Evolutionary Algorithm Using Adaptive Fuzzy Dominance and Reference Point for Many-Objective Optimization

Siddhartha Shankar Das, Md. Monirul Islam, *Member IEEE*

Department of Computer Science and Engineering, Bangladesh University of Engineering And Technology (BUET), Dhaka Bangladesh

Naheed Anjum Arafat

School of Computing, National University of Singapore (NUS), Singapore

Abstract

Many-objective optimization is very important for numerous practical applications. It, however, poses a great challenge to the Pareto dominance based evolutionary algorithms. In this paper, a fuzzy dominance based evolutionary algorithm is proposed for many-objective optimization. The essence of the proposed algorithm is that it adaptively determines the fuzzy membership function for each objective of a given many-objective optimization problem. Furthermore, it emphasizes both convergence and diversity of all the evolved solutions in the same way by using one selection criterion. This is why our algorithm employs the reference points for clustering the evolved solutions and selects the best ones from different clusters in a round-robin fashion. The proposed algorithm has been tested extensively on a number of benchmark problems in evolutionary computing, including eight Waking-Fish-Group (WFG) and three Deb-Thiele-Laumanns-Zitzler (DTLZ) problems having 2 to 25 objectives. The experimental results show that the proposed algorithm is able to solve many-objective optimization problems efficiently, and it is compared favorably with the other evolutionary algorithms devised for such problems. A parametric study is also provided to understand the influence of a key parameter of the proposed algorithm.

Keywords: Many-objective optimization, Evolutionary algorithm, Pareto dominance, Fuzzy dominance, Reference point

1. Introduction

Handling an optimization problem with a large number of objectives (more than three), also known as many-objective optimization problem (MaOP), using multi-objective evolutionary algorithm (EMO) has been one of the major research topics in recent years. Although Pareto dominance based EMO algorithms can efficiently solve optimization problems with two/three objectives, they face several difficulties in solving MaOPs [1], [2]. The reduction of selection pressure is the main difficulty faced by such algorithms. Another prominent problem is the conflict between convergence and diversity, which also aggravates with the increase of objectives [3].

There are also several studies that deal with improving diversity maintenance among the solutions of an evolving population. Adra and Fleming [4] employ a diversity management operator to adjust diversity requirement in the mating and environmental selection of an EMO algorithm. Wang *et al.* [5] propose a new diversity measure which can be used for enhancing diversity in MaOPs. In [6], Li *et al.* propose a general modification of density estimation, termed shift-based density estimation (SDE), to make Pareto-based algorithms suitable for MaOPs. A knee point driven approach [7] prefers knee point among non-dominated solutions for maintaining diversity. Deb and Jain [1] propose NSGA-III in which they employ reference point based clustering for ensuring diversity. In [8], Mario *et al.* use hierarchical clustering in decision space for diversity management.

As an alternative, two types of non-Pareto based approaches have been found to be promising in the EMO literature. They are decomposition based approaches and indicator based approaches. The former approach first decomposes an MOP into a number of single objective optimization sub-problems and then solves them simultaneously. Multi-objective evolutionary algorithm based on decomposition (MOEA/D) [9] is the most typical implementation of this class. This algorithm and its variants [10–12] have been quite successful in solving various MaOPs. Another decomposition based algorithm RVEA [13] uses reference vectors to decompose solutions into different sub-problems. In [14], a preference inspired coevolutionary algorithms, PICEAg, is proposed based on a concept of co-evolving a population of candidate solutions with a set of goals.

The main idea of indicator based approaches is to use a single performance indicator to optimize a desired property of an evolving population. Among the different indicators, hypervolume [15] is the only quality measure known to be Pareto-compliant and is ever used in multiobjective search. However, one prominent problem of hypervolume based algorithms is their extreme computational overhead. Bader and Zitzler [16] proposed a hypervolume estimation (HypE) algorithm to reduce computational burden. Recently, a two-archive algorithm (Two_Arch) for MaOP [17] has been proposed by combining the indicator and Pareto based approaches in one algorithm.

Despite the recent advancements in solving MaOPs, more effective EMOs are still needed. Among the non-Pareto based approaches, the notion of fuzzy concept has proven to be effective for MaOPs. Its strength comes from the fact that fuzzy concept can continuously differentiate solutions into different degrees of optimality beyond the classification of the original Pareto dominance. Which is beneficial for MaOPs because solutions which were previously incomparable can be now compared and complete or partial order of solutions can be found. Moreover given the membership functions, the fuzzy procedure does not usually require any extra computational overhead even for a very large number of objectives. The simplicity inherent within the fuzzy dominance computation makes it an appealing candidate for solving MaOPs. But fuzzy concepts have the issues of diversity loss and selection of appropriate parameters in different fuzzy steps. There are a few studies [18–22] that utilize this concept in solving MaOPs using evolutionary algorithms and the field is still under-explored.

This paper proposes a fuzzy dominance based evolutionary algorithm (F-DEA) for efficiently solving MaOPs. The proposed algorithm exploits the strength of fuzzy dominance in conjunction with clustering based active diversity promotion mechanism in order to efficiently solve MaOPs. The novelty of F-DEA comes from the following new features,

- Diversity maintenance is the primary issue faced by any fuzzy based approaches. To maintain diversity among solutions, F-DEA employs reference points based clustering to select solutions for the next generation of an evolutionary process. To the best of our knowledge, F-DEA is the first algorithm that employs fuzzy dominance and reference point synergistically. Unlike existing reference point based approaches [1, 13, 14, 23–25], F-DEA uses preferred reference points based clustering to provide better cluster uniformity, remove dependency on population size and effectively handle irregular-shaped Pareto fronts.

- F-DEA introduces the anti-symmetric Sigmoid membership function with an aim of ensuring proper discrimination ability of the membership function. Existing approaches either use domain knowledge [21] or approximation procedure [18].

- To handle the scaling issue of objectives, F-DEA uses a separate membership function for each objective and estimates its parameter(s) adaptively. The estimation procedure handles the bias induced by isolated solutions.

- The proposed algorithm emphasizes both convergence and diversity in the same way from beginning to end of an evolutionary process. Existing fuzzy based algorithms use fuzzy dominance as a primary selection criterion and diversity measure as a secondary one. This is problematic in the sense that algorithms would rarely employ the secondary criterion as fuzzy dominance is capable of discriminating solutions.

The rest of this paper is organized as follows. Section 2 introduces the basic concepts of fuzzy in relation with EMO algorithms, issues arises therein and related works. We present the proposed F-DEA at length in Section 3. In Section 4, several different kinds of algorithms for MaOPs are included in our experimental studies, comparisons, and discussions. Finally, Section 5 concludes the paper with a brief summary and a few remarks.

2. Preliminaries and Related Work

2.1. Preliminaries

The term fuzzy logic refers to a form of many-valued logic in which the output of a set of inputs may be any real value between 0 and 1. The application of this concept to solve a problem usually requires three steps: fuzzification, fuzzy inference and defuzzification. Fuzzification converts input data into membership degrees (values) using some functions, called membership functions. As a membership function, the left Gaussian function, as depicted in Fig. 1, is usually employed in the EMO literature (e.g. [18], [21]). An interesting feature of this function is its monotonic decreasing nature. The analytic form of this function is

$$\gamma_g(v) = \frac{1}{\sigma\sqrt{2\pi}} \mathrm{e}^{-\frac{(v-c)^2}{2\sigma^2}}. \tag{1}$$

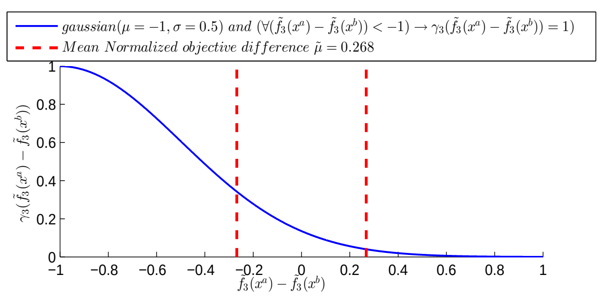


Figure 1: Left Gaussian Membership Function. This particular case shows the position of mean nomalized objective difference value ($\bar{\mu_3}$) for the third objective of WFG2 problem obtained from the 250^{th} generation of a particular seed.

It is seen from Eq. (1) that the Gaussian function is defined by two parameters, c and σ. While c represents mean and it is set to -1, σ defines the spread of the Gaussian function and it is set to 0.5 as a compromise. It has been indicated that setting σ too large or too small leads to the inability to discriminate $v \in [-1, 0]$ or $v \in [0, 1]$ [18]. Even setting σ in this way may create uneven discrimination between the entire domain of $v \in [-1, 1]$. This can be attributed from Fig. 1 that the discrimination ability of $\gamma_g(v)$ for $v \in [-1, 0]$ and for $v \in [0, 1]$ is not similar because the curve's nature in the said ranges are not identical. Pareto dominance based EMO algorithms compare two solutions $\mathbf{x}^a = x_1^a, x_2^a, \ldots, x_n^a$ and $\mathbf{x}^b = x_1^b, x_2^b, \ldots, x_n^b$ based on their objective vectors $\mathbf{f}(\mathbf{x}^a) = f_1(\mathbf{x}^a), f_2(\mathbf{x}^a), \ldots, f_m(\mathbf{x}^a)$ and $\mathbf{f}(\mathbf{x}^b) = f_1(\mathbf{x}^b), f_2(\mathbf{x}^b), \ldots, f_m(\mathbf{x}^b)$, respectively. In line with this, fuzzy based EMO algorithms utilize the difference of objective vectors as the argument of the membership function i.e., $v = f_i(\mathbf{x}^a) - f_i(\mathbf{x}^b)$ or $v = f_i(\mathbf{x}^b) - f_i(\mathbf{x}^a)$.

An MaOP may or may not have an identical range of values for its each objective, f_i. To handle this issue, we may use a different membership function for each of different objectives i.e., a separate σ for each objective. A similar concept is used in [21] where σ is chosen manually. This will, however, require rich domain knowledge for each problem we like to solve. Alternatively, we may normalize v by the maximum difference among all pairs of solutions in that objective and can use the same membership function for all objectives. This alternative approach is used in some previous studies (e.g. [18]). The downside of normalization is the loss of information and influence of very large/small isolated value(s).

In the inference step, to compute dominance of one solution with respect to other (e.g. $\mathbf{x}^a$ over $\mathbf{x}^b$), the fuzzy membership values of m-objective differences $(f_i(\mathbf{x}^a) - f_i(\mathbf{x}^b), i = 1, 2, \cdots, m)$ are combined. Irrespective of the type of membership function, any suitable fuzzy set theoretic operation can be used for combination [26]. The most popular intersection set operation used in MaOPs is the product operation, also known as t-norm operation.

The following equations show the fuzzy inference and relative dominance computation procedure of two m-objective solutions, $\mathbf{x}^a$ and $\mathbf{x}^b$.

$$dom(\mathbf{x}^a, \mathbf{x}^b) = \prod_{i=1}^{m} \gamma_i(f_i(\mathbf{x}^a) - f_i(\mathbf{x}^b)) \tag{2}$$

$$dom(\mathbf{x}^b, \mathbf{x}^a) = \prod_{i=1}^{m} \gamma_i(f_i(\mathbf{x}^b) - f_i(\mathbf{x}^a)) \tag{3}$$

$$\phi(\mathbf{x}^a, \mathbf{x}^b) = \frac{dom(\mathbf{x}^a, \mathbf{x}^b)}{dom(\mathbf{x}^a, \mathbf{x}^b) + dom(\mathbf{x}^b, \mathbf{x}^a)} \tag{4}$$

$$\phi(\mathbf{x}^b, \mathbf{x}^a) = \frac{dom(\mathbf{x}^b, \mathbf{x}^a)}{dom(\mathbf{x}^a, \mathbf{x}^b) + dom(\mathbf{x}^b, \mathbf{x}^a)} \tag{5}$$

where $dom(\mathbf{x}^i, \mathbf{x}^j)$ is a scalar value and represents how much $\mathbf{x}^a$ dominates $\mathbf{x}^b$. The term $\phi(\mathbf{x}^a, \mathbf{x}^b)$ indicates the relative dominance of $\mathbf{x}^a$ to $\mathbf{x}^b$. We consider $\mathbf{x}^a$ dominates $\mathbf{x}^b$ iff $dom(\mathbf{x}^a, \mathbf{x}^b) > dom(\mathbf{x}^b, \mathbf{x}^a)$ and non-dominated iff $dom(\mathbf{x}^a, \mathbf{x}^b) = dom(\mathbf{x}^b, \mathbf{x}^a)$. It is evident from the above relations that if one solution is Pareto dominated then it will also be fuzzy dominated and if two solutions are fuzzy non-dominated they are Pareto non-dominated. However, it is very unlikely that two different solutions will have a same fuzzy dominance value (fuzzy non-dominated) even though they are Pareto non-dominated. Because of the different objective differences, fuzzy dominance is able to discriminate two solutions even when they are Pareto non-dominated. Therefore Pareto dominance can be considered as a special case of fuzzy dominance.

In defuzzification step, the dominance impact of one solution with respect to other solutions can be combined, ranked or used instead of Pareto dominance [18–22]. An important observation is that irrespective of a membership function and a set theoretic operation, we end up with only a scalar fuzzy value which does not tell anything about diversity. If the solutions are selected solely based on the fuzzy dominance values, the chosen solutions will be less diverse and will not able to cover the entire Pareto front of a given problem. This can be seen from the experimental evidences provided in a recent study [6]. Some existing approaches (e.g. [18], [22]) use a threshold parameter to divide solutions into different fronts but still fails to provide enough diversity in high-dimensional and non-regular Pareto [27].

2.2. Related Work

There are a very few fuzzy based EMO algorithms in the literature.

In [21], Farina and Amato proposed definitions of fuzzy-based optimality for multi-objective optimization. The main idea behind the given definitions is to introduce different degree of optimality. For each objective, the authors used a combination of three Gaussian functions as a non-adaptive membership function. The purpose of such a combined mechanism is to distinguish amongst better, equal and worse objective differences between two solutions. The proposed definitions have been applied on two simple multi-criteria decision making problems and two MOPs.

The work described in [20] studied the fuzzyfication of Pareto dominance relation and its application to the design of an EMO algorithm. In fuzzyfication, the authors used a non-symmetric membership function. Solutions those have high performance in one objective but poor in others will be preferred in the given fuzzy dominance definition. As no additional diversity measure was used in [20], the evolving population will lose diversity. To verify the usefulness of proposed EMO algorithm, an analytic study of the Pareto-Box problem was provided.

Nasir *et al.* [22] introduced a decomposition based fuzzy dominance algorithm (MOEA/DFD). For all objectives, the algorithm uses a general membership function of Ae^{-x} with uneven discrimination power. As the same membership function is used for all objectives, scaling issues are not addressed here. In comparing solutions, MOEA/DFD employed fuzzy dominance when a solution's dominance level is found greater than a particular threshold value. Otherwise, it used decomposition based weighted approach, which is able to handle diversity. The performance of MOEA/DFD was evaluated on twelve benchmark problems having two-objective to five-objective.

In [18, 19], the authors utilized fuzzy dominance concept to continuously differentiate individuals of a population into different degrees of optimality. They used the same left Gaussian function as a membership function for all objectives of a given MaOP. To handle the scaling issue, the objective differences are normalized by corresponding maximum objective difference value. The fuzzy concept was incorporated into NSGA-II and SPEA2 as a case study and termed them FD-NSGA-II and FD-SPEA2. In FD-NSGAII, a threshold value is used to divide solutions into different fronts based on the fuzzy dominance value. However, the solutions in the last front are chosen based on crowding distance. As fuzzy dominance is capable of discriminating solutions, crowding distance will rarely be used. Moreover, crowding distance has already found to be ineffective on MaOPs [1, 23, 28]. In FD-SPEA2, the best N solutions for next generation are taken based on fuzzy dominance values. The performance of the proposed algorithm was evaluated on seven DTLZ problems and two WFG problems having 5-, 10- and 20-objective.

3. Proposed Algorithm

In order to avoid the detrimental effect of uneven discrimination and objective normalization, F-DEA employs a separate membership function for each objective and determines its parameters adaptively. It also employs preferred reference points based clustering for ensuring diversity in its environmental selection. These features make the proposed algorithm different from others.

The algorithm starts with a randomly generated parent population P_t of N solutions and a set of generated/supplied reference points R^g/R^s. It then creates an offspring population Q_t of size N by applying crossover and mutation. Based on the positional information in the objective space, F-DEA finds p preferred points ($p \leq N$) and constructs clusters using the solutions as members and the preferred points as centers. It then adaptively constructs m fuzzy membership functions i.e., one function for each objective and utilizes them to compute dominance degrees of the solutions. Finally, F-DEA assigns fitness to the solutions and selects the best ones from different clusters in a round-robin fashion to from a new population P_{t+1} for the next generation. We summarize the steps of our method in Algorithm 1, which are explained further as follows.

3.1. Reference Point Generation

A number of existing studies [1, 13, 14, 23–25] employ reference points for assigning fitness values. However, F-DEA uses such points for clustering the solutions with a hope of maintaining both convergence and

Algorithm 1 Generation t of F-DEA

Input: R^g/R^s (generated reference points or supplied points), P_t (parent population of t-th generation), N (population size)

Output: P_{t+1}

$Q_t \leftarrow$ Recombination and Mutation on P_t
$C_t \leftarrow P_t \cup Q_t$
Select $p \leq N$ preferred reference points:
$R^p \leftarrow PreferredReferencePoint(C_t, R^g/R^s, p)$
Construct clusters using R^p:
$R^p(\{\mathbf{r}^j, X(\mathbf{r}^j)|1 \leq j \leq p\})$ % $\mathbf{r}^j$: j-th reference point in R^p, $X(\mathbf{r}^j)$: associated solution with $\mathbf{r}^j$
Construct m membership functions:
$(\gamma_1, \gamma_2 \cdots \gamma_m) \leftarrow AdaptiveMembershipFunction(C_t)$
Assign fitness to solutions within individual cluster:
$FitnessAssignment(R^p, C_t, \gamma)$
Sample solution from each cluster:
$P_{t+1} \leftarrow SamplingSelection(R^p, N)$

Algorithm 2 $PreferredReferencePoint(C_t, R^g/R^s, p)$

Input: R^g/R^s, C_t (combined population), p (maximum size of preferred reference points)

Output: $R^p(\{\mathbf{r}^j, X(\mathbf{r}^j)|1 \leq j \leq p\})$ (set of preferred reference points $\mathbf{r}^j$s with associated cluster of solutions $X(\mathbf{r}^j)s$)

for each solution $\mathbf{x} \in C_t$ **do**
  Normalize $f(\mathbf{x})$: $\tilde{f}(\mathbf{x}) = \tilde{f}_1(\mathbf{x}), \tilde{f}_2(\mathbf{x}), \ldots \tilde{f}_m(\mathbf{x})$
end for
$R^a = \{\emptyset\}$
for each solution $\mathbf{x} \in C_t$ **do**
  $\mathbf{r} = \mathbf{r^g}$: $\text{argmax}_{\mathbf{r^g} \in R^g/R^s}(S(\mathbf{r^g}, \tilde{\mathbf{f}}(\mathbf{x})))$
  if $\mathbf{r} \notin R^a$ **then**
    $X(\mathbf{r}) = \{\mathbf{x}\}$
    $R^a = R^a \bigcup \{\mathbf{r}\}$
  else
    $X(\mathbf{r}) = X(\mathbf{r}) \bigcup \{\mathbf{x}\}$
  end if
end for
if $|R^a| > p$ **then**
  Sort R^a: $MinMax(R^a)$
  $R^p =$ First p points of R^a
  $R^r = (R^a - R^p)$
  for each $\mathbf{r}^r \in R^r$ **do**
    for each solution $\mathbf{x} \in X(\mathbf{r}^r)$ **do**
      $\mathbf{r} = \mathbf{r}^p$: $\text{argmax}_{\mathbf{r}^p \in R^p}(S(\mathbf{r}^p, \tilde{\mathbf{f}}(\mathbf{x})))$
      $X(\mathbf{r}) = X(\mathbf{r}) \bigcup \{\mathbf{x}\}$
    end for
  end for
else
  $R^p = R^a$
end if

diversity. There are two important considerations for generating the reference points. Firstly, the reference points are to be generated in such a way so that they are uniformly distributed over the m-dimensional objective space. Secondly, the generation procedure has to be scaleable and computationally efficient. Keeping these considerations in mind, F-DEA uses the Das and Dennis's procedure [29] like others [1, 9, 23] for generating the reference points, which are distributed uniformly along the m-dimensional hyper-plane. The

Algorithm 3 $MinMax(R)$

Input: R (reference points)
Output: R^{st} (reference points sorted according to $MinMax$ distance)

$R^{st} = \emptyset$
$R^m =$ A set of m extreme points (one for each objective) chosen from R
$R^{st} = R^{st} \bigcup R^m$
$R' = R - R^{st}$
for each $\mathbf{r}$ in R' **do**
  Distance measure: $dist(\mathbf{r}) = \min\limits_{\forall \mathbf{r}^{st} \in R^{st}} d(\mathbf{r}, \mathbf{r}^{st})$
end for
while $R' \neq \emptyset$ **do**
  $\mathbf{r}^b = \mathbf{r} : \text{argmax}_{\mathbf{r} \in R'} \, dist(\mathbf{r})$
  $R^s = R^s \bigcup \{\mathbf{r}^b\}$
  $R' = R' - \{\mathbf{r}^b\}$
  for each $\mathbf{r}$ in R' **do**
    Update: $dist(\mathbf{r}) = \min\left(dist(\mathbf{r}), d(\mathbf{r}, \mathbf{r}^b)\right)$
  end for
end while

procedure generates a set of reference points, R^g, spanning the whole plane, each at $\delta = \frac{1}{\lambda}$ distance apart from the others. Analytically, $|R^g| = \binom{m+\lambda-1}{\lambda}$ reference points are generated where λ denotes the number of divisions in each objective-coordinate.

The main problem of this generation procedure or others [9] is the exponential increase of reference points with the increase of objectives. A simple technique to handle this issue is to increase the population size with respect to $|R^g|$. This in turn will increase evolution time. As an alternative, some studies (e.g. [1, 23]) generate a smaller number of reference points and impose this number as a constraint on the population size. F-DEA, on the other hand, first generates a large number reference points and then selects a set of preferred points based on the population size. This alleviates the problem of imposing constraint on the population size.

3.2. Preferred Reference Point

All the generated reference points may not be equally important with respect to the existing solutions of an evolving population. The proposed F-DEA thus selects a set of preferred reference points, R^p. It first finds the active reference points from R^g/R^s and then applies the *MinMax* procedure on them to choose p diverse points. A reference point is called an active reference point if it maintains some associations with one (or more) solution(s). The upper bound of p i.e., $|R^p|$ is N, the population size. The solutions of a population may reside in some (not all) regions of the m-dimensional objective space. In that case, p will be less than N. Our algorithm constructs p clusters using the preferred points as their centers and the solutions of C_t as their members. The procedure for selecting the preferred reference points and clusters is given in Algorithm 2.

3.2.1. Active Reference Point

As the values of the generated reference points lie in the range between 0 and 1, we adaptively normalize the objective values of the solutions to measure how close a solution is with respect to a particular reference point. Following the adaptive normalization procedure of [1], the i-th objective value, $f_i(\mathbf{x}^a)$, of any solution $\mathbf{x}^a$ can be normalized as

$$\tilde{f}_i(\mathbf{x}^a) = \frac{f_i(\mathbf{x}^a) - f_i^{min}(\mathbf{x}^u)}{\mathbf{z}^i - f_i^{min}(\mathbf{x}^u)} \quad \forall i \in 1, 2, \cdots, m \tag{6}$$

where $\tilde{f}_i(\mathbf{x}^a)$ is the i-th normalized objective value of the solution $\mathbf{x}^a$. The symbol $f_i^{min}(\mathbf{x}^u)$ refers to the minimum value of the i-th objective with respect to all solutions in C_t and the solution $\mathbf{x}^u$ has the minimum i-th objective value. The symbol $\mathbf{z}_i$ refers to the intercept computed from the i-th objective axis and a m-dimensional linear hyper-plane. The hyper-plane is constituted from m solutions, where a solution $\mathbf{x}^h \in C_t$ makes the following achievement scalarizing function (ASF) minimum for an objective direction $\mathbf{w}_i$.

$$ASF(\mathbf{x}^h, \mathbf{w}_i) = \max_{i=1}^{m} \frac{f(\mathbf{x}^h) - f^{min}(\mathbf{x}^u)}{\mathbf{w}_i} \quad \mathbf{x}^h \in C_t \tag{7}$$

Normalization using intercepts helps solutions to expand it's objective space.

The closeness of solutions indicates their associations with the reference points. We utilize cosine similarity measure for this purpose. The cosine similarity measure, $S(\mathbf{r^j}, \tilde{\mathbf{f}}(\mathbf{x}^a))$, between the normalized fitness vector $\tilde{\mathbf{f}}(\mathbf{x}^a) = \tilde{f}_1(\mathbf{x}^a), \tilde{f}_2(\mathbf{x}^a), \ldots \tilde{f}_m(\mathbf{x}^a)$ of any solution $\mathbf{x}^a$ and the m-dimensional j-th reference point $\mathbf{r}^j$ can be computed as

$$S(\mathbf{r^j}, \tilde{\mathbf{f}}(\mathbf{x}^a)) = \frac{\tilde{\mathbf{f}}(\mathbf{x}^a).\mathbf{r}^j}{|\tilde{\mathbf{f}}(\mathbf{x}^a)||\mathbf{r}^j|} \quad \forall j \in 1, 2, \cdots, |R^g| \tag{8}$$

As we generate a set of reference points, R^g, there will be $|R^g|$ similarity measures for $\mathbf{x}^a$. The largest similarity measure obtained for any of the reference points is considered as the active reference point that maintains the highest association with $\mathbf{x}^a$. While associating the solutions with the reference points, it is possible that some points may associate more than one solutions, some may associate one solution or some may associate no solution at all.

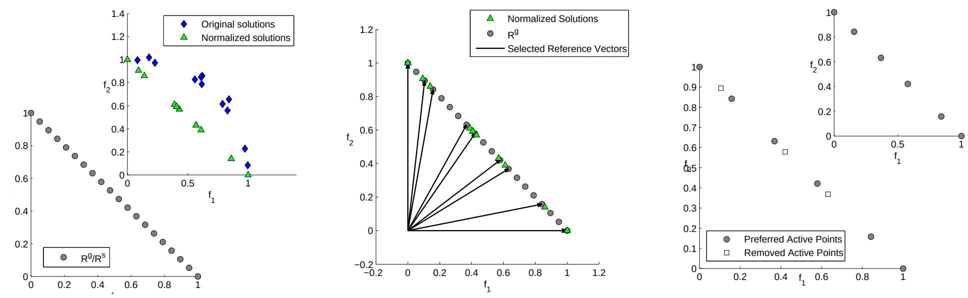
3.2.2. MinMax Measure

The objective space size increases exponentially with an increasing number of objectives. It would be beneficial if we can cover the whole objective space by a reasonable number of active reference points in the least crowded manner. We devise a procedure based on the *MinMax* measure to select p diverse active reference points, which F-DEA employs as the clusters' centers, from the crowded ones (Algorithm 3).

The procedure first selects m number of most extreme active reference points i.e., one for each objective and then incrementally selects the other points. To select the other points, it calculates the minimum Euclidean distance between the already selected points and the remaining ones. The point with the maximum distance is selected as the next point and the minimum distance of the remaining points are updated based on the selected point. This procedure continues until all the points are selected. In this way, we finally obtain a sorted list of active reference points that are far apart from each other i.e., diverse. This procedure will be effective for disconnected, degenerated and other irregular shaped geometry because we are using the reference points only where solutions exist. Thus we are effectively maintaining cluster uniformity using the same number of reference points as other approaches but we are using them where it is needed. It may possible to devise other techniques as well, which can be a future research topic. Fig. 2 demonstrates the process of selecting the preferred reference points from a set of generated reference points.

3.3. Clustering

The proposed algorithm constructs clusters where the preferred reference points and solutions (data points) are used as the clusters' centers and members, respectively. An important question may arise why we do not use a classical approach for clustering. The classical approach usually partition a set of data points into a set of meaningful sub-classes with an aim of understanding the natural grouping/structure among them. This is why the clusters' centers are determined from the existing data points. On the other hand,

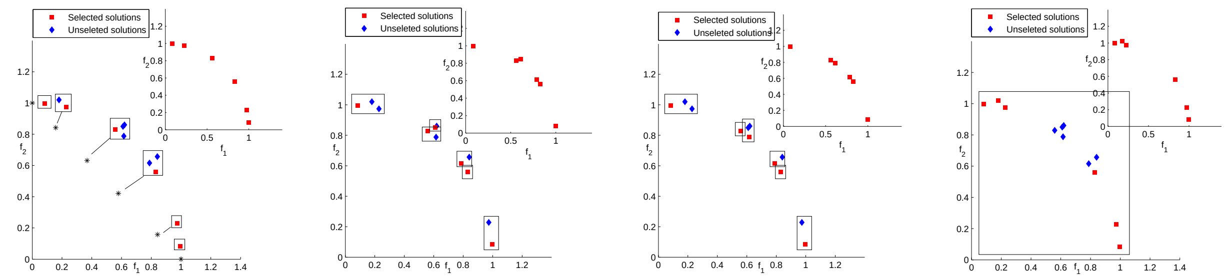


(a) $|R^g| = 20$, Original Solutions $|C_t| = 12$

(b) Active reference points $|R^a| = 9$ (marked with vector line)

(c) Selection of a number of preferred reference points ($p = 6$)

Figure 2: Process of selecting the preferred reference points from a set of generated reference points



(a) Preferred Reference points based clustering

(b) k-means clustering with Euclidean distance measure

(c) k-means clustering with cosine similarity measure

(d) No clustering i.e., all solutions in one cluster

Figure 3: Solutions are grouped (rectangle box) by different clustering mechanisms. The red squared solutions are the selected solutions obtained by applying fuzzy fitness based environmental selection procedure within each cluster.

the aim of our clustering is to facilitate diversity in selecting solutions for the next generation. We thus use a set of preferred points, which are diverse and selected from a set of uniformly generated reference points, as the clusters' centers. Although the generation of uniform reference points may suffer in case of an irregular Pareto front, the use of the preferred reference points as the clusters' centers alleviates this problem to some extent.

Fig. 3 shows the effect of preferred reference points based clustering and traditional k-means clustering. It is clear from the figure that k-means clustering with Euclidean distance measure won't work as it considers locally crowded regions to form clusters. Thus, the solutions that are Pareto dominated might be grouped together in the objective space. Although k-means clustering with cosine measure handles the aforementioned issue, it does not consider and maintain uniformity in clusters. In contrast, the reference points based clustering we employ in F-DEA maintains clusters uniformity by employing a uniformly distributed reference points as the clusters' centers.

Some previous non-fuzzy EMO algorithms [1, 8, 23] also employ clustering in solving MaOPs. The clustering procedure of F-DEA differs from the one used in [1] with two notable exceptions. Firstly, F-DEA employs preferred reference points based clustering to provide better cluster uniformity, remove dependency on population size and handle irregular shaped Pareto fronts. Secondly, it uses cosine similarity instead of Euclidean distance to find the solutions' association with the generated reference points. It is suitable because angle between a reference point and a candidate solution remains constant irrespective of exact distance from the ideal point, which handles the scaling issue of the generated reference points. The θ-dominance algorithm [23] also uses reference points based clustering similar to [1] with an exception in normalization procedure. CEGA [8], a clustering based elitist genetic algorithm, utilizes a bottom up hierarchical approach in the decision variable space not in the objective space.

3.4. Adaptive Membership Function

In this work, we for the first time use the Sigmoid membership function (Fig. 4) into a fuzzy based EMO algorithm. This function is not only monotonically decreasing but also anti-symmetric at mean. We set its growth parameter i.e., α in such a way so that it handles the extreme values to some extent and work in support of our clustering approach. The Sigmoid membership function for the i-th objective can be defined as

$$\gamma_i(f_i(\mathbf{x}^a) - f_i(\mathbf{x}^b)) = \frac{1}{1 + e^{-\alpha_i((f_i(\mathbf{x}^a) - f_i(\mathbf{x}^b)))}} \tag{9}$$

where mean is set to zero for making the membership function anti-symmetric at that value.

An MOP may or may not have an identical range of values for each objective. To capture this notion, it is better to determine α_i adaptively. We calculate it for each objective at every generation of evolution. The mean objective difference (μ_i) and mean objective variance (σ_i^2) are used to compute α_i. We obtain μ_i and σ_i^2 using absolute objective differences of all pairs of solutions. The growth parameter α_i has been defined in such a manner so that the membership value obtained from Eq. (9) is 0.99 at the point $(-\mu_i - \sigma_i)$. We compute α_i as

$$\alpha_i = \frac{\ln p - \ln(1-p)}{q_i} \tag{10}$$

where $p = 0.99$ and $q_i = -\mu_i - \sigma_i$. The way we define α_i is advantageous in three aspects.

- Firstly, as α_i is defined based on mean and variance instead of maximum or minimum objective difference, inappropriate normalization effect of a significantly large/small objective difference mentioned in Section 2 will be minimized to some degree.
- Secondly, if the objectives are in different scales, the corresponding mean and variance of the objective differences will be different which in turn will produce different α_is i.e., different membership functions.
- Thirdly, as we cluster the solutions, all objective differences in the same cluster will be small. It ensures that most of the objective differences will lie in the range between $-\mu_i - \sigma_i$ and $\mu_i + \sigma_i$. The shape of membership function indicates that the solutions are evenly discriminable within this range.

In short, the anti-symmetric property and α_i's definition resolve the issues of uneven discrimination ability, objective difference normalization and objective scaling.

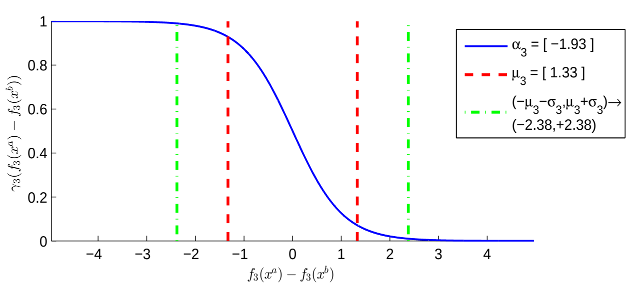


Figure 4: Sigmoid Membership Function. This particular case shows the position of mean objective difference value (μ_3) for the third objective of WFG2 problem obtained from the 250^{th} generation of a particular seed.

3.5. Fuzzy Dominance and Fitness Assignment

The essence of our fuzzy dominance computation and fitness assignment is that they are local. We utilize the solutions in the same cluster for computing their dominance degrees and assigning their fitness scores. An advantage of this approach is that there is an opportunity to employ parallelism for such computation and assignment. We employ the membership function represented by Eq. (9) to compute membership values which in turn are used for obtaining fuzzy dominance among all pairs of solutions in each cluster. We use Eqs. (2) and (3) for obtaining fuzzy dominance.

For assigning fitness to a solution of any cluster, we first compute the relative dominance degrees, ϕs, using Eqs. (4) and (5). Note that the relative dominance degree of any solution $\mathbf{x}^a$ in a particular cluster is considered only with respect to all other solutions in that cluster. We then add all these degrees and assign it as the solution's fitness as represented by Eq. (11).

$$fit(\mathbf{x}^a) = \sum_{b=1, b\neq a}^{b=n} \phi(\mathbf{x}^b, \mathbf{x}^a) \tag{11}$$

Here n is the number of solutions in any cluster. If a solution is least dominated by other solutions of the same cluster, then its fitness value will be smallest and it will be selected for the next generation.

3.6. Environmental Selection

As mentioned before, F-DEA constructs clusters using the evolved solutions as their members and the preferred reference points as their centers. The aim of our environmental selection procedure is to choose solutions for the next generation by considering not only their convergence but also their diversity. To achieve these goals, F-DEA first sorts the clusters by applying the *MinMax* procedure based on the clusters' centers. This is done to give priority on the distant clusters. The algorithm then selects the solutions from the sorted clusters in a round robin fashion. For a minimization problem, a solution having a minimum dominance score compared to the remaining ones of the same cluster is considered for selection. It means the selected solution is least dominated by the remaining ones of the same cluster. Giving preference to the distant clusters and selecting the least dominated solutions from them indicates F-DEA's emphasis on both diversity and convergence in its environmental selection.

To visualize the essence of our reference points based clustering, Fig. 3 shows the effect of reference points based clustering, no clustering, k-means clustering with Euclidean distance and k-means clustering with cosine similarity measure. It is evident from this figure that reference points based clustering is able to maintain well both convergence and diversity in selecting solutions for the next generation. In contrast, when no clustering is employed, fuzzy dominance ranks all solutions based on their scalar values and the corner solutions are selected in concave surface due to bias introduced by fuzzy dominance (Fig. 3(d)). Diversity maintenance of k-means clustering with Euclidean distance (Fig. 3(b)) is better than no clustering (Fig. 3(d)) but worse than k-means clustering with cosine similarity measure (Fig. 3(c)). To get a further insight of how much contribution we get from reference points based clustering and fuzzy dominance over Pareto dominance, detail experimentation has been conducted and presented in Section 4.8.

3.7. Computational Complexity

The basic F-DEA algorithm contains seven lines (Algorithm 1). As the first two lines are common in an evolutionary algorithm, the remaining five lines that call five procedures actually determine F-DEA's complexity. Algorithm 2 finds p preferred reference points from $|R^g/R^s|$ generated/supplied points and constructs clusters using such points. The overall computation including adaptive normalization (O(mN)) can be performed with $max(O(mN\,|R^g|\,/\,|R^s|\,, O(mN^2))$ operations. The construction of adaptive membership function using mean μ and variance σ^2 requires $\mathrm{O}(mN^2)$ operations.To calculate μ and σ^2 in a single pass, we use the Knuth's running-mean-variance approximation process [30]. Let the maximum number of solutions in a cluster is χ. As there can be p different clusters, the time complexity of fitness assignment is at most $O(mp\chi^2)$, including the selection of best solutions that requires $max(O(mp^2), O(p\chi))$ or $O(mN^2)$ in the worst case. Thus the overall time complexity of F-DEA is $max(O(mN\,|R^g/R^s|), O(mN^2))$.

4. Experimental Studies

We perform a series of experiments to investigate and compare the optimization capability of our algorithm. Two well known benchmark test suites WFG [31] and DTLZ [32] are utilized for this purpose. The WFG problems are truly non-linear, non-separable and multimodal, and they do not have an identical range of values for each objective [31]. These characteristics make the WFG problems more challenging than DTLZ ones. To investigate the performance on degenerate problems, we also include the Rectangle problem [27] for experimentation.

4.1. Benchmark Problems

The WFG test suite contains nine problems and we choose eight of them for experimentation. We exclude the WFG3 problem because it has a non-degenerate part [33] which might create erroneous result during performance evaluation. To reduce experimentation burden, we choose three (DTLZ1, DTLZ3 and DTLZ7) out of seven problems of From the DTLZ suite. We omit the DTLZ2, DTLZ4 problems from comparative studies as they are relatively easy problems and the DTLZ5, DTLZ6 problems due to their ambiguity in Pareto fronts beyond 3-objective [31].

The problems in WFG and DTLZ test suites can be scaled to any number of objectives and decision variables. We consider the number of objectives $m \in \{2, 3, 5, 7, 10, 12, 15, 20, 25\}$. As per recommendation from the WFG Toolkit[1], we set the distance related parameter $l = 20$, the position related parameter $k = 4$ for $m = 2$, $k = 2 \times (m - 1)$ for $3 \leq m \leq 10$, and $k = (m - 1)$ for $m > 10$. The number of decision variables, n, is set equal to $l + k$. We also follow the suggestions of [1, 32] in setting n and k of DTLZ problems. We set n equal to $m + k - 1$ for all DTLZ problems we consider in this work. We set $k = 5$ for DTLZ1, $k = 10$ for DTLZ3 and $k = 20$ for DTLZ7.

4.2. Performance metrics

The performance of any evolutionary algorithm for an MOP is usually measured from two aspects: convergence and diversity. Inverse generalization distance (IGD) [34] and hypervolume (HV) [15] are two performance metrics that capture in one scalar both convergence and diversity. To calculate IGD, a well uniform sample set of true Pareto front is required. It is, however, challenging to get such a set for an increasing number of objectives [17].

The other performance metric HV has nicer mathematical properties and is the only quality measure known to be strictly Pareto-compliant [34]. These good features make HV a fair indicator for comparing different algorithms. For a non-dominated solution set A obtained in final generation by an algorithm, HV is calculated with respect to a reference point $\mathbf{r}$. It has been known that choosing $\mathbf{r}$ slightly larger than the nadir point, $\mathbf{z}^{nad}$, is suitable [2]. In our experiments, we set $\mathbf{r}$ to $1.1\mathbf{z}^{nad}$, which can be analytically obtained or approximated [31]. Following [35], the points which do not dominate $\mathbf{r}$ are discarded in computing HV. We use exact HV calculation for problems with objective less than 5 and the Monte Carlo based fast HV approximation algorithm [16] with $10,00,000$ sampling points for others.

[1] http://www.wfg.csse.uwa.edu.au/toolkit/README.txt

Table 1: Population Size N, number of divisions λ used in NSGAII [36] and F-DEA, number of weight vectors Z used in MOEA/D [9] and goals in PICEAg [14] for different objectives

Obj. No. (m)	N used in all algorithms	λ used in NSGAIII	λ used in F-DEA	Z used in MOEA/D	G used in PICEAg
2	204	200	2000	20400	200
3	204	18	100	20400	300
5	212	6	30	21200	500
7	212	4	13	21200	700
10	220	3	9	22000	1000
12	160	2, 2	7	16000	1200
15	240	2, 2	6	24000	1500
20	212	2	5	21200	2000
25	328	2	4	32800	2500

In experimental studies, we have used HV, IGD and visualization figures to evaluate performances of the algorithms.

4.3. Other Algorithms in Comparison

There exists a few fuzzy dominance based EMO algorithms in the literature. FD-NSGAII [18] is one such algorithm, which has been found better than other similar algorithms. We thus select FD-NSGAII for comparison. We also choose NSGAIII [1] for comparison because it exhibits superior performance compared to several well-known algorithms. Decomposition based algorithm MOEA/D [9] with weighted Tchebycheff approach has been chosen as a representative of decomposition, aggregation and reference weights/points based approach. We choose a variant of SDE [6], SPEA2+SDE, for comparison. This particular variant shows the best overall performance among its other variants [6]. Preference based co-evolutionary algorithm (PICEAg) [14] and HypE [16], an indicator based algorithm, are also chosen for comparison.

4.4. Parameter Setting

The population size N of NSGAIII cannot be arbitrarily specified, rather it has to be set equal to the number of reference points. The procedure employed for generating such points uses a division parameter λ that determines this number. Although our algorithm uses reference points, it does not put any constraint in choosing N. Table 1 shows λ of NSGAIII and F-DEA, the weights Z of MOEA/D and the number of goals G for PICEAg. To make a fair comparison, the population size is set same for all competing algorithms.

All competing algorithms employ simulated binary crossover and polynomial mutation for generating offspring. The crossover and mutation probabilities are set to 1 and $1/n$, respectively. We also use the same mutation distribution index i.e., 20 for these algorithms. The crossover distribution index is set to 30 for F-DEA and NSGAIII, 20 for SDE, and 15 for HypE, MOEA/D, FD-NSGAII, and PICEAg. Beside the general and common parameters, there are some specific parameters for competing algorithms. The parameter p of F-DEA is set equal to N. The bound of reference point and the number of sampling points used in HypE [16] has been set to 200 and $10,000$, respectively. In MOEA/D [9, 10], the neighborhood size, T, is chosen 5% of the population and the maximum number of population slots, η_r, has been chosen 1% of T. The fuzzy ranking threshold parameter β of FD-NSGAII [18] has been set to 0.50. In SDE [6], the archive is set equal to population size, N. The number of goals G used in PICEAg [14] is set to $m \times 100$.

Each algorithm is run independently with 20 different seeds for each problem instance. We set the termination criterion to 250 generations for each run. The Wilcoxon rank-sum test [37] with a 5%-significance level is used while comparing two algorithms on any problem instance over 20 runs. For reducing Type-I error in pairwise testing, Šidák corrections [38] are also employed. F-DEA has been implemented by the authors in *JMetal*[2] framework and it's source code is available online[3]. HypE and MOEA/D from MOEA framework[4]. We use a C++ implementation[5] for NSGAIII. FD-NSGAII has been implemented by the authors from [18] in *JMetal* framework. The source codes of SDE and PICEAg algorithms are received from the authors of the respective algorithms. The true Pareto front of the WFG problems is generated using the *WFG-Toolkit* [6]. The uniform Pareto front of DTLZ1, DTLZ3 is generated from NSGAIII implementation tools[5]. All algorithms were run on Intel 2.40GHz core i5 processor with 4GB RAM. The performance evaluation and visualization codes are shared in *Github* [7].

4.5. Experiment on WFG Problems

This section presents evaluation and comparison of our F-DEA and six other algorithms on eight WFG problems with 2-, 3-, 5-, 7-, 10-, 12-, 15-, 20- and 25-objective. Table 2 shows the average HV with standard deviation and ranks of different algorithms obtained by the Wilcoxon rank-sum test based on HVs of 20 independent runs. The lower a rank is, the better an algorithm is.

4.5.1. WFG1 Problem

The WFG1 problem has most transformation functions among the other problems of the same suite. Hence, it is difficult to maintain diversity with sufficient convergence for this problem F-DEA exhibited the best performance among all competing algorithms from 2-objective to 25-objective.

4.5.2. WFG2 Problem

NSGAIII was the top performer in 2-objective, while PICEAg and SDE were the top two performers from 3-objective to 25-objective. F-DEA secured overall the third position and showed very competitive performance with respect to average HVs achieved by the top performers.

4.5.3. WFG4 Problem

For a smaller number of objectives, SDE and PICEAg were the two top performing algorithms while F-DEA showed competitive performance. F-DEA, however, tied with SDE in 7-objective and outperformed all other competing algorithms from 10-objective to 25-objective.

4.5.4. WFG5 Problem

An important aspect of this problem is its deceptive nature, which challenges the ability of an algorithm to find good quality solutions. F-DEA handled the challenge successfully and outperformed all competing algorithms from 15-objective to 25-objective.

4.5.5. WFG6 Problem

F-DEA was the top performer from 12-objective to 25-objective of the non-separable reduced problem, WFG6. The proposed algorithm shared the top position with SDE and PICEAg for 2-objective and with PICEAg for 10-objective.

---

[2]www.jmetal.org
[3]https://github.com/siddhartha047/FDEA
[4]www.moeaframework.org
[5]http://web.ntnu.edu.tw/ tcchiang/publications/nsga3cpp/nsga3cpp.htm
[6]http://www.wfg.csse.uwa.edu.au/toolkit/
[7]https://github.com/siddhartha047/MOEAevaluation-plot

4.5.6. WFG7 Problem

This is a separable unimodal problem. F-DEA showed better performance with an increasing number of objectives. For example, it outperformed all other algorithms from 15-objective to 25-objective.

4.5.7. WFG8 Problem

SDE, PICEAg and F-DEA were the top performing algorithms with close average HV values for this non-separable problem. SDE was the top performer for 2-, 3- and 20-objective, PICEAg for 5- and 10-objective, and F-DEA for 15- and 25-objective.

4.5.8. WFG9 Problem

This is a non-separable deceptive problem for which NSGAIII and SDE were two top performers for 2-, 3- and 5-objective. As the number objective increases, the performance of F-DEA enhanced and shared the top position with SDE, PICEAg for the 7- and 10-objective. However, F-DEA outperformed all other algorithms from 12-objective to 25-objective.

Table 3 summarizes the obtained results of different algorithms with respect to the number of objectives. We count the number of times F-DEA is better, worse or equal than any competing algorithm based on the Wilcoxon rank-sum test. For better understanding, we present the results into three groups: 2-objective to 3-objective, 5-objective to 7-objective and 10-objective to 25-objective. For 2-objective and 3-objective, F-DEA was outperformed by SDE and PICEAg, but it was found better than MOEA/D, HypE, FD-NSGAII and NSGAIII. SDE and PICEAg were also the top performing algorithms for 5-objective to 7-objective, while F-DEA secured the overall third position and outperformed others. This scenario is totally different for a higher number of objectives i.e., 10-objective to 25-objective for which F-DEA was found better than all competing algorithms.

To investigate the performances of different algorithms visually, we present the attainment surface plots and parallel coordinate plots of competing algorithms on 15- objective of WFG9 problem. For brevity, we present only the parallel coordinate plots of five competing algorithms. We use the non-dominated solutions obtained in the final generation for obtaining plots. In terms of HV, the solutions were taken from the 10th run of the sorted 20 independent runs. To visualize the achieved convergence and diversity more clearly, the obtained non-dominated solutions of a particular algorithm are divided into two categories: converged and non-converged. The solutions that are at most d-distance apart from the normalized surface are considered as converged and remaining are considered as non-converged.

Fig. 5 shows the parallel coordinate plots of the five algorithms for the WFG9 problem with 15-objective. In terms of the number of converged solutions, FD-NSGAII 5(c), F-DEA 5(a) and PICEAg 5(e) secured the first, second and third positions, respectively. However, only F-DEA was able to maintain good diversity which could be attributed to the variations in the objective values of the solutions. While FD-NSGAII failed miserably to maintain diversity, PICEAg was able to maintain diversity moderately. Although most of the solutions from SDE (Fig. 5(d)), NSGAIII (Fig. 5(b)) were not converged but they maintained better diversity than FD-NSGAII and PICEAg. The bottom figures of five competing algorithms show the zoomed version of all the obtained non-dominated solutions. It is clear from these figures that F-DEA is better in terms of simultaneous minimization of all the objectives while maintaining diversity.

4.6. Experiments on DTLZ Problems

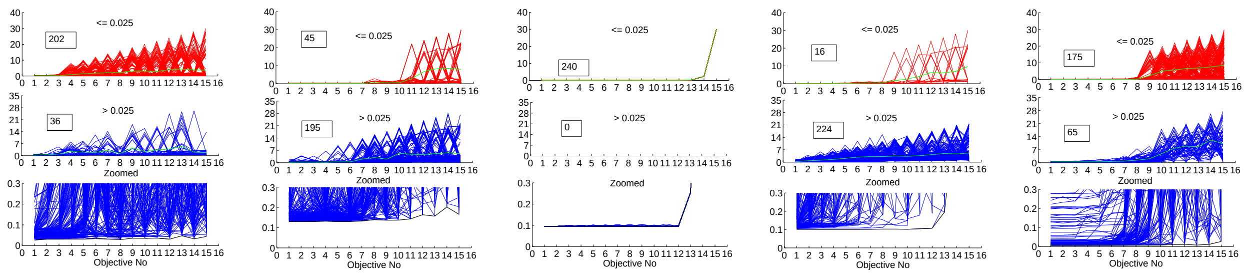
We apply F-DEA and other competing algorithms on three DTLZ problems, DTLZ1, DTLZ3 and DTLZ7. These problems are complex compared to other DTLZ problems. DTLZ1 is difficult to converge, DTLZ3 has a large number of local fronts and DTLZ7 is a disconnected problem. We use HV and IGD for comparison.

Table 2: Average HV of different algorithms on eight WFG problems over 20 independent runs. The best result based on the Wilcoxon rank sum test with a significance level of 0.05 is marked in bold-face. The rank of a particular algorithm is shown in bracket.

Prob.	m	HYPE	MOEA/D	FD-NSGAII	SDE	NSGAIII	PICEA-g	F-DEA
wfg1	2	0.0970±0.0257(6)	0.1467±0.0264(5)	0.0495±0.0214(7)	0.2324±0.0434(4)	**0.3333±0.0689(1.5)**	0.3233±0.0037(3)	**0.3438±0.0371(1.5)**
	3	0.2968±0.0248(7)	0.4042±0.0485(5)	0.3959±0.0226(5)	0.5614±0.0497(2.5)	0.4094±0.0396(5)	0.5689±0.0052(2.5)	**0.6120±0.0413(1)**
	5	0.3515±0.0347(7)	0.5187±0.0246(5)	0.5979±0.0474(3)	0.5622±0.0232(4)	0.4065±0.0732(6)	0.7117±0.0114(2)	**0.7516±0.0337(1)**
	7	0.4164±0.0279(7)	0.5985±0.0402(5)	0.8524±0.0531(2.5)	0.7058±0.0256(4)	0.4824±0.0275(6)	0.8429±0.0297(2.5)	**0.9058±0.0560(1)**
	10	0.6005±0.0225(6)	0.8432±0.0692(5)	1.2047±0.0754(2)	0.9937±0.0418(4)	0.3671±0.0924(7)	1.0310±0.0307(3)	**1.4206±0.0684(1)**
	12	0.6543±0.0292(6)	0.8989±0.0669(5)	1.3867±0.1012(2)	1.1189±0.0517(4)	0.2839±0.0633(7)	1.2233±0.0310(3)	**1.6017±0.1296(1)**
	15	0.8732±0.0501(6)	1.2230±0.0545(5)	2.1247±0.1903(2)	1.6596±0.0671(3)	0.4128±0.1145(7)	1.5930±0.0473(4)	**2.3314±0.1024(1)**
	20	1.2481±0.0439(6)	1.6443±0.1007(5)	**3.4625±0.3626(1.5)**	2.7029±0.1280(3)	0.3940±0.1587(7)	2.5527±0.0635(4)	**3.5738±0.2874(1.5)**
	25	2.1095±0.1037(6)	2.4165±0.1568(5)	6.6621±0.3718(2)	2.8201±0.0539(4)	1.8261±0.2503(7)	4.0829±0.0980(3)	**7.1479±0.4528(1)**
wfg2	2	0.6806±0.0149(5.5)	0.6820±0.0183(5.5)	0.3459±0.0572(7)	0.7407±0.0054(2)	**0.7452±0.0087(1)**	0.7406±0.0091(3)	0.7363±0.0064(4)
	3	0.9702±0.0767(5.5)	1.0575±0.0972(5.5)	0.3862±0.0656(7)	1.2207±0.0581(2)	1.1573±0.0929(3.5)	**1.2450±0.0022(1)**	1.2119±0.0418(3.5)
	5	1.3316±0.1315(4.5)	1.3283±0.1175(6)	0.4409±0.0009(7)	1.5401±0.0840(2)	1.3872±0.1229(4.5)	**1.5546±0.1049(1)**	1.5277±0.0838(3)
	7	1.6032±0.1593(5.5)	1.4962±0.1489(5.5)	0.5498±0.0678(7)	1.8686±0.1022(2)	1.6951±0.1574(4)	**1.8875±0.1285(1)**	1.8347±0.1241(3)
	10	2.2659±0.1715(4.5)	2.1991±0.1021(6)	0.5655±0.2030(7)	2.5531±0.0064(2)	2.3060±0.0507(4.5)	**2.5889±0.0018(1)**	2.5280±0.0097(3)
	12	2.6969±0.0885(4.5)	2.5541±0.1337(6)	0.6473±0.2821(7)	3.0745±0.0166(2)	2.6757±0.0975(4.5)	**3.1266±0.0034(1)**	3.0375±0.0228(3)
	15	3.7255±0.1172(4)	3.5520±0.1362(5.5)	0.8039±0.3350(7)	4.1263±0.0084(2)	3.5388±0.0700(5.5)	**4.1692±0.0022(1)**	4.0798±0.0196(3)
	20	5.7851±0.4480(4.5)	5.6843±0.2581(4.5)	1.2816±0.6254(7)	6.6467±0.0169(2)	4.0451±0.7806(6)	**6.7064±0.0094(1)**	6.5645±0.0299(3)
	25	9.6551±0.3095(4.5)	9.1922±0.3394(6)	2.2394±0.9957(7)	10.6314±0.0470(2.5)	8.9972±1.8322(4.5)	**10.8188±0.0074(1)**	10.6293±0.0517(2.5)
wfg4	2	0.3946±0.0041(6)	0.4058±0.0044(5)	0.1278±0.0377(7)	**0.4184±0.0010(1.5)**	**0.4182±0.0010(1.5)**	0.4143±0.0014(4)	0.4172±0.0006(3)
	3	0.6447±0.0203(6)	0.6697±0.0115(5)	0.1220±0.0053(7)	**0.7501±0.0028(1)**	0.7096±0.0038(4)	0.7432±0.0021(2)	0.7321±0.0024(3)
	5	0.9151±0.0969(6)	1.0343±0.0251(5)	0.2472±0.0915(7)	1.2164±0.0083(2.5)	1.0503±0.0134(4)	**1.2475±0.0262(1)**	1.2220±0.0083(2.5)
	7	1.1010±0.1397(6)	1.3328±0.0423(4)	0.3230±0.1273(7)	**1.6016±0.0149(1.5)**	1.3015±0.0559(4)	1.3295±0.0982(4)	**1.6127±0.0221(1.5)**
	10	1.5518±0.1823(6)	2.0844±0.0399(4.5)	0.4548±0.1907(7)	2.3121±0.0292(2)	2.0664±0.0923(4.5)	2.1635±0.0753(3)	**2.3892±0.0111(1)**
	12	1.6225±0.2620(6)	2.4331±0.1179(3)	0.4192±0.1924(7)	2.7870±0.0416(2)	1.9808±0.0874(5)	2.3082±0.1245(4)	**2.8607±0.0273(1)**
	15	2.3364±0.3730(6)	3.4223±0.0955(3.5)	0.6341±0.2520(7)	3.7551±0.0348(2)	2.7353±0.1271(5)	3.3657±0.1431(3.5)	**3.9442±0.0311(1)**
	20	3.6517±0.3742(6)	5.2892±0.1486(3.5)	0.7661±0.2789(7)	5.9939±0.2018(2)	4.0528±0.2925(5)	5.1166±0.3164(3.5)	**6.3855±0.0460(1)**
	25	7.4075±0.6866(6)	8.8991±0.1289(3.5)	1.1546±0.1966(7)	9.4890±0.4354(2)	8.2805±0.4261(5)	8.7945±0.5704(3.5)	**10.5396±0.0492(1)**
wfg5	2	0.3243±0.0122(6)	0.3603±0.0068(5)	0.0861±0.0202(7)	**0.3743±0.0005(1.5)**	0.3740±0.0006(3.5)	**0.3741±0.0009(1.5)**	0.3737±0.0008(3.5)
	3	0.5445±0.0262(6)	0.6056±0.0145(5)	0.1142±0.0305(7)	**0.7010±0.0035(1)**	0.6648±0.0056(4)	0.6966±0.0033(2)	0.6882±0.0022(3)
	5	0.8947±0.0602(6)	0.9683±0.0248(5)	0.1278±0.0006(7)	1.1715±0.0074(3)	0.9993±0.0119(4)	**1.1854±0.0038(1)**	1.1800±0.0073(2)
	7	1.0418±0.1275(6)	1.2624±0.0405(4.5)	0.1554±0.0003(7)	1.5562±0.0097(3)	1.2482±0.0517(4.5)	**1.5814±0.0173(1.5)**	**1.5809±0.0085(1.5)**
	10	1.4758±0.1146(6)	1.9581±0.0466(4)	0.2064±0.0005(7)	2.2843±0.0079(2)	1.8706±0.0613(5)	2.1577±0.0755(3)	**2.2964±0.0062(1)**
	12	1.6040±0.1641(6)	2.2598±0.0914(4)	0.2487±0.0005(7)	**2.7525±0.0127(1.5)**	2.0394±0.1186(5)	2.3688±0.0823(3)	**2.7670±0.0236(1.5)**
	15	2.4435±0.1978(6)	3.2303±0.0904(4)	0.3316±0.0016(7)	3.6807±0.0185(2)	2.7944±0.1447(5)	3.3723±0.0774(3)	**3.7649±0.0091(1)**
	20	3.6012±0.4559(6)	5.0711±0.2051(3.5)	0.5312±0.0011(7)	5.9066±0.0244(2)	4.5393±0.3738(5)	5.1737±0.1697(3.5)	**6.0356±0.0172(1)**
	25	6.5479±0.7306(6)	8.3672±0.3236(4.5)	0.8543±0.0013(7)	8.6133±0.2848(4.5)	8.9789±0.2287(2.5)	8.8711±0.1808(2.5)	**9.7850±0.0161(1)**
wfg6	2	0.2660±0.0187(6)	0.3587±0.0115(5)	0.0864±0.0027(7)	**0.3845±0.0040(2)**	0.3735±0.0127(4)	**0.3836±0.0041(2)**	**0.3817±0.0045(2)**
	3	0.3448±0.0424(6)	0.6295±0.0123(5)	0.1070±0.0052(7)	**0.7094±0.0039(1)**	0.6736±0.0066(4)	0.7039±0.0069(2)	0.6874±0.0048(3)
	5	0.5599±0.0876(6)	0.9626±0.0238(5)	0.1341±0.0026(7)	1.1874±0.0072(2)	1.0276±0.0183(4)	**1.2105±0.0074(1)**	1.1804±0.0129(3)
	7	0.6954±0.1325(6)	1.2821±0.0561(4.5)	0.1632±0.0021(7)	1.6131±0.0125(2)	1.2492±0.0339(4.5)	**1.6319±0.0137(1)**	1.5948±0.0155(3)
	10	1.2015±0.1278(6)	1.9249±0.0507(4)	0.2175±0.0022(7)	2.2412±0.0276(3)	1.8219±0.0262(5)	**2.2302±0.0846(1.5)**	**2.2819±0.0258(1.5)**
	12	1.3825±0.1510(6)	2.1664±0.1233(4)	0.2638±0.0023(7)	2.6691±0.0246(2)	2.0114±0.1159(5)	2.5231±0.1406(3)	**2.7368±0.0532(1)**
	15	2.1409±0.2052(6)	3.1680±0.1164(4)	0.3519±0.0029(7)	3.5523±0.0467(2.5)	2.7916±0.1132(5)	3.5061±0.1279(2.5)	**3.7578±0.0354(1)**
	20	3.0136±0.4601(6)	4.7739±0.2878(4)	0.5655±0.0055(7)	5.5752±0.1867(2)	3.9772±0.5521(5)	5.3687±0.3096(3)	**6.0223±0.0860(1)**
	25	5.6551±0.6803(6)	8.2901±0.2623(4)	0.9077±0.0050(7)	8.4299±0.4208(4)	8.5485±0.5225(4)	9.1704±0.3702(2)	**9.8641±0.0942(1)**
wfg7	2	0.3460±0.0130(6)	0.3829±0.0105(5)	0.1092±0.0007(7)	**0.4204±0.0004(1)**	0.4196±0.0007(2)	0.4188±0.0002(3.5)	0.4188±0.0005(3.5)
	3	0.5615±0.0440(6)	0.6549±0.0211(5)	0.1209±0.0000(7)	**0.7583±0.0010(1)**	0.7360±0.0017(4)	0.7548±0.0010(2)	0.7449±0.0014(3)
	5	0.8518±0.0860(6)	1.0421±0.0267(5)	0.1463±0.0000(7)	1.2677±0.0038(2)	1.0804±0.0163(4)	**1.2840±0.0031(1)**	1.2584±0.0045(3)
	7	1.1269±0.0937(6)	1.2974±0.0650(5)	0.1771±0.0004(7)	**1.7049±0.0086(1.5)**	1.3397±0.0280(4)	1.6038±0.0907(3)	**1.7000±0.0106(1.5)**
	10	1.3918±0.1735(6)	2.0242±0.0660(4.5)	0.2357±0.0000(7)	**2.4482±0.0060(2)**	2.0276±0.0773(4.5)	**2.4141±0.0638(2)**	**2.4440±0.0099(2)**
	12	1.4725±0.2122(6)	2.1892±0.2464(4.5)	0.2851±0.0004(7)	**2.9726±0.0110(1)**	2.3038±0.1306(4.5)	2.6645±0.1339(3)	2.9359±0.0194(2)
	15	2.2205±0.2748(6)	2.7870±0.3252(5)	0.3800±0.0008(7)	3.9866±0.0159(2)	3.2219±0.1321(4)	3.7738±0.1628(3)	**4.0371±0.0115(1)**
	20	3.6111±0.3759(6)	4.2830±0.3362(5)	0.6168±0.0177(7)	6.4104±0.0543(2)	4.8939±0.4791(4)	5.8699±0.1624(3)	**6.5277±0.0246(1)**
	25	6.4233±1.0674(6)	7.5748±0.4440(5)	0.9969±0.0541(7)	10.2048±0.2123(2.5)	9.4596±0.4048(4)	10.1284±0.3726(2.5)	**10.6904±0.0235(1)**
wfg8	2	0.2809±0.0075(6)	0.3027±0.0064(4.5)	0.0076±0.0230(7)	**0.3299±0.0017(1)**	0.3009±0.0015(4.5)	0.3221±0.0024(3)	0.3243±0.0016(2)
	3	0.4898±0.0258(6)	0.5556±0.0176(5)	0.1204±0.0004(7)	**0.6538±0.0024(1)**	0.6176±0.0046(4)	0.6422±0.0030(2)	0.6304±0.0031(3)
	5	0.7401±0.0472(6)	0.7954±0.0399(5)	0.1461±0.0001(7)	1.0653±0.0072(2)	0.9567±0.0099(4)	**1.0781±0.0072(1)**	1.0411±0.0074(3)
	7	0.9667±0.0709(5.5)	0.9092±0.1073(5.5)	0.1769±0.0000(7)	**1.4248±0.0154(2)**	1.2062±0.0381(4)	**1.4170±0.0538(2)**	**1.4186±0.0123(2)**
	10	1.1224±0.0912(6)	1.5561±0.1768(5)	0.2354±0.0005(7)	2.1759±0.0216(2.5)	1.8744±0.0598(4)	**2.1823±0.0500(1)**	2.1766±0.0124(2.5)
	12	1.3276±0.1107(6)	1.9617±0.2471(4.5)	0.2849±0.0004(7)	**2.6802±0.0418(1.5)**	2.0247±0.1341(4.5)	2.5040±0.1345(3)	**2.6891±0.0191(1.5)**
	15	1.9463±0.1602(6)	2.9531±0.3379(4)	0.3793±0.0007(7)	3.6871±0.0869(2)	2.8278±0.1333(5)	3.5647±0.0864(3)	**3.7733±0.0353(1)**
	20	2.9824±0.3292(6)	5.1240±0.3526(4)	0.6108±0.0013(7)	**6.2214±0.0311(1)**	4.1653±0.2093(5)	5.7492±0.2372(3)	6.1634±0.0429(2)
	25	5.5585±0.6066(6)	8.9032±0.2442(4.5)	1.0571±0.3308(7)	10.2317±0.0798(2)	9.1665±0.4875(4.5)	9.7334±0.1773(3)	**10.3434±0.0374(1)**
wfg9	2	0.3037±0.0237(6)	0.3564±0.0169(3.5)	0.1030±0.0271(7)	**0.3835±0.0330(1.5)**	**0.4055±0.0039(1.5)**	0.3493±0.0231(5)	0.3758±0.0335(3.5)
	3	0.4596±0.0595(6)	0.6116±0.0240(5)	0.0919±0.0000(7)	**0.6848±0.0383(1.5)**	**0.6743±0.0251(1.5)**	0.6370±0.0023(4)	0.6669±0.0339(3)
	5	0.7079±0.0664(6)	0.9108±0.0357(5)	0.1165±0.0000(7)	**1.0942±0.0492(2)**	**1.0267±0.0549(2)**	**1.0618±0.0028(2)**	1.0627±0.0219(4)
	7	0.9327±0.0564(6)	1.0914±0.0761(5)	0.1431±0.0001(7)	**1.4376±0.0731(2)**	1.2560±0.1225(4)	**1.3869±0.0063(2)**	**1.3916±0.0126(2)**
	10	1.4499±0.1661(6)	1.7477±0.1010(4.5)	0.1892±0.0001(7)	1.9608±0.1248(3)	1.7120±0.1009(4.5)	**2.1912±0.0919(1.5)**	**2.2331±0.0414(1.5)**
	12	1.5584±0.1890(6)	1.8958±0.1707(4.5)	0.2287±0.0000(7)	2.2009±0.1204(3)	2.0178±0.1406(4.5)	2.3813±0.1246(2)	**2.6000±0.0902(1)**
	15	2.5248±0.1976(5.5)	2.6371±0.2223(5.5)	0.3067±0.0123(7)	2.9727±0.1766(3.5)	2.9336±0.1684(3.5)	3.4368±0.1596(2)	**3.6800±0.1132(1)**
	20	3.9541±0.3817(5)	3.9678±0.4635(6)	0.5114±0.0422(7)	4.2424±0.2701(3.5)	4.4592±0.4513(3.5)	5.1012±0.3868(2)	**5.8761±0.2987(1)**
	25	6.9389±0.4429(5)	6.8244±0.4366(5)	0.8860±0.0942(7)	6.7878±0.3133(5)	8.9492±0.7306(2.5)	8.9584±0.5207(2.5)	**9.8070±0.4047(1)**

Table 3: Summary of HV performance of competing algorithms for eight WFG problems. Here B, E AND W indicate the number of times F-DEA was found better, equal and worse compared to a particular algorithm.

WFG	HV Performance on 2- and 3- objective					
F-DEA vs	HypE	MOEA/D	FD-NSGAII	SDE	NSGAIII	PICEA-g
B	16	15	16	2	8	5
E	0	1	0	1	3	4
W	0	0	0	13	5	7
	HV Performance on 5- and 7- objective					
B	16	16	15	4	15	4
E	0	0	0	5	0	3
W	0	0	1	7	1	9
	HV Performance on 10-, 12-, 15-, 20- and 25-objective					
B	40	40	35	29	40	31
E	0	0	2	4	0	3
W	0	0	3	7	0	6



(a) F-DEA (b) NSGAIII (c) FD-NSGAII (d) SDE (e) PICEAg

Figure 5: Parallel coordinate plot of different algorithms for the WFG9 problem with 15-objective. Here the non-dominated solutions are separated into two categories based on a threshold distance value from the normalized Pareto front. The solutions with a distance less than or equal to 0.025 is regarded as converged solutions (top figure, red colored), while the other ones are regarded as non-converged solutions (middle figure, blue colored). Also, to observe the simultaneous minimization of different objectives, the bottom figure shows closer inspection of all the solutions.

But as DTLZ7 is a disconnected problem, it is difficult to get the reliable estimation of IGD value for this case. Hence, we did not employ IGD for DTLZ7.

Tables 4 and 5 show the performances of different algorithms on DTLZ1, DTLZ3 and DTLZ7 problems in terms of HV and IGD, respectively.

4.6.1. DTLZ1 Problem

Although DTLZ1 has a simple linear Pareto Front ($\sum_{i=1}^{m} f_i = 0.5$), a large number of local optima ($= 11^5 - 1$) makes it difficult for an algorithm to converge into the hyper-plane. In terms of average HV and IGD, NSGAIII was better than all other competing algorithms for a smaller number of objectives. F-DEA, however, outperformed others as the number of objectives increased.

4.6.2. DTLZ3 Problem

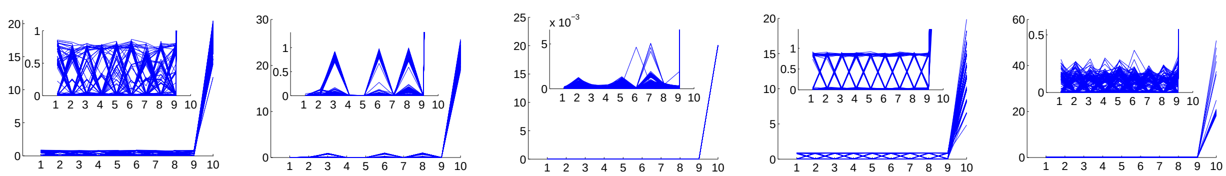
This problem has concave geometrical shape ($\sum_{i=1}^{m} f_i^2 = 1$) with a large number of local Pareto fronts parallel to the global one. This property makes it a very challenging problem. SDE and MOEA/D were top performers on this problem. In terms of HV and IGD, F-DEA was one of the best performers in 3-objective and secured the second position after SDE for the 25-objective. It shared the second position with MOEA/D in many cases while compared with respect to HV and for a larger number of objectives. MOEA/D, however, outperformed F-DEA in terms of IGD values, which caused F-DEA to achieve overall the third position.

4.6.3. DTLZ7 Problem

This problem has disconnected regions which make it interesting and challenging. In terms of HV, F-DEA exhibited superior performance for a larger number of objectives while SDE showed superior performance for a smaller ones (Table 4). For example, SDE and MOEA/D jointly secured the first position for 2-objective and the former one independently secured the first position from the 3-objective to 7-objective. NSGAIII obtained the second position for 3-objective and F-DEA secured the second position for 5-objective and 7-objective. For a larger number of objectives (from 10-objective to 25-objective), F-DEA was the best and PICEAg was next to it. Most of the algorithms did not able to converge within the reference point bound which constituted their small HV values for a larger number of objectives.

Tables 6 and 7, respectively, show the number of times F-DEA is found better, worse or equal than any competing algorithm based on the Wilcoxon rank-sum test applied on the HV and IGD values of the DTLZ problems. In terms of HV and IGD, the three algorithms SDE, NSGAIII, PICEAg and F-DEA showed a very similar performance for a smaller number of objectives, 2-objective to 7-objective. However, for higher number of objectives F-DEA outperforms others in DTLZ1, DTLZ7 problems and shows competitive performance with SDE and MOEA/D at DTLZ3 problem.

Fig. 6 shows the parallel coordinate plot of the competing algorithms for the DTLZ7 problem with 10-objective. The upper bound of the last objective for this problem is $2 \times m$ or $f_{10} \leq 20$. It can be seen from the figure that F-DEA was able to maintain diversity and convergence together within the Pareto optimal front. SDE (Fig. 6(d)) maintained diversity and convergence well but F-DEA outperforms SDE by having more objective value variation in the first 9 objectives. PICEAg (Fig. 6(e)) converged in the first nine objective but few solutions converge on 10-th objective. Similarly, NSGAIII (Fig. 6(b)) also converges in the first 9 objectives but only few converges in the 10-th objective. FD-NSGAII (Fig. 6(c)) converged solutions into a region as expected.



(a) F-DEA (b) NSGAIII (c) FD-NSGAII (d) SDE (e) PICEAg

Figure 6: Parallel coordinate plot of all competing algorithms in 10− objective DTLZ7 problem. The inset figure shows the closer inspection of the first 9 objectives.

4.7. Discussion

The results presented in the previous sections and supplementary material give an idea about the performances of F-DEA with respect to different competing algorithms. This section briefly explains the reasons behind such performances.

F-DEA maintains cluster uniformity based on preferred reference points. If a parent population lies in crowded regions and any offspring solution lies in a different region, then the solution will form a new cluster and F-DEA will select it for the next generation. This property helps F-DEA expanding its search region whenever possible. And this is beneficial for a deceptive problem containing large-size hill and disconnected problems having isolated regions. However, this property is not beneficial for a degenerate problem. As the isolated solution increase the search region, it will take some generations to converge. The fuzzy dominance with adaptive membership functions can optimize different objectives well in case of a large number of objectives (Fig. 5), and the bias induced for that was mitigated from the cluster uniformity.

The poor performance of FD-NSGAII [19] was due to lack of diversity among solutions (5(c) and 6(c)). NSGAIII [1] emphasizes on solutions that are non-dominated and close to reference line. When the number

Table 4: Average HV of different algorithms on DTLZ1, DTLZ3 and DTLZ7 problems over 20 independent runs. The best result based on the Wilcoxon rank sum test with a significance level of 0.05 is marked in bold-face. The rank of a particular algorithm is shown in bracket.

Prob.	m	HYPE	MOEA/D	FD-NSGAII	SDE	NSGAIII	PICEA-g	F-DEA
dtlz1	2	0.6596±0.0422(6)	**0.7069±0.0005(2)**	0.0000±0.0000(7)	**0.7066±0.0002(2)**	**0.7063±0.0012(2)**	0.7061±0.0005(4.5)	0.7063±0.0002(4.5)
	3	0.6849±0.5932(6)	1.0884±0.0088(5)	0.0000±0.0000(7)	1.1261±0.0023(3.5)	**1.1246±0.0277(1.5)**	**1.1313±0.0045(1.5)**	1.1249±0.0020(3.5)
	5	0.0067±0.0116(6)	1.5142±0.0098(5)	0.0000±0.0000(7)	1.5481±0.0046(4)	**1.5707±0.0076(1)**	1.5297±0.0459(2.5)	1.5582±0.0030(2.5)
	7	0.6145±0.7117(6)	1.8803±0.0094(4)	0.0000±0.0000(7)	1.9108±0.0031(3)	**1.9095±0.1091(1)**	1.7799±0.0819(5)	1.9272±0.0026(2)
	10	0.6077±1.0526(6)	2.4924±0.0229(3)	0.0090±0.0331(7)	2.5591±0.0044(2)	1.7479±0.9172(4.5)	2.2692±0.3215(4.5)	**2.5829±0.0024(1)**
	12	0.0141±0.0231(6)	2.9727±0.0335(3)	0.0306±0.1175(7)	3.0855±0.0093(2)	1.8917±1.1930(4.5)	2.5134±0.6594(4.5)	**3.1253±0.0034(1)**
	15	0.7807±0.6792(6)	4.0354±0.0273(3)	0.0152±0.0680(7)	4.1336±0.0064(2)	3.8412±0.3188(4.5)	3.4812±1.0382(4.5)	**4.1680±0.0021(1)**
	20	0.0000±0.0000(6.5)	6.4428±0.0967(3)	0.1024±0.2214(6.5)	6.6532±0.0101(2)	3.2618±2.7249(4.5)	5.1065±2.0933(4.5)	**6.7005±0.0072(1)**
	25	0.3519±0.6122(5.5)	0.0000±0.0000(7)	0.2169±0.3563(5.5)	10.7567±0.0128(2)	6.9000±4.0868(4)	10.3842±0.3817(3)	**10.8130±0.0069(1)**
dtlz3	2	0.0830±0.1437(5.5)	0.4139±0.0068(1.5)	0.0000±0.0000(7)	**0.4104±0.0096(1.5)**	0.3796±0.0911(4)	0.1540±0.1810(5.5)	0.4041±0.0120(3)
	3	0.1338±0.2317(5.5)	0.6931±0.0150(3)	0.0000±0.0000(7)	0.4101±0.2904(4)	0.1595±0.2052(5.5)	**0.4626±0.3281(1.5)**	**0.7261±0.0130(1.5)**
	5	0.0000±0.0000(6.5)	1.1638±0.0221(2)	0.0000±0.0000(6.5)	**1.2779±0.0197(1)**	0.0673±0.1730(4.5)	0.6202±0.4698(3)	0.1174±0.2130(4.5)
	7	0.0000±0.0000(6)	1.4667±0.3553(2)	0.0000±0.0000(6)	**1.7475±0.0093(1)**	0.1311±0.3230(6)	0.5283±0.6051(3.5)	0.8720±0.7061(3.5)
	10	0.0000±0.0000(6)	1.9846±0.4769(2.5)	0.0000±0.0000(6)	**2.4098±0.2289(1)**	0.0000±0.0000(6)	0.6794±0.7619(4)	1.7546±0.9708(2.5)
	12	0.0000±0.0000(5.5)	1.6366±1.0472(2.5)	0.0000±0.0000(5.5)	**2.5020±1.0833(1)**	0.0000±0.0000(5.5)	0.1847±0.5736(5.5)	1.3304±1.3433(2.5)
	15	0.0000±0.0000(6)	3.4835±0.0759(2.5)	0.0000±0.0000(6)	**4.0894±0.0245(1)**	0.0000±0.0000(6)	0.9834±1.3996(4)	2.2773±1.8582(2.5)
	20	0.0000±0.0000(6)	4.6077±1.7082(2)	0.0000±0.0000(6)	**6.5836±0.0571(1)**	0.0000±0.0000(6)	1.3989±2.1580(3.5)	1.5242±2.5204(3.5)
	25	0.0000±0.0000(5.5)	0.0000±0.0000(5.5)	0.0000±0.0000(5.5)	**10.7668±0.0279(1)**	0.0000±0.0000(5.5)	5.5089±3.6702(2.5)	3.9419±4.2086(2.5)
dtlz7	2	0.3190±0.0101(6)	**0.3337±0.0000(1.5)**	0.1133±0.0143(7)	**0.3337±0.0000(1.5)**	0.3337±0.0000(3)	0.3165±0.0372(5)	0.3334±0.0001(4)
	3	0.3753±0.0141(6)	0.4103±0.0070(5)	0.1207±0.0001(7)	**0.4351±0.0008(1)**	0.4327±0.0006(2)	0.4033±0.0560(3.5)	0.4281±0.0015(3.5)
	5	0.4074±0.0018(5)	0.3423±0.0376(6)	0.1436±0.0003(7)	**0.5248±0.0036(1)**	0.4318±0.0241(3.5)	0.4079±0.0494(3.5)	0.4834±0.0038(2)
	7	0.3774±0.0159(3)	0.2361±0.0553(6)	0.1535±0.0030(7)	**0.5093±0.0074(1)**	0.3231±0.0322(5)	0.3547±0.0204(4)	0.4822±0.0061(2)
	10	0.2845±0.0445(3.5)	0.0345±0.0243(7)	0.1649±0.0014(5)	0.3119±0.0479(3.5)	0.0848±0.0555(6)	0.3405±0.0253(2)	**0.4563±0.0149(1)**
	12	0.2441±0.0042(2.5)	0.0052±0.0063(6)	0.1735±0.0048(4.5)	0.1591±0.0329(4.5)	0.0010±0.0023(7)	0.2522±0.0418(2.5)	**0.5079±0.0507(1)**
	15	0.2748±0.0394(3)	0.0002±0.0003(6)	0.1636±0.0012(4)	0.0283±0.0099(5)	0.0001±0.0005(7)	0.3346±0.0274(2)	**0.3831±0.0245(1)**
	20	0.1714±0.0362(2)	0.0000±0.0000(6.5)	0.0834±0.0066(3.5)	0.0086±0.0125(5)	0.0000±0.0000(6.5)	0.0889±0.0836(3.5)	**0.3275±0.0571(1)**
	25	0.0000±0.0000(5.5)	0.0000±0.0000(5.5)	0.0000±0.0000(5.5)	0.0005±0.0005(3)	0.0000±0.0000(5.5)	0.1203±0.0984(2)	**0.1932±0.0399(1)**

Table 5: Average IGD of different algorithms on DTLZ1 and DTLZ3 problems over 20 independent runs. The best result based on the Wilcoxon rank sum test with a significance level of 0.05 is marked in bold-face. The rank of a particular algorithm is shown in bracket.

Prob.	m	HYPE	MOEA/D	FD-NSGAII	SDE	NSGAIII	PICEA-g	F-DEA
dtlz1	2	0.0016±0.0010(6)	0.0000±0.0000(2)	0.3873±0.1489(7)	0.0000±0.0000(3)	**0.0000±0.0000(1)**	0.0000±0.0000(4)	0.0001±0.0000(5)
	3	0.0117±0.0164(6)	0.0021±0.0001(5)	0.5881±0.2721(7)	**0.0011±0.0000(1)**	0.0009±0.0011(4)	0.0011±0.0002(2)	0.0012±0.0001(3)
	5	0.0512±0.0312(6)	0.0052±0.0000(4)	0.3606±0.1076(7)	0.0041±0.0000(2)	**0.0020±0.0013(1)**	0.0076±0.0033(5)	0.0042±0.0001(3)
	7	0.0592±0.0444(6)	0.0077±0.0001(4)	0.2067±0.0495(7)	0.0067±0.0001(2.5)	0.0040±0.0044(2.5)	0.0182±0.0020(5)	**0.0063±0.0001(1)**
	10	0.0631±0.0359(6)	0.0116±0.0007(3)	0.0821±0.0310(7)	0.0092±0.0001(2)	0.0254±0.0083(5)	0.0226±0.0021(4)	**0.0083±0.0002(1)**
	12	0.1058±0.0682(7)	0.0152±0.0007(3)	0.1077±0.0509(6)	**0.0113±0.0002(1)**	0.0335±0.0125(4)	0.0301±0.0106(5)	0.0116±0.0004(2)
	15	0.0929±0.0674(7)	0.0132±0.0010(3)	0.0775±0.0366(6)	0.0095±0.0002(2)	0.0202±0.0016(4)	0.0235±0.0058(5)	**0.0094±0.0002(1)**
	20	0.1201±0.0407(7)	0.0204±0.0006(3)	0.0728±0.0335(6)	0.0155±0.0003(2)	0.0370±0.0128(5)	0.0320±0.0110(4)	**0.0141±0.0009(1)**
	25	0.0165±0.0093(6)	0.1045±0.0217(7)	0.0150±0.0049(5)	**0.0033±0.0000(1)**	0.0077±0.0029(4)	0.0060±0.0003(3)	0.0036±0.0002(2)
dtlz3	2	0.0435±0.0396(5)	**0.0004±0.0003(1)**	1.6814±0.7393(7)	0.0008±0.0003(2)	0.0039±0.0122(4)	0.0586±0.0755(6)	0.0008±0.0005(3)
	3	0.1586±0.1130(6)	0.0056±0.0004(2)	1.1440±0.4942(7)	0.0271±0.0269(3)	0.0493±0.0292(4)	0.0317±0.0395(5)	**0.0037±0.0003(1)**
	5	0.6058±0.2773(6)	0.0150±0.0004(2)	0.9245±0.2767(7)	**0.0130±0.0005(1)**	0.1037±0.0649(5)	0.0608±0.0397(3)	0.0904±0.0624(4)
	7	0.9081±0.3801(7)	0.0272±0.0120(2)	0.8025±0.2808(6)	**0.0173±0.0006(1)**	0.1592±0.1150(5)	0.1012±0.0586(4)	0.0528±0.0418(3)
	10	0.9161±0.2690(7)	0.0404±0.0119(2)	0.6995±0.2610(5)	**0.0245±0.0048(1)**	0.6902±0.3276(6)	0.1096±0.0593(4)	0.0396±0.0246(3)
	12	1.3816±0.4275(6)	0.0786±0.0540(2)	1.1271±0.4369(5)	**0.0494±0.0359(1)**	1.0651±0.6395(7)	0.2625±0.1429(4)	0.1004±0.0884(3)
	15	0.8515±0.3101(6)	0.0469±0.0046(2)	0.5108±0.1996(5)	**0.0289±0.0031(1)**	0.9245±0.4516(7)	0.1055±0.0299(4)	0.0510±0.0317(3)
	20	1.2758±0.4770(6)	0.0761±0.0293(2)	0.6600±0.2519(5)	**0.0374±0.0033(1)**	1.4316±0.4919(7)	0.1644±0.0839(4)	0.1337±0.0769(3)
	25	2.8646±0.7739(6)	3.0273±0.3555(7)	0.1204±0.0454(4)	**0.0107±0.0010(1)**	0.3112±0.0964(5)	0.0249±0.0084(3)	0.0230±0.0098(2)

Table 6: Summary of HV performance of competing algorithms for DTLZ1, DTLZ3 and DTLZ7 problems. Here B, E AND W indicate the number of times F-DEA was found better compared to a particular algorithm.

DTLZ	HV Performance on 2- and 3- objective					
F-DEA vs	HypE	MOEA/D	FD-NSGAII	SDE	NSGAIII	PICEA-g
B	6	3	6	1	2	2
E	0	0	0	1	0	3
W	0	3	0	4	4	1
	HV Performance on 5- and 7- objective					
B	6	4	6	2	3	3
E	0	0	0	0	1	2
W	0	2	0	4	2	1
	HV Performance on 10-, 12-, 15-, 20- and 25-objective					
B	15	11	15	10	15	13
E	0	3	0	0	0	2
W	0	1	0	5	0	0

Table 7: Summary of IGD performance of competing algorithms for DTLZ1 and DTL3 problems. Here B, E AND W indicate the number of times F-DEA was found better compared to a particular algorithm.

DTLZ	IGD Performance on 2- and 3- objective					
F-DEA vs	HypE	MOEA/D	FD-NSGAII	SDE	NSGAIII	PICEA-g
B	4	2	4	1	3	2
E	0	0	0	0	0	0
W	0	2	0	3	1	2
	IGD Performance on 5- and 7- objective					
B	4	2	4	1	3	3
E	0	0	0	0	0	0
W	0	2	0	3	1	1
	IGD Performance on 10-, 12-, 15-, 20- and 25-objective					
B	10	6	10	3	10	10
E	0	0	0	0	0	0
W	0	4	0	7	0	0

of objectives is large, the Pareto-dominance relied on by NSGAIII lacks enough selection pressure to push the population towards Pareto front. In a sense, NSGA-III stresses diversity more than convergence [23]. Our F-DEA uses fuzzy dominance that is able to maintain good selection pressure in high dimensional objective space. As the number of objectives increases fuzzy dominance becomes more effective to differentiating which solutions are better. Also the preferred reference points based clustering method promotes diversity using those reference points where solution exists rather than entire high dimensional objective space.

SDE [6] performs relatively well in the DTLZ problem suite than the WFG one. The reason most likely is the normalized nature of the former problem suite. The performance of F-DEA for the WFG problem signifies that the proposed algorithm was able to handle the scaling issue well due to the use of scale independent adaptive fuzzy membership function.

The co-evolutionary algorithm PICEAg is goal oriented and Pareto based algorithm. The maintenance of an increasing number of goal vectors by this algorithm enhances the comparability among solutions in a high dimensional objective space. The inherent tendency to maintain diversity sometimes responsible for reducing selection pressure. This can be understood by looking the results of the algorithm for the WFG1 problem for which it performed worse compared to its performance on other problems (Table 2).

The obtained solutions of MOEA/D in high dimension might achieve good aggregation values but far away from the corresponding weight vector. This property makes harder for MOEA/D to maintain diversity for problems with a large number of objectives. HypE failed to maintain diversity in some cases and pushed the solutions into the corner of the Pareto front.

The test problems in the WFG suite are far more difficult to solve and hard to maintain diversity. It is because the WFG problems have more transformation functions and they have different ranges for different objectives. The proposed algorithm F-DEA was able to outperform competing algorithms on these problems with a larger number of objectives. From the results of the DTLZ suite, we see that for a larger number of objectives, F-DEA was able to solve difficult to converge DTLZ1 and disconnected DTLZ7 problems better than others.

4.8. Effect of Reference Point Based Clustering and Fuzzy Dominance

To investigate how much benefit we get from reference point based clustering and fuzzy dominance over Pareto dominance, we evaluate two variants, F-DEA* and F-DEA$^\#$, of basic F-DEA on DTLZ1 and DTLZ3 problems with the usual settings. While F-DEA* does not employ reference points based clustering, F-DEA$^\#$ employs Pareto-dominance based fitness assignment instead of fuzzy dominance based fitness assignment. Tables 8 shows the comparative performances of F-DEA* and F-DEA$^\#$ against basic F-DEA in terms of HV and IGD. For brevity, we consider DTLZ1 and DTLZ3 problems with 7-, 10- and 15-objective.

It is clear from Tables 8 that the basic F-DEA outperforms both F-DEA* and F-DEA$^\#$ significantly in all problem instances. The large IGD values and small (in fact zero) HV values of F-DEA$^\#$ in DTLZ1

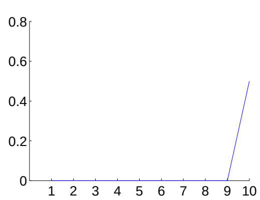
Table 8: Comparison among three different versions of the proposed algorithm, basic version (F-DEA), F-DEA without clustering (F-DEA*), and F-DEA with Pareto dominance (F-DEA$^\#$), on DTLZ problems based on IGD and HV value.

		IGD		
Prob.	Obj.	F-DEA*	F-DEA$^\#$	F-DEA
DTLZ1	7	0.0296±0.0152(2)	9.0918±1.5703(3)	**0.0063±0.0001(1)**
	10	0.0302±0.0155(2)	7.6205±1.1507(3)	**0.0083±0.0002(1)**
	15	0.0269±0.0159(2)	5.8475±1.1274(3)	**0.0094±0.0002(1)**
DTLZ3	7	0.0861±0.0001(2)	83.6979±7.0178(3)	**0.0528±0.0418(1)**
	10	0.0905±0.0120(2)	70.5014±6.9913(3)	**0.0396±0.0246(1)**
	15	0.0856±0.0001(2)	56.6080±6.5905(3)	**0.0510±0.0317(1)**
		HV		
DTLZ1	7	0.1410±0.0723(2)	0.0000±0.0000(3)	**1.9272±0.0026(1)**
	10	0.1877±0.0963(2)	0.0000±0.0000(3)	**2.5829±0.0024(1)**
	15	0.2839±0.1682(2)	0.0000±0.0000(3)	**4.1680±0.0021(1)**
DTLZ3	7	0.1684±0.0053(2)	0.0000±0.0000(3)	**0.8720±0.7061(1)**
	10	0.2156±0.0511(2)	0.0000±0.0000(3)	**1.7546±0.9708(1)**
	15	0.3683±0.0096(2)	0.0000±0.0000(3)	**2.2773±1.8582(1)**

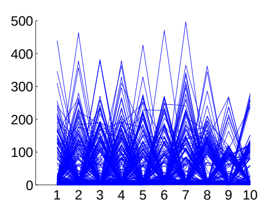
and DTLZ3 suggest that the Pareto based selection failed to create necessary selection pressure on hard to converge problems. F-DEA*, on the other hand, convergences into a part of Pareto front due to the bias created from fuzzy dominance. The IGD and HV values of F-DEA* are better than F-DEA$^\#$ as no solution obtained by F-DEA$^\#$ converges into Pareto front.

Fig.7 shows the non-dominated solutions obtained by F-DEA*, F-DEA$^\#$ and basic F-DEA for DTLZ1 problem with in 10-objective. Basic F-DEA achieved good convergence (overall objective range $[0 - 0.5]$) while maintaining good diversity (Fig. 7(c)). In contrast, F-DEA$^\#$ (Fig. 7(b)) maintained diversity due to reference points based clustering but failed to converge (overall objective range $[0 - 500]$) because of Pareto based selection. F-DEA* (Fig. 7(a)) converged to a part of Pareto front but severely lacked diversity.

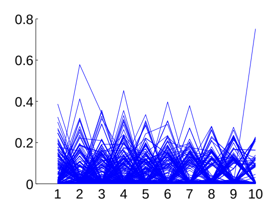
The performances of two variants F-DEA* and F-DEA$^\#$ indicate the importance of using clustering and fuzzy dominance based selection in solving MaOPs. The fuzzy based selection alone cannot maintain diversity and clustering does not work with Pareto based approach due to its lack of comparability for a large number of objective. This is why F-DEA harvests the benefits from these two techniques, clustering and fuzzy dominance, and compliments each others' weakness.



(a) F-DEA*



(b) F-DEA$^\#$



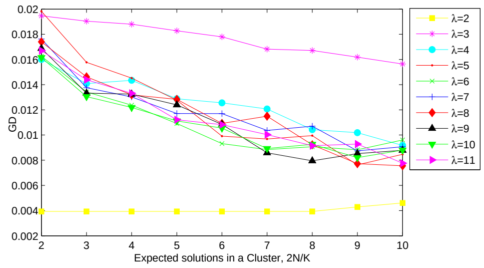
(c) Basic F-DEA

Figure 7: Parallel coordinate plots of variants of F-DEA on 10− objective DTLZ1 problem.

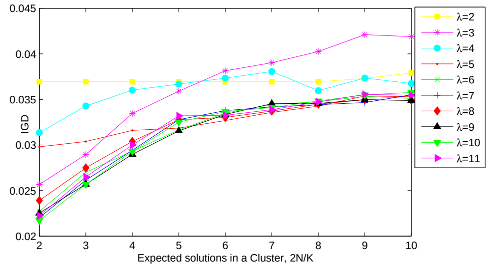
4.9. Parameter Sensitivity

In the absence of supplied reference points, F-DEA employs the Das and Dennis [29] procedure for generating such points. This procedure requires a parameter λ, the number of division in an objective. It is worth mentioning that λ is necessary if any evolutionary algorithm (see, for example, [1]) employs this procedure. The aim of this section is to show why we select p (=N, the population size) preferred points from a large number of generated points.

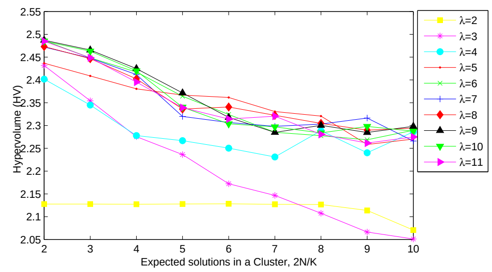
Trivially, an evolutionary algorithm maintains diversity and convergence by its own way, where the decision maker has no control. In F-DEA, the decision maker can control them using p. Furthermore, λ and p together



(a) GD values of DTLZ2 with 10-objective



(b) IGD values of DTLZ2 with 10-objective



(c) HV values of DTLZ2 with 10-objective

Figure 8: Effect of λ and χ on GD, IGD and HV performance of F-DEA on the DTLZ2 problem with 10-objective.

remove the constraint to population size for any number of objectives unlike NSGAIII [1, 23] in which the population size is bounded by λ and not flexible for an arbitrary number of objectives.

Fig. 8 shows impact of λ and p on the GD, IGD and HV performances of DTLZ2 problem with 10-objective. The experiments were conducted with the usual parameter settings but varying λ and p. For convenience, p was changed in such a way that the expected number of solutions in each cluster, χ, becomes $2, 3, \cdots, 10$.

We can see as λ increases so does overall HV (Fig. 8(c)) and reduces GD (Fig. 8(a)) and IGD (Fig. 8(b)) up to a certain level. Increasing λ increases the number of generated reference points exponentially. A small number of reference points do not cover the entire objective space. Generating a large number of reference points maintains better cluster uniformity which in turn improves overall performance. But if we continue increasing reference points then after a certain λ, the clusters' representative points will be nearly same as before and performance will be clipped. It is also expensive to generate a huge number of reference points. Hence, a reasonable number of reference points should be used. By inspecting the performance on DTLZ2, it is clear that after $\lambda = 9$ the performance doesn't improve increasing λ (Fig. 8). It is to be noted that HV for $\lambda = 3$ is greater than the HV value for $\lambda = 4$ and this behavior is problem dependent. Therefore it is sufficient to use $\lambda = 9$ for 10-objective problem.

As p decreases, the number of clusters decreases and the expected number of solutions, χ, in a cluster increases for a fixed-size combined population, $|C_t| = 2N$ (e.g. for $p = N$, $\chi = 2N/N = 2$ and for $p = \frac{N}{2}$, $\chi = 2N/\frac{N}{2} = 4$). The fuzzy dominance relation within a cluster promotes faster convergence. By decreasing p, the solutions achieve faster convergence while losing some diversity. A closer inspection of Fig. 8(a) reveals that as p decreases (i.e., increasing χ), the overall GD value decreases i.e., convergence increases. The overall performance of HV decreases (Fig. 8(c)) and IGD value increases (Fig. 8(b)), which indicates reduction in diversity. Therefore $p = N$ or $\chi = 2$ is preferable as it works as the best compromise between convergence and diversity. We might require more converged optimal solutions compromising some diversity in some real-world applications. It is possible to achieve this goal by decreasing p. An opposite scenario i.e., obtaining more diverse solutions can also be achieved by increasing p.

5. Conclusion and Future Work

Evolutionary algorithms can provide several candidate solutions in a single run, which make them popular to solve many practical problems including MaOPs. However, the loss of selection pressure is a challenging issue for such algorithms while solving MaOPs. In this paper, we have incorporated fuzzy-dominance and reference points in the environmental selection mechanism of the proposed F-DEA with an aim of improving selection pressure. The introduction of reference point in conjunction of fuzzy-dominance not only helps in maintaining diversity of the evolved solutions but also convergence.

F-DEA has been extensively evaluated and compared using eight WFG and three DTLZ problems having 2- to 25-objectives. The simulation results reveal that F-DEA in general performs better than other algorithms on complex problems with an increasing number of objectives. It has also been found that F-DEA can balance between the conflicting goal of convergence and diversity well in comparison with other algorithms, especially for complex problems.

In its current implementation, the reference point generation procedure [29] used in F-DEA has one user-specified parameter, which was set after some preliminary experiments. One of the future avenues would be to make it adaptive. It would be interesting in the future to analyze F-DEA further and identify its strength and weakness. It would also be interesting to apply F-DEA to real-world problems.

References

[1] K. Deb, H. Jain, An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: solving problems with box constraints, IEEE Transactions on Evolutionary Computation 18 (4) (2014) 577–601.

[2] H. Ishibuchi, Y. Hitotsuyanagi, N. Tsukamoto, Y. Nojima, Many-objective test problems to visually examine the behavior of multiobjective evolution in a decision space, in: Parallel Problem Solving from Nature, PPSN XI, Springer, 2010, pp. 91–100.

[3] S. Yang, M. Li, X. Liu, J. Zheng, A grid-based evolutionary algorithm for many-objective optimization, IEEE Transactions on Evolutionary Computation 17 (5) (2013) 721–736.

[4] S. F. Adra, P. J. Fleming, Diversity management in evolutionary many-objective optimization, IEEE Transactions on Evolutionary Computation 15 (2) (2011) 183–195.

[5] H. Wang, Y. Jin, X. Yao, Diversity assessment in many-objective optimization, IEEE Transactions on Cybernetics PP (99) (2016) 1–13.

[6] M. Li, S. Yang, X. Liu, Shift-based density estimation for pareto-based algorithms in many-objective optimization, IEEE Transactions on Evolutionary Computation 18 (3) (2014) 348–365.

[7] X. Zhang, Y. Tian, Y. Jin, A knee point-driven evolutionary algorithm for many-objective optimization, IEEE Transactions on Evolutionary Computation 19 (6) (2015) 761–776.

[8] M. Garza-Fabre, G. Toscano-Pulido, C. A. C. Coello, Two novel approaches for many-objective optimization, in: Proceedings of the IEEE Congress on Evolutionary Computation (CEC), 2010, pp. 1–8.

[9] Q. Zhang, H. Li, Moea/d: A multiobjective evolutionary algorithm based on decomposition, IEEE Transactions on Evolutionary Computation 11 (6) (2007) 712–731.

[10] H. Li, Q. Zhang, Multiobjective optimization problems with complicated pareto sets, moea/d and nsga-ii, IEEE Transactions on Evolutionary Computation 13 (2) (2009) 284–302.

[11] Q. Zhang, W. Liu, H. Li, The performance of a new version of moea/d on cec09 unconstrained mop test instances., in: Proceedings of the IEEE Congress on Evolutionary Computation (CEC), Vol. 1, 2009, pp. 203–208.

[12] X. Cai, Z. Yang, Z. Fan, Q. Zhang, Decomposition-based-sorting and angle-based-selection for evolutionary multiobjective and many-objective optimization, IEEE Transactions on Cybernetics PP (99) (2016) 1–14.

[13] R. Cheng, Y. Jin, M. Olhofer, B. Sendhoff, A reference vector guided evolutionary algorithm for many-objective optimization, IEEE Transactions on Evolutionary Computation 20 (5) (2016) 773–791.

[14] R. Wang, R. C. Purshouse, P. J. Fleming, Preference-inspired coevolutionary algorithms for many-objective optimization, IEEE Transactions on Evolutionary Computation 17 (4) (2013) 474–494.

[15] E. Zitzler, L. Thiele, Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach, IEEE transactions on Evolutionary Computation 3 (4) (1999) 257–271.

[16] J. Bader, E. Zitzler, Hype: An algorithm for fast hypervolume-based many-objective optimization, IEEE Transactions on Evolutionary Computation 19 (1) (2011) 45–76.

[17] H. Wang, L. Jiao, X. Yao, An improved two-archive algorithm for many-objective optimization, IEEE Transactions on Evolutionary Computation 19 (4) (2015) 524–541.

[18] Z. He, G. G. Yen, J. Zhang, Fuzzy-based pareto optimality for many-objective evolutionary algorithms, IEEE Transactions on Evolutionary Computation 18 (2) (2014) 269–285.

[19] Z. He, G. G. Yen, A new fitness evaluation method based on fuzzy logic in multiobjective evolutionary algorithms, in: Proceedings of the IEEE Congress on Evolutionary Computation (CEC), 2012, pp. 1–8.

[20] M. Köppen, R. Vicente-Garcia, B. Nickolay, Fuzzy-pareto-dominance and its application in evolutionary multi-objective optimization, in: Evolutionary Multi-Criterion Optimization, Springer, 2005, pp. 399–412.

[21] M. Farina, P. Amato, A fuzzy definition of" optimality" for many-criteria optimization problems, IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans 34 (3) (2004) 315–326.

[22] M. Nasir, A. Mondal, S. Sengupta, S. Das, A. Abraham, An improved multiobjective evolutionary algorithm based on decomposition with fuzzy dominance, in: Proceedings of the IEEE Congress on Evolutionary Computation (CEC), 2011, pp. 765–772.

[23] Y. Yuan, H. Xu, B. Wang, X. Yao, A new dominance relation-based evolutionary algorithm for many-objective optimization, IEEE Transactions on Evolutionary Computation 20 (1) (2016) 16–37.

[24] K. Deb, J. Sundar, N. Udaya Bhaskara Rao, S. Chaudhuri, Reference point based multi-objective optimization using evolutionary algorithms, International Journal of Computational Intelligence Research 2 (3) (2006) 273–286.

[25] P. C. Roy, M. M. Islam, K. Murase, X. Yao, Evolutionary path control strategy for solving many-objective optimization problem, IEEE Transactions on Cybernetics 45 (4) (2015) 702–715.

[26] J. M. Mendel, Fuzzy logic systems for engineering: a tutorial, Proceedings of the IEEE 83 (3) (1995) 345–377.

[27] M. Li, S. Yang, X. Liu, A test problem for visual investigation of high-dimensional multi-objective search, in: Proceedings of the IEEE Congress on Evolutionary Computation (CEC), 2014, pp. 2140–2147.

[28] R. C. Purshouse, P. J. Fleming, On the evolutionary optimization of many conflicting objectives, IEEE Transactions on Evolutionary Computation 11 (6) (2007) 770–784.

[29] I. Das, J. E. Dennis, Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems, SIAM Journal on Optimization 8 (3) (1998) 631–657.

[30] D. E. Knuth, The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms, 1997.

[31] S. Huband, P. Hingston, L. Barone, L. While, A review of multiobjective test problems and a scalable test problem toolkit, IEEE Transactions on Evolutionary Computation 10 (5) (2006) 477–506.

[32] K. Deb, L. Thiele, M. Laumanns, E. Zitzler, Scalable Test Problems for Evolutionary Multiobjective Optimization, 2005, pp. 105–145.

[33] H. Ishibuchi, H. Masuda, Y. Nojima, Pareto fronts of many-objective degenerate test problems, IEEE Transactions on Evolutionary Computation 20 (5) (2016) 807–813.

[34] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, V. G. Da Fonseca, Performance assessment of multiobjective optimizers: an analysis and review, IEEE Transactions on Evolutionary Computation 7 (2) (2003) 117–132.

[35] T. Wagner, N. Beume, B. Naujoks, Pareto-, aggregation-, and indicator-based methods in many-objective optimization, in: Evolutionary multi-criterion optimization, Springer, 2007, pp. 742–756.

[36] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: Nsga-ii, IEEE Transactions on Evolutionary Computation 6 (2) (2002) 182–197.

[37] W. Haynes, Wilcoxon rank sum test, in: Encyclopedia of Systems Biology, Springer, 2013, pp. 2354–2355.

[38] H. Abdi, The bonferonni and šidák corrections for multiple comparisons, Encyclopedia of measurement and statistics 3 (2007) 103–107.