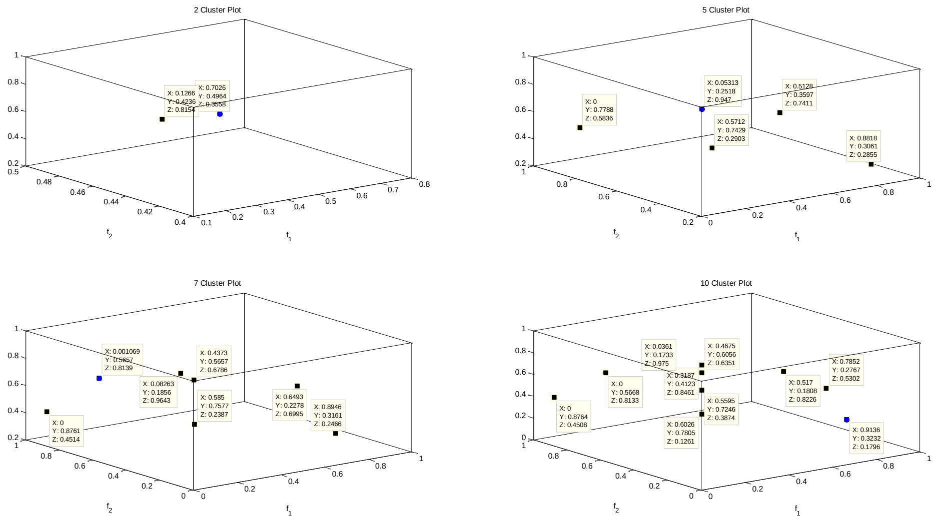

Figure 47: Pareto front centroids of problem CF9 for k=2,5,7, and 10 clusters

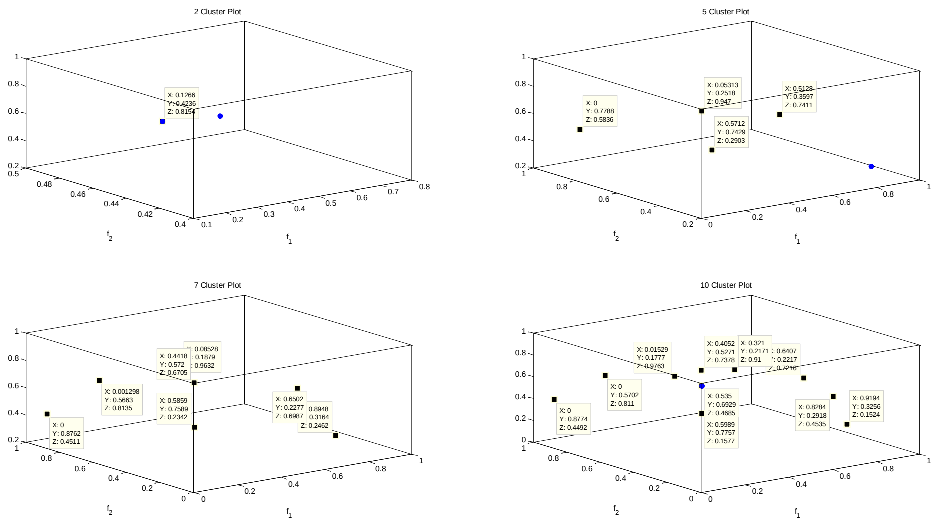

Figure 48: Pareto front centroids of problem CF10 for k=2,5,7, and 10 clusters

Solving MOOPs using multi-objective EAs not gives a single optimal solution (i.e., only one point), but it gives a set of Pareto optimal solutions (nondominated solutions), in which one objective value cannot be improved without sacrificing other objectives values. But, for all practical applications we need to select one solution or a set of limited number of solutions; which should preserve the diversity of Pareto front. This paper presents hybrid genetic K-means approach which intends to not only make Pareto front tractable but to select the most compromise set of solution depending on the DM choice. The proposed methodology is tested by various problems of MOO test instances: the Special Session and Competition 2009 (CEC09) to demonstrate its ability to select the most compromise set of solution. Results of simulation experiments using several problems form CEC09 prove our claim and show the ability of the proposed algorithm to solve the clustering problem for the Pareto front. From the above results the selected solution has a diversity characteristics also cover the entire Pareto front domain and allows the DM to attain a reasonable representation of the Pareto-optimal front. Finally, the proposed algorithm has a diversity preserving mechanism to overcome the cruise of huge number of Pareto front.

CONCLUSION

Solving MOOPs using multi-objective EAs not gives a single optimal solution, but a set of Pareto-optimal solutions, in which one objective cannot be improved without sacrificing other objectives. But, for practical applications we need to select one solution or a set of limited number of solutions; which preserve the diversity of Pareto front. In this paper, a new enhanced evolutionary algorithm is presented, our proposed algorithm has been enriched with modified k-means cluster scheme, On the first hand, in phase I, K-means clustering technique is implemented to partition the population into a determined number of subpopulation with-dynamic-sizes. In this way, different genetic algorithms (GAs) operators can apply to each sub-population, instead of one GAs operator applied to the whole population. On the other hand, phase II applies K-means algorithm to make the algorithms practical by allowing the decision maker to control the precision of the Pareto-set approximation by choosing a suitable number of the needed clusters. The proposed methodology is examined by various problems of MOO test instances: the Special Session and Competition 2009 (CEC09) to demonstrate its ability to select the most compromise set of solution. Results of simulation experiments using several problems form CEC09 prove our claim and show the ability of the proposed algorithm to solve the clustering problem for the Pareto front.

The main conclusions of the research work presented in this roject can be summarized as follows:

1. In order to improve the performance of the GAs, hybridization of the GAs with k-means is presented. This algorithm is made of a classical genetic algorithm based on the ideas of co-evolution coupled with a k-means as cluster technique in order to improve the spread of the solutions found so far.
2. The numerical analysis shows that our hybridization algorithm is very efficient for benchmark problems.
3. The proposed approach was capable to eliminate the drawbacks of GAs
4. Allowing a decision maker to control the precision of Pareto front by defining desired number of cluster k values.
5. Empirical results show that our approach is very efficient in detecting the comprise set of solution from the Pareto set.
6. The proposed algorithm has been effectively applied to deal with the MOP, with no limitation in solving higher-dimensional problems.
7. The proposed approach was capable to find well distributed Pareto-front.
8. The success of our approach on most of the test problems not only provides confidence but also stress the importance of hybrid evolutionary algorithms in solving multiobjective optimization problems.
9. The reality of using the proposed approach to handle complex problems of realistic dimensions has been approved due to procedure simplicity.

Recommendations for Future Researches

Several possible areas for future work have arisen from this research concerning cluster analysis. Those are summarized as follows:

1. An implementation of the proposed algorithm for solving real problems.
2. Performance of the proposed algorithm can also be improved by further investigation of the various GAs parameters.
3. Evolutionary Algorithms, such as GAs, are a very important area of investigation and have personally provided a great deal of interesting work and stimulation in the complexity and diversity of applications in the last few years has been enormous.

REFERENCES

[1] K. Miettinen, "Non-linear multiobjective optimization", Dordrecht: Kluwer Academic Publisher (2002).

[2] A. A. Mousa, "A new Stopping and Evaluation Criteria for Linear Multiobjective Heuristic Algorithms", Journal of Global Research in Mathematical Archives , vol. 1(8), August (2013).

[3] QiSen Cai, Defu Zhang, Bo Wu, Stehpen C.H. Leung , "A Novel Stock Fore-casting Model based on Fuzzy Time Series and Genetic Algorithm", Procedia Computer Science, vol. 18, pp. 1155-1162,(2013).

[4] He Ni, Yongqiao Wang , "Stock index tracking by Pareto efficient genetic algorithm", Applied Soft Computing, Vol. 13(12), pp. 4519-4535, December (2013).

[5] Janko Straßburg, Christian Gonzàlez-Martel, Vassil Alexandrov, "Parallel genetic algorithms for stock market trading rules", Procedia Computer Science, Vol. 9, pp. 1306-1313, (2012).

[6] Xiao-Wei Xue, Min Yao, Zhaohui Wu, Jianhua Yang , "Genetic Ensemble of Extreme Learning Machine", Neurocomputing, vol. 129, pp. 175- 184, (2014).

[7] Jiadong Yang, Hua Xu, Peifa Jia , "Effective search for genetic-based machine learning systems via estimation of distribution algorithms and embedded feature reduction techniques", Neurocomputing, vol.113, pp. 105-121, August (2013).

[8] Hongming Yang, Jun Yi, Junhua Zhao, ZhaoYang Dong , "Extreme learning machine based genetic algorithm and its application in power system economic dispatch ", Neurocomputing, vol. 102, pp. 154-162, February (2013).

[9] M.S.Osman , M.A.Abo-Sinna , and A.A. Mousa " A Solution to the Optimal Power Flow Using Genetic Algorithm ", Journal of Applied Mathematics & Computation (AMC), vol .155(2), pp 391-405, August (2004).

[10] M.S.Osman, M.A.Abo-Sinna, and A.A. Mousa

"Epsilon-Dominance based Multiobjective Genetic Algorithm for Economic Emission Load Dispatch Optimization Problem", Electric Power Systems Research, vol. 79, pp.1561–1567,(2009).

[11] B. N. AL-Matrafi , A. A. Mousa , "Optimization methodology based on Quantum computing applied to Fuzzy practical unit commitment problem", International Journal of Scientific & Engineering Research, vol. 4(11), pp. 1138-1146, November(2013).

[12] Abd Allah A. Galal, Abd Allah A. Mousa, Bekheet N. Al-Matrafi , "Hybrid Ant Optimization System for Multiobjective Optimal Power Flow Problem Under Fuzziness", Journal of Natural Sciences and Mathematics, vol. 6(2), pp.179-199, July (2013).

[13] C.M. Fonseca, P. J. Fleming, " Multiobjective optimization and multiple constraint handling with evolutionary algorithms", : Part I: a unified formulation. IEEE Transactions on Systems and Cybernetics — Part A: Systems and Humans, vol. 28(1), pp. 26–37, (1998)

[14] K. Harada, J . Sakuma, Ono Isao, S. Kobayashi, "Constraint-handling method for multiobjective function optimization: pareto descent repair operator", In: Proceedings of the fourth international conference on evolutionary multicriterion optimization, pp. 156–70, (2007).

[15] Ahmed A. EL-Sawy, " Mohamed A. Hussein, EL-Sayed M. Zaki, A. A. Mousa, "An Introduction to Genetic Algorithms: A survey A practical Issues", International Journal of Scientific & Engineering Research, Volume 5(1), pp. 252-262, January(2014).

[16] A. A. Mousa, M. A. El-Shorbagy, Identifying a Satisfactory Operation Point for Fuzzy Multiobjective Environmental/Economic Dispatch Problem, American Journal of Mathematical and Computer Modelling,1(1): 1-14,(2016).

[17] Malki, AAL; Rizk, MM; El-Shorbagy, MA; Mousa, AA; ,Identifying the Most Significant Solutions from Pareto Front Using Hybrid Genetic K-Means Approach,International Journal of Applied Engineering Research,11,14,8298-8311,(2016).

[18] Al Malki, A., Rizk, M.M., El-Shorbagy, M.A. and Mousa, A.A. , Hybrid Genetic Algorithm with K-Means for Clustering Problems, Open Journal of Optimization, 5, 71-83,(2016).

[19] A.A. Mousa, M.A. El-Shorbagy and M. A. Farag, K-means-Clustering Based Evolutionary Algorithm for Multi-objective Resource Allocation Problems, Appl. Math. Inf. Sci. 11, No. 6, 1-12, (2017).

[20] Osman M.S., M.A.Abo-Sinna, and A.A. Mousa " IT-CEMOP: An Iterative Co-evolutionary Algorithm for Multiobjective Optimization Problem with Nonlinear Constraints" Journal of Applied Mathematics & Computation (AMC) 183, pp373-389, (2006).

[21] Waiel. F. Abd El-Wahed, A.A.Mousa , M. A. El-Shorbagy, Integrating particle swarm optimization with genetic algorithms for solving nonlinear optimization problems, Journal of Computational and Applied Mathematics 235:1446–1453, (2011).

[22] M. Verma, M. Srivastava, N. Chack, A.K. Diswar, N. Gupta, A Comparative Study of Variou Clustering Algorithms in Data Mining, International Journal of Engineering Reserch and Applications (IJERA) 2(3), 1379-1384, (2012).

[23] S. Levine, I. Pilowski, D.M. Boulton, The classication of depression by numerical taxonomy. The British Journal of Psychiatry 115, 937-945, (1969).

[24] S. Dolnicar, Using cluster analysis for market segmentation - typical misconceptions, established methodological weaknesses and some recommendations for improvement, University of Wollongong Research Online, (2003).

[25] F.R. Hodson, Numerical typology and prehistoric archaeology. In F. R. Hodson, D. G. Kendall, and P. A. Tautu, editors, Mathematics in the Archaeological and Historical Sciences. University Press, Edinburgh, (1971).

[26] D. Pratima, N. Nimmakanti, P. Recognition, Algorithms for Cluster Identification Problem. Special Issue of International Journal of Computer Science & Informatics (IJCSI) II (1, 2) (2011) 2231–5292, (2011).

[27] Doreswamy, K.S. Hemanth, Similarity Based Cluster Analysis on Engineering Materials Data Sets, Advances in Intelligent Systems and Computing 167,161-168, (2012).

[28] D. Dilts, J. Khamalah , A. Plotkin, Using Cluster Analysis for Medical Resource Decision Making, Cluster Analysis for Resource Decisions 15(4) (1995) 333-347.

[29] A. Jain, Data clustering: 50 years beyond k-means. Pattern Recognition Letters 31(8) (2010) 651–666.

[30] J. Macqueen, The classification problem, Western Management Science Institute Working Paper No. 5 (1962).

[31] L. Morissette, S. Chartier, The k-means clustering technique: General considerations and implementation in Mathematical, Tutorials in Quantitative Methods for Psychology 9 (1) (2013) 15-24.

[32] A. Jain, Data clustering: 50 years beyond k-means. Pattern Recognition Letters 31(8) (2010) 651–666.

[33] Hongming Yang, Jun Yi, Junhua Zhao, ZhaoYang Dong ,Extreme learning machine based genetic algorithm and its application in power system economic dispatch Neurocomputing, Volume 102, 15 February 2013, Pages 154-162

[34] Z. Michalewiz ,"Genetic Algorithms + Data Structures = Evolution Programs",Springer-Verlag,3 Rd Edition (1996).

[35] M.S.Osman , M.A.Abo-Sinna , and A.A. Mousa " An Effective Genetic Algorithm Approach to Multiobjective Resource Allocation Problems (MORAPs) " Journal Of Applied Mathematics & Computation 163, pp. 755-768, (2005).

[36] M. A. Farag, M. A. El-Shorbagy, I. M. El-Desoky, A. A. El-Sawy, Genetic Algorithm Based on K-Means-Clustering Technique for Multi-Objective Resource Allocation Problems, British Journal of Applied Science & Technology 8(1) (2015), 80-96 .

[37] T. Murata, H. Ishibuchi, H. Tanaka, Multiobjective Genetic Algorithm and its Application Toflowshop Scheduling, Computers and Industrial Engineering 30 (4) (1996) 957–968.

[38] K. Arai, X.Q. Bu. ISODATA Clustering with Parameter (Threshold For Merge And Split) Estimation Based on GA: Genetic Algorithm, Reports of the Faculty of Science and Engineering, Saga University 36(1) (2007)17-23.

[39] C.A. Anderson, K.F. Jones, J. Ryan, A Two-Dimensional Genetic Algorithm for the ISING Problem, Complex Systems 5 (1991) 327-333

[40] C. P. Cheng, C. W. Liu, Unit Commitment by Annealing Genetic Algorithm, International Journal Electrical Power & Energy Systems 24(2) (2002) 149–158

[41] L. Sun, Y. Zhang, C. Jiang, A Matrix Real-Coded Genetic Algorithm to the Unit Commitment Problem, Electric Power Systems Research 76(9-10) (2006) 716–728.

[42] T. Senjyu, H. Yamashiro, K. Uezato, T. Funabashi, A Unit Commitment Problem By Using Genetic Algorithm Based On Unit Characteristic Classification, 2002. IEEE Power Engineering Society Winter Meeting 1 (2002) 58 - 63

[43] A. Trivedi, Enhanced Multiobjective Evolutionary Algorithm based on Decomposition for Solving the Unit Commitment Problem, IEEE Transactions on Industrial Informatics 11 (6) (2015) 1346 - 1357.

[44] M. A. Farag, M. A. El-Shorbagy, I. M. El-Desoky, A. A. El-Sawy, A. A. Mousa, Binary-Real Coded Genetic Algorithm Based K-Means Clustering for Unit Commitment Problem, Applied Mathematics 6 (2015) 1873-1890.

[45] A.A. Mousa, M.A. El-Shorbagya, W.F. Abd-El-Wahed, Local Search Based Hybrid Particle Swarm Optimization Algorithm for Multiobjective Optimization , Swarm and Evolutionary Computation 3 (2012) 1–14.

[46] Q. Zhang, A. Zhou, S.Z. Zhao, P.N. Suganthan, W. Liu, S. Tiwari, Multiobjective optimization test instances for the CEC09 special session and competition, special session on performance assessment of multiobjective optimization algorithms, Technical Report, University of Essex, Colchester, UK and Nanyang Technological University, Singapore (2008).