



KnRVEA: A hybrid evolutionary algorithm based on knee points and reference vector adaptation strategies for many-objective optimization

Gaurav Dhiman¹  · Vijay Kumar¹

Published online: 19 January 2019
© Springer Science+Business Media, LLC, part of Springer Nature 2019

Abstract

In this paper, a many-objective evolutionary algorithm, named as a hybrid evolutionary algorithm based on knee points and reference vector adaptation strategies (KnRVEA) is proposed. Knee point strategy is used to improve the convergence of solution vectors. In the proposed algorithm, a novel knee adaptation strategy is introduced to adjust the distribution of knee points. KnRVEA is compared with five well-known evolutionary algorithms over thirteen benchmark test functions. The results reveal that the proposed algorithm provides better results than the others in terms of Inverted Generational Distance and Hypervolume. The computational complexity of the proposed algorithm is also analyzed. The statistical testing is performed to show the statistical significance of proposed algorithm. The proposed algorithm is also applied on three real-life constrained many-objective optimization problems to demonstrate its efficiency. The experimental results show that the proposed algorithm is able to solve many-objective real-life problems.

Keywords Evolutionary multi-objective optimization · Many-objective optimization · Convergence · Diversity

1 Introduction

In the last few decades, researchers have developed a number of new Evolutionary based Multiobjective Optimization (EMO) algorithms for solving problems that have two or three objectives [3–5]. Evolutionary Algorithms (EA) are population based heuristic algorithms that are capable for obtaining the set of solutions [63–68, 72]. However, many real-life problems such as automotive engine calibration [6], water distribution systems [7], fuzzy systems [69, 70, 73], software engineering [71], and land use management [8] problems, often require a number of objectives. Therefore, Multiobjective Evolutionary Algorithms (MOEA) face some challenges to handle large number of objectives. Most of MOEAs are unable to produce optimal solution due to

their selection strategies. This strategy is helpful to converge the solutions towards the Pareto front. When the number of objectives increases, it is difficult to maintain the diversity as the solutions will allocate very sparsely in the objective search space.

The two well-known multi-objective evolutionary algorithms are Non-dominated Sorting Genetic Algorithm 2 (NSGA-II) [9] and Strength Pareto Evolutionary Algorithm 2 (SPEA2) [42]. Both of these algorithms use dominance based selection strategy which fails to solve more than three objective problems because the non-dominated candidate solutions lie in limited range of population. To solve this problem, researchers have developed new optimization algorithm named as Many-objective Optimization (MaOP) to handle a large number of objectives [50]. The main challenges in MaOP are as follows:

- The visualization of Pareto front in a given search space requires special techniques [54].
- The number of solutions has increased exponentially to describe the Pareto front of multi dimensional problems [1, 2].

Diversity maintenance and convergence improvement are the main focus of MOEAs. However, these algorithms do not consider the use of preferences for many-objective

✉ Gaurav Dhiman
gdhiman0001@gmail.com

Vijay Kumar
vijaykumarchahar@gmail.com

¹ Computer Science and Engineering Department,
Thapar Institute of Engineering & Technology,
Patiala 147004, Punjab, India

problems. This is so because the researchers may be focused on Pareto optimal solutions and less computational efforts to achieve a demonstrative subset of the Pareto front using evolutionary approaches [50].

Apart from this, there are various performance indicator based algorithms that have been developed to improve the efficiency of algorithms for many-objective optimization problems [10–12]. One of the most popular hypervolume indicator based algorithms is the Hypervolume Estimation Algorithm (HypE) [13] which uses Monte Carlo simulation to estimate the exact hypervolume values. Unfortunately, when the number of objective increases, then computational complexity of the algorithm also increases [15].

The main contribution of this paper is to propose a hybrid algorithm that utilizes the concepts of Knee Point Driven Evolutionary Algorithm (KnEA) and Reference Vector Guided Evolutionary Algorithm (RVEA). The algorithm is named as a hybrid evolutionary algorithm based on knee points and reference vector adaptation strategies (KnRVEA). The proposed KnRVEA algorithm has five main strategies namely mating selection, variation, finding knee point, environmental selection, and knee adaptation strategy. The mating selection strategy is used to exchange the information between individuals and pick the most promising optimal solution from the current population. The variation utilizes simulated binary crossover (SBX) [59] and the polynomial mutation [60] operators to produce new offspring. The knee points are helpful to improve the search ability of the algorithm. The solutions are decomposed and merged into non-dominated fronts using environmental selection strategy. The knee adaptation strategy is used to decompose the many-objective optimization problem into a sub-problems for handling irregular Pareto front.

The performance of KnRVEA has been evaluated on thirteen well established benchmark test problems with respect to two performance measures namely Inverted Generational Distance (*IGD*) and Hypervolume (*HV*).

The results have been compared with five well-known techniques such as Reference Vector Guided Evolutionary Algorithm (RVEA) [14], Hypervolume Estimation Algorithm (HypE) [13], Niche-elimination Operation based Non-dominated Sorting Genetic Algorithm (NSGA-III-NE) [56], Improved many-objective Optimization Algorithm based on Decomposition (I-MOEA/D) [44], and Knee Point Driven Evolutionary Algorithm (KnEA) [31]. KnRVEA has further been applied on three real-life many-objective problems.

The rest of the paper is structured as follows: Section 2 presents the related work done in the field of many-objective optimization techniques. Brief description of KnEA and RVEA is presented in Section 3. Section 4 presents motivation and description of proposed KnRVEA algorithm.

Section 5 discusses benchmark test problems, experimental setup, and results. Section 6 discusses the performance of KnRVEA on three real-life problems followed by conclusions in Section 7.

2 Related work

In the last few decades, the number of the evolutionary algorithms have been developed to solve multi-objective optimization problems. These algorithms are broadly classified into two categories such as decomposition based algorithms and performance indicator based algorithms.

2.1 Decomposition based approaches

Ishibuchi and Murata [40] proposed a Multi-objective Genetic Local Search (MOGLS) algorithm which was further improved by Jaskiewicz [41]. MOGLS algorithm transformed the optimization problem into the collection of weighted Tchebycheff functions. However, the performance of MOGLS is greatly affected from recombination operator. The another popular algorithm is Non-dominated Sorting Genetic Algorithm 2 (NSGA-II) [9]. NSGA-II has been widely used in many real-life applications with small number of objectives. Hence, it is unable to solve the problems that have a large number of objectives. ϵ -NSGA-II utilized the concepts of NSGA-II and epsilon-dominance for solving multi-objective problems [16, 18]. Bi and Wang [56] proposed an improved version of NSGA-III based on niche-elimination operation. In the proposed algorithm, an adaptive penalty distance (APD) function is presented to consider the importance of convergence and diversity in the different stages of the evolutionary process. MOEA/D [23] used a decomposition approach to divide the optimization problem into a number of sub-problems. These problems are further optimized using an evolutionary approach. Li et al. [39] proposed an extended version of MOEA/D which is based on the combination of dominance and decomposition approaches. However, the performance of MOEA/DD is greatly affected from neighborhood size and selection probability. Zheng et al. [44] proposed a new decomposition based approach that uses weighted mixture style method. It utilized two approaches namely weighted sum decomposition and Tchebycheff decomposition. This approach is known as I-MOEA/D that improves the effectiveness of the algorithm.

Hadka and Reed [19, 20] proposed a multi-objective evolutionary algorithm for handling many-objective problems using auto-adaptive recombination operators to improve the search process. Asafuddoula et al. [25, 26] proposed a

evolutionary technique based on decomposition with adaptive epsilon (DBEA-Eps) that employs the reference directions to adjust the search space and maintain the balance between convergence and diversity. However, DBEA-Eps's performance is greatly depends upon input parameters and adaptive rules. Xiao et al. [49] proposed a novel immune dominance selection multi-objective optimization algorithm (IDSMOA) to solve multi-objective problems. The main idea of IDSMOA is to promote two populations, i.e., population B and population T to co-evolve through immune selection, immune clone, immune gen, and memory selection operators to produce a satisfying Pareto front.

Deb and Jain [24] proposed NSGA-III algorithm which utilizes a reference point based approach for many-objective optimization. These reference points are same as the population size in which each population member is assisted with a reference point. There are number of improved versions has been proposed for NSGA-III. Jain and Deb [28] proposed an adaptive-NSGA-III (A-NSGA-III) technique that has an ability to add and remove the reference points. The same authors also presented an improved version of adaptive-NSGA-III algorithm (A²-NSGA-III) to remove the limitations of A-NSGA-III [27] with improvements in replacement strategy of reference points. Another extension of NSGA-III is the unified evolutionary algorithm (U-NSGA-III) [29] which is able to solve single-objective, multi-objective, and many-objective optimization problems.

Zhang et al. [31] proposed a knee point driven evolutionary algorithm (KnEA) which adopts a neighbor punishment density estimation strategy. An adaptive strategy was proposed for identifying the knee points in a small neighbourhood. The purpose of this strategy is to combine the parent and offspring populations to improve the convergence and promote diversity. The main drawback of KnEA is that the knee points are sensitive towards the given problem. Recently, Roy et al. [51] proposed an Evolutionary Path Control Strategy (EPCS) based algorithm to handle many-objective optimization problems. A reference vector strategy was introduced to reduce the number of non-dominated solutions. A novel fitness assignment procedure was proposed for survival selection which uses classical methods, aggregation methods or combination of these two methods. The main advantages of EPCS algorithm are low computational complexity and better diversity. Cheng et al. [14] proposed a Reference Vector Guided Evolutionary Algorithm (RVEA) for many-objective optimization problems. In RVEA, Angle Penalized Distance (APD) is used to balance both convergence and diversity. APD used adaptive strategy to adjust the reference vectors that ensure uniform distribution of solution in the given search space. Therefore, RVEA is able to handle irregular Pareto fronts.

2.2 Indicator based approaches

Zitzler and Künzli [43] proposed an Indicator-based Evolutionary Algorithm (IBEA) which utilizes a binary performance indicator in selection process. The main drawback of IBEA is high computational cost. There are several extended versions of IBEA such as IBMOLS that uses local search operator [45]. Wagner et al. [46] reported that IBEA algorithm is able to solve MaOPs. Emmerich et al. [47] proposed S Metric Selection based Evolutionary Multi-objective Algorithm (SMS-EMOA) that uses non-dominated sorting as a ranking criterion. They used hypervolume indicator to remove the individuals that have lowest hypervolume. The computational cost of S-Metric is high for large number of objective functions. Bader and Zitzler [13] proposed a fast hypervolume indicator based evolutionary algorithm known as Hypervolume Estimation Algorithm (HypE). The main concept of behind this algorithm is the ranking of solutions done through hypervolume. It uses Monte-Carlo simulation to estimate the hypervolume values. Three main components of HypE are mating selection, variation, and environmental selection. The binary tournament selection is proposed to select the offsprings. The variation used mutation and recombination operators to produce offspring. In environmental selection, the parent and children population are combined and decomposed into non-dominated fronts. Thereafter, the best fronts are included in new population. HypE suffers from overhead of maintaining necessary information. Bringmann et al. [21] proposed a Approximation-Guided Evolution (AGE) technique which uses the concept of approximation. However, it suffers from high computational cost. To reduce this problem, Wagner et al. [22] proposed fast approximation-guided evolutionary algorithm (AGE-II). AGE-II used archive for storing additive ϵ -approximation of non-dominated solutions. Gomez and Coello Coello [48] introduced a Many-objective Metaheuristic based on R2 Indicator (MOMBI) algorithm. The fundamental concept behind this algorithm is utility functions. These are used to group the solutions and provide them rank. The whole process is run until each member of the population is ranked. The ranking procedure is done by R2-ranking strategy. They used binary tournament selection to select the individuals based on their ranks. However, it suffers from loss of diversity for high dimensionality problems. Wang et al. [30] proposed an improved version of two-archive algorithm known as Two-Arch² for handling many-objective optimization. They used Convergence Archive (CA) and Diversity Archive (DA). $I_{\epsilon+}$ indicator and Pareto dominance concepts are used to improve the convergence and diversity. However, it does not provide optimal solution

for some WFG test instances because it does not consider extreme points.

3 Background

The basic concepts of Multi-objective Problems (MOPs) are presented in this section. Thereafter, the brief description of KnEA and RVEA algorithms are presented.

3.1 Basic concepts

Multi-objective Optimization Problems (MOPs) involve more than one objective which is to be optimized. The MOPs can be formulated as follows [17]:

$$\text{Minimize} : F(z) = [f_1(z), f_2(z), \dots, f_n(z)] \tag{1}$$

$$\begin{aligned} \text{Subject to:} \\ g_i(z) \geq 0, i = 1, 2, \dots, m \end{aligned} \tag{2}$$

$$h_i(z) = 0, i = 1, 2, \dots, p \tag{3}$$

where $z = [z_1, z_2, \dots, z_k]^T$ is the vector of decision variables, m is the number of inequality constraints, p is the number of equality constraints, g_i is the i^{th} inequality constraints, h_i is the i^{th} equality constraints, and n is the number of objective functions $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$.

There is a possibility that an optimal solution may not optimize all the objectives simultaneously. Whereas, Pareto optimal solutions representing the trade-offs between the objectives. In objective search space, the Pareto solutions are known as Pareto front (PF) and these solutions are known as Pareto set (PS) [55, 57] in the decision space.

Definition 1 Pareto Dominance: Assume there are two vectors such as: $y=(y_1, y_2, \dots, y_r)$ and $x=(x_1, x_2, \dots, x_r)$. Vector y is said to dominate vector x (such as $y \succ x$) if and only if:

$$\begin{aligned} \forall i \in \{1, 2, \dots, r\}, \\ [f(y_i) \geq f(x_i)] \wedge [\exists i \in 1, 2, \dots, r : f(y_i)] \end{aligned} \tag{4}$$

Definition 2 Pareto Optimality:

A solution $x \in X$ is called Pareto optimal if and only if:

$$\nexists x \in X | f(x) \succ f(y) \tag{5}$$

Definition 3 Pareto optimal set:

The set of all Pareto-optimal solutions including all non-dominated solutions of a problem is called Pareto optimal set if and only if:

$$P_s = \{y, x \in X | \nexists f(x) \succ f(y)\} \tag{6}$$

Definition 4 Pareto optimal front: A set containing the objective values corresponding to Pareto optimal solutions in Pareto optimal set is called Pareto optimal front and it is defined as follows:

$$P_f = \{f(y) | y \in P_s\} \tag{7}$$

Definition 5 Knee point:

It is defined as the Pareto-optimal point that have maximum reflex angle computed from its neighbors. It means that if small improvement is done in one objective, then there will be severe degradation in at least one other objective [37]. Figure 1 shows the knee points and boundary points.

3.2 Knee Point Driven Evolutionary Algorithm (KnEA)

Multi-objective Evolutionary Algorithms (MOEAs) suffers from two major problems. These speed up the convergence performance and enhance the diversity. To resolve these problems, several evolutionary algorithms have been proposed. Knee Point Driven Evolutionary Algorithm (KnEA) is a recently developed many-objective evolutionary algo-

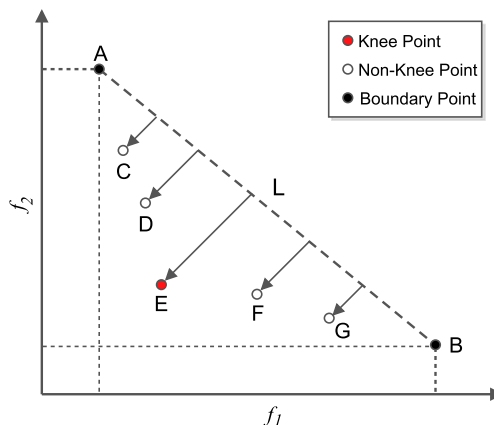


Fig. 1 Relation between Knee point and Boundary point

rithm [31]. The knee points are used as secondary selection criteria to improve the search ability in MOEAs. The main components of KnEA are described as follows:

- An initial population is randomly generated of size N .
- A binary tournament selection strategy is used to select the individuals and generate N number of individual offsprings using variation method. The selection strategy adopts three tournament metrics such as dominance relationship, knee point, and a weighted distance measure.
- The non-dominated sorting method is applied on population to identify the solutions located in knee regions.
- An environmental selection strategy is performed to select N number of individuals.

The above-mentioned components of KnEA are described in Algorithm 1.

Algorithm 1 KnEA

Input: Population P , Population size N , Set of knee points K , Rate of knee points T , Initial iteration $\leftarrow 0$, Maximum number of iteration $Max_{iteration}$
Output: Non-dominated solutions P

```

1: procedure KnEA
2:    $K \leftarrow \phi$ 
3:    $P \leftarrow Initialize(N)$ 
4:   while  $iteration < Max_{iteration}$  do
5:      $P' \leftarrow MatingSelection(P, K, N)$ 
6:      $P \leftarrow P \cup Variation(P', N)$ 
7:      $F \leftarrow NondominatedSort(P)$ 
8:      $[k, r, t] \leftarrow FindingKneePoint(F, T, r, t)$ 
9:      $P \leftarrow EnvironmentalSelection(F, K, N)$ 
10:     $iteration \leftarrow iteration + 1$ 
11:  return  $P$ 

```

The mating selection in KnEA uses binary tournament selection strategy that comprises of three tournament strategies such as dominance comparison, knee point criterion, and weighted distance (see Algorithm 2). Two individuals are randomly selected from the given population. If one solution dominates the other, then the dominated solution is selected. If both solutions are non-dominated, then determine whether they are knee points or not. If one of them is knee point, then the knee points is selected. If both of these are knee points or neither of them is a knee point, then

a weighted distance method is used to compare the solutions. A weighted distance DW is used to select a winner solution if neither the knee point criterion nor dominance comparison can differentiate the two solution participated in selection strategy [31]. An efficient non-dominated sorting (ENS) algorithm [61] is used to sort the non-dominated solutions present in the population.

Algorithm 2 Matingselection(P,K,N)

Input: Population P , Knee points K , Population size N
Output: L

```

1:  $L \leftarrow \phi$ 
2: while  $|L| < N$  do
3:   Choose the value of  $a$  and  $b$  from  $P$ 
4:   if  $a < b$  then
5:      $L \leftarrow L \cup \{a\}$ 
6:   else if  $b < a$  then
7:      $L \leftarrow L \cup \{b\}$ 
8:   else
9:     if  $a \in K$  and  $b \notin K$  then
10:       $L \leftarrow L \cup \{a\}$ 
11:     if  $a \notin K$  and  $b \in K$  then
12:       $L \leftarrow L \cup \{b\}$ 
13:     else
14:       if  $DW(a) > DW(b)$  then
15:          $L \leftarrow L \cup \{a\}$ 
16:       else if  $DW(a) < DW(b)$  then
17:          $L \leftarrow L \cup \{b\}$ 
18:       else
19:         if  $rand() < 0.5$  then
20:            $L \leftarrow L \cup \{a\}$ 
21:         else
22:            $L \leftarrow L \cup \{b\}$ 
23:         end if
24:       end if
25:     end if
26:   end if
27: end while
28: return  $L$ 

```

An adaptive strategy is used to find knee points present in the population. Algorithm 3 describes the main steps of adaptive strategy for finding knee points. The same procedure is repeated for all non-dominated fronts in the combined population until knee points are detected for all non-dominated fronts.

Algorithm 3 FindingKneePoint(F,T,r,t)

Input: Sorted population F , Knee points rate T , Adaptive parameters r, t
Output: K, r, t

- 1: $K \leftarrow \phi$
- 2: **for each** $F_i \in F$ **do**
- 3: $E \leftarrow$ Find extreme solution from the set of solution F_i
- 4: $L \leftarrow$ Compute extreme hyperplane from E
- 5: Update the value of r
- 6: $f_{mx} \leftarrow$ Maximum value of objectives in F_i
- 7: $f_{mn} \leftarrow$ Minimum value of objectives in F_i
- 8: Compute the value of R
- 9: Compute the distance between solutions in F_i and L
- 10: Sort population F_i in a descending order
- 11: Compute the size of population as $S_{F_i} \leftarrow |F_i|$
- 12: **for each** $p \in F_i$ **do**
- 13: $NB \leftarrow \{a \in F_i \rightarrow |f_a^j - f_p^j| \leq R^j, 1 \leq j \leq M\}$
- 14: $K \leftarrow K \cup \{p\}$
- 15: $F_i \leftarrow F_i / NB$
- 16: **end for**
- 17: $t \leftarrow |K| / S_{F_i}$
- 18: **end for**
- 19: **return** K, r , and t

Environmental selection process is used to select fitter solutions as parents for the next generation. KnEA selects parents for next generation from the combination of parent and offspring populations of this generation. The main steps of environmental selection in KnEA are presented in Algorithm 4.

Algorithm 4 Environmentalselection(F,K,N)

Input: Sorted population F , Set of knee points K , Population size N
Output: L

- 1: $L \leftarrow \phi$
- 2: $L \leftarrow F_1 \cup \dots \cup F_{j-1}$
- 3: $L \leftarrow L \cup (K \cap F_j)$
- 4: **if** $|L| > N$ **then**
- 5: Discard $|L| - N$ solutions from L
- 6: **if** $|L| < N$ **then**
- 7: Add $N - |L|$ solutions to L
- 8: **end if**
- 9: **return** Q

3.3 Reference Vector Guided Evolutionary Algorithm (RVEA)

Reference Vector Guided Evolutionary Algorithm (RVEA) is a recently developed evolutionary algorithm for solving many-objective optimization problems [14]. The main components of RVEA are offspring creation, reference vector guided selection, and reference vector adaptation. The genetic operators are used to generate the offspring population. The reference vector guided selection has four sub-steps such as objective value translation, population partitions, computation of Angle Penalized Distance (APD), and elitism selection [14]. It adopts a set of reference vectors for their selection strategy. In RVEA, Angle Penalized Distance (APD) is used to balance the both convergence and diversity dynamically. It uses adaptive strategy to adjust the reference vectors that ensure uniform distribution of solution in the search space. The preference articulation approach is used to generate the distributed Pareto optimal solution in a search space. Algorithm 5 describes the main components of RVEA [14].

Algorithm 5 RVEA

Input: Set of reference vectors $V_0 \leftarrow \{v_1, v_2, v_3, \dots, v_N\}$, Maximum number of iteration $Max_{iteration}$
Output: Non-dominated solutions P

- 1: **procedure** RVEA
- 2: Initialize the population P and set counter $c \leftarrow 0$
- 3: **while** $c < Max_{iteration}$ **do**
- 4: Create offspring O_c from parent population P_c
- 5: Combine the parent and offspring populations, ($M_c \leftarrow P_c \cup O_c$)
- 6: $P_{c+1} \leftarrow$ ReferenceVectorGuidedSelection(c, P_c, V_c)
- 7: $V_{c+1} \leftarrow$ ReferenceVectorAdaptation(c, P_{c+1}, V_c, V_0)
- 8: $c \leftarrow c + 1$
- 9: **return** P

3.3.1 Reference vector guided selection

Reference vector guided selection strategy consists of four main steps. These are objective value translation, population partition, angle penalized distance calculation, and elitism selection. The objective values of individuals in population are translated into RVEA to guide the search process. To

generate a uniformly distributed set of reference vectors, a set of uniformly distributed reference points is generated on a unit hyperplane using the canonical simplex lattice design method [58].

3.3.2 Assignment of individuals to reference vectors

After the generation of the reference vectors, individuals are assigned to them. An individual is assigned to a reference vector if and only if the angle between it and the reference vector is minimum among all reference vectors. In this way, assignment of individuals to the reference vectors partitions the population into sub-populations.

3.3.3 Selection mechanism in each sub-population

After the generation of reference vectors and the assignment of individuals then, one individual is selected from each sub-population. The selection criterion consists of two sub-criteria that are meant for managing convergence and diversity, respectively. Convergence is taken care of by the distance between the translated objective vector and the origin. Diversity is accounted by the angle between the translated objective vector and the reference vector. The individual with the smallest angle is preferred over other individuals.

3.3.4 Adaptation of reference vectors

For some optimization problems where objective functions are scaled to different ranges, a uniformly distributed set of reference vectors is not best suited as shown in Fig. 2.

To tackle this issue, one possible solution is to adapt the reference vectors. In RVEA, reference vectors are adaptive.

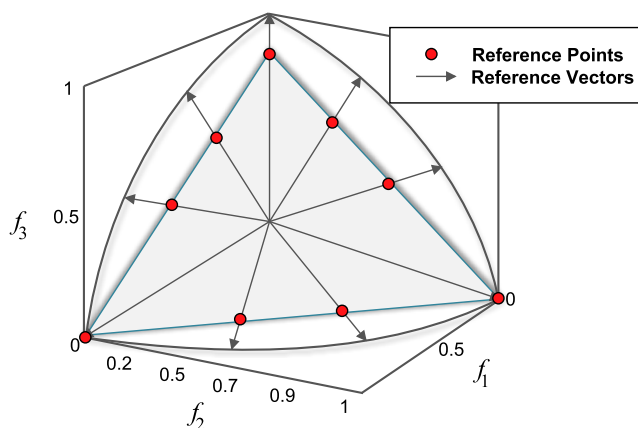


Fig. 2 Uniformly distributed reference vectors in 3D search space. In this case, nine uniformly distributed reference points are generated and then mapped to generate the nine reference vectors

In other words, they change their position according to the location of individuals in the objective space.

4 Proposed Knee Point and Reference Vector Adaptation based Evolutionary Algorithm (KnRVEA)

This section first describe the motivation behind proposed algorithm followed by a detailed description of proposed algorithm.

4.1 Motivation

The main contribution of this paper is to propose a hybrid many objective optimization algorithm that utilizes the knee points and reference vector adaption strategies. The primary intention to combine these strategies is to minimize the each other's weakness and promote respective strengths. It is worth mentioning that the knee points are designated for exploratory search and reference vector adaption strategy is assigned for uniform distribution of solutions even the objective functions are not normalized same range. In the proposed algorithm, only reference vector adaption strategy is used for uniform distribution of knee points. The collaboration between knee point and reference vector adaption strategy enhances the exploratory search capability such that reference vector adaptation strategy tune the knee points according to the distribution of candidate solutions which makes the exploration process more directive. Hence, a novel knee adaption strategy is proposed which is inspired from reference vector adaption strategy.

4.2 Proposed algorithm

The basic procedure of proposed KnRVEA algorithm is described in Algorithm 6. KnRVEA algorithm is inspired from knee points and reference vector adaptation strategies. KnRVEA consists of six main steps such as mating selection, variation, non-dominated sorting, identifying knee points, environmental selection, and knee adaptation strategy. Initially, parent population P is randomly initialized with N number of individuals. The mating selection procedure is used to select the promising solutions in a given search space. The variation uses simulated binary crossover (SBX) [59] and polynomial mutation [60] operators to generate N offspring individuals. Non-dominated sorting is applied on both parent and offspring population followed by a search strategy to identify knee points in each non-dominated front. The environmental selection is used to select N individuals. Finally, knee adaptation strategy is performed to obtain uniformly distributed Pareto optimal solutions set.

Algorithm 6 KnRVEA

Input: Population P , Population size N , Set of knee points K , Rate of knee points T , Initial iteration $\leftarrow 0$, Set of knee points $V_0 \leftarrow \{v_1, v_2, v_3, \dots, v_N\}$, Control parameter F_c , Maximum number of iteration $Max_{iteration}$

Output: Non-dominated solutions P

```

1: procedure KnRVEA
2:    $K \leftarrow \phi$ 
3:    $P \leftarrow Initialize(N)$ 
4:   while  $iteration < Max_{iteration}$  do
5:      $P' \leftarrow MatingSelection(P, K, N)$ 
6:      $P \leftarrow P \cup Variation(P', N)$ 
7:      $F \leftarrow NondominatedSort(P)$ 
8:      $[k, r, t] \leftarrow FindingKneePoint(F, T, r, t)$ 
9:      $P \leftarrow EnvironmentalSelection(F, K, N)$ 
10:     $V_{c+1} \leftarrow KneeAdaptation(iteration, P,$ 
       $V_{iteration}, V_0, F_c)$ 
11:     $iteration \leftarrow iteration + 1$ 
12:  return  $P$ 

```

4.2.1 Binary tournament mating selection

There are three tournament selection strategies are employed in KnEA algorithm such as dominance comparison, knee point criterion and a weighted distance.

The two random solutions are selected in binary tournament selection. If one solution dominates the other one then the former solution is selected. If both of these two solutions are non-dominated with each other, then the proposed algorithm check if they are knee points or not. If only one solution is knee point, then the knee point is chosen. If both of these solutions are knee point or none of them is a knee point then a weighted method is used for comparing these solutions. The solution which has the large weighted distance can win the tournament. If both of these solutions have the same distance then the solution will be chosen randomly for reproduction.

An important component of KnEA algorithm is a weighted distance measurement for choosing a winning solution in the binary tournament mating selection if both the dominance comparison and the knee points criterion are not able to distinguish the two solutions in the tournament. The weighted distance of a solution is described as follows based on the k -nearest neighbors:

$$DW(p) = \sum_{i=1}^k w_{p_i} d_{pp_i} \tag{8}$$

$$w_{p_i} = \frac{r_{p_i}}{\sum_{i=1}^k r_{p_i}} \tag{9}$$

$$r_{p_i} = \frac{1}{|d_{pp_i} - \frac{1}{k} \sum_{i=1}^k d_{pp_i}|} \tag{10}$$

where p_i indicates the i th neighbor of population p , w_{p_i} represents the weight of p_i , d_{pp_i} is the Euclidean distance between p and p_i , and r_{p_i} represents the rank of distance d_{pp_i} .

4.2.2 Variation

The proposed approach uses simulated binary crossover (SBX) [59] and polynomial mutation [60] operators for variation to generate N offspring individuals. These operators are responsible for better exploration and exploitation between the individuals. The normal boundary insertion method is used for offsprings which lies outside the boundary [62].

4.2.3 Non-dominated sorting

KnRVEA performs non-dominated sorting to select the non-dominated solutions from the first front, i.e., $PF1$. If the number of solutions in $PF1$ is greater than the size of population N , then knee points in $PF1$ are firstly selected as parents for the next population. Let the number of knee points in $PF1$ be N_{PF1} . In case N_{PF1} is larger than N , then the knee points that have larger distance to the hyperplane are selected. Otherwise, N_{PF1} knee points are selected together with other solutions in $PF1$ that have a larger distance to the hyperplane of $PF1$.

4.2.4 Adaptive strategy for identifying knee points

In KnEA, an adaptive strategy is proposed for finding the knee points in a population. There is a extreme line L having the maximum of f_1 and f_2 functions. Then, the distance of each solution is identified as a knee point if its distance is maximum with respect to the extreme line (see Fig. 1). Here, L can be defined by $ax + by + c = 0$ for a bi-objective minimization problem. The solution is also considered as a knee point if there is only one solution in its neighbourhood. The distance from a solution $A(x_A, y_A)$ to extreme line L with more than two objectives is computed as follows:

$$d_s(A, L) = \begin{cases} \frac{|ax_A + by_A + c|}{\sqrt{a^2 + b^2}}, & \text{if } ax_A + by_A + c < 0 \\ -\frac{|ax_A + by_A + c|}{\sqrt{a^2 + b^2}}, & \text{otherwise} \end{cases} \tag{11}$$

A strategy to tune the size of neighborhood solutions is proposed in KnEA algorithm if all the solutions lie in a same neighbourhood. However, the proposed adaptive strategy is able to find the knee points in its neighbourhood.

4.2.5 Environmental selection

KnEA uses elitist approach to selects the parents for the next generation. KnEA prefers knee points rather than

the non-dominated solutions in environmental selection. It performs non-dominated sorting before environmental selection to select the non-dominated solutions in the first non-dominated front. If these solutions are larger than the population size, then knee points are selected first for the next generation of population.

4.2.6 Knee adaptation strategy

The proposed knee adaptation strategy utilizes the concept of reference vector adaptation [14] strategy to obtain a set of Pareto optimal solutions. This strategy is mentioned in Algorithm 7. These solutions are the points that are intersection between knee points and Pareto front. However, these knee points will not produce uniformly distributed solutions when the objective function values are normalized in the same range. To remove this conflict, the range of the knee points will be adapted as follows:

$$v_{k+1,j} = \frac{v_{0,j} \circ (O_{k+1}^{max} - O_{k+1}^{min})}{\|v_{0,j} \circ (O_{k+1}^{max} - O_{k+1}^{min})\|} \tag{12}$$

where $j = 1, 2, \dots, N$, $v_{k+1,j}$ represents the j^{th} knee point adapted for generation $k + 1$, $v_{0,j}$ represents the j^{th} uniformly distributed knee point. O_{k+1}^{min} and O_{k+1}^{max} are the minimum and maximum values of objective function, respectively. The \circ operator indicates the Hadamard product that element-wise multiply the two knee points of same size [14]. The knee adaptation strategy has obtained the uniformly distributed solutions even if the objective functions are not normalized.

Algorithm 7 Knee adaptation strategy

Input: Generation index k , Population P , Current unit knee point set $V_k \leftarrow \{v_{k,1}, v_{k,2}, \dots, v_{k,N}\}$, Initial unit knee point set $V_0 \leftarrow \{v_{0,1}, v_{0,2}, \dots, v_{0,N}\}$, Control parameter F_c

Output: Next generation knee point set V_{k+1}

- 1: **procedure** KNEEADAPTATION (k, P, V_k, V_0, F_c)
 - 2: **if** $\left(\frac{k}{Max_{iteration}} \bmod F_c == 0\right)$ **then**
 - 3: Compute the minimal O_{k+1}^{min} and maximal O_{k+1}^{max} objective values, respectively
 - 4: **for** $j = 1$ to N **do**
 - 5: $v_{k+1,j} \leftarrow \frac{v_{0,j} \circ (O_{k+1}^{max} - O_{k+1}^{min})}{\|v_{0,j} \circ (O_{k+1}^{max} - O_{k+1}^{min})\|}$
 - 6: **end for**
 - 7: **else**
 - 8: $v_{k+1,j} \leftarrow v_{k,j}$
 - 9: **end if**
-

There is a controlling parameter F_c that control the frequency of employing the adaptation strategy. It ensures that knee adaptation strategy have to be performed in

selected generation because there is no need to employ this strategy very frequently [52]. Example 2 illustrates the workings of control parameter F_c in adaptation strategy.

Example 2: (Control parameter) For instance, the value of F_c is set to 0.4. The knee point will be adapted only at $k = 0, k = 0.4 \times Max_{iteration}, k = 0.8 \times Max_{iteration}, k = 0.12 \times Max_{iteration}$ and so forth.

4.3 Computational complexity

In this subsection, the computational complexity of proposed algorithm is discussed. Both time and space complexities of the proposed algorithm are given below.

4.3.1 Time complexity

1. Initialization of KnRVEA population needs $\mathcal{O}(M \times N)$ time where M and N represent the number of objectives and population size, respectively.
2. Mating selection procedure requires $\mathcal{O}(M \times N \times N)$, i.e., $\mathcal{O}(M \times N^2)$ time.
3. Variation procedure uses $\mathcal{O}(N)$ time.
4. Non-dominated sorting procedure requires $\mathcal{O}(MN^2)$.
5. Knee point identification requires $\mathcal{O}(MN^2)$.
6. Environmental selection needs $\mathcal{O}(N \log N)$.
7. Knee adaptation strategy requires $\mathcal{O}(MN/F_c \times Max_{iteration})$, where F_c represents control parameter and $Max_{iteration}$ represents the maximum number of iterations.

Therefore, summing up the complexities of all the above steps and the total time complexity of KnRVEA for maximum number of generations is $\mathcal{O}(M \times N^2 \times Max_{iteration})$.

4.3.2 Space complexity

The space complexity of KnRVEA algorithm is the maximum amount of space which is considered at any one time during its initialization process. Thus, the total space complexity of KnRVEA algorithm is $\mathcal{O}(M \times N)$.

5 Experimentation and results

In order to demonstrate the effectiveness of proposed KnRVEA, thirteen benchmark test problems have been used for experimentation which are taken from two well-known test suites such as DTLZ [32] and WFG [33]. The performance of KnRVEA has been compared with five well-established algorithms such as Reference Vector Guided Evolutionary Algorithm (RVEA) [14], Hypervolume Estimation Algorithm (HypE) [13], Niche-elimination

Table 1 The characteristics of benchmark test problems

Problems	Objectives	Properties
<i>DTLZ1</i>	3, 5, 10, 15	Linear, multi-modal
<i>DTLZ2</i>	3, 5, 10, 15	Concave
<i>DTLZ3</i>	3, 5, 10, 15	Concave, multi-modal
<i>DTLZ4</i>	3, 5, 10, 15	Concave, biased
<i>WFG1</i>	3, 5, 10, 15	Mixed, flat biased
<i>WFG2</i>	3, 5, 10, 15	Convex, disconnected, non-separable
<i>WFG3</i>	3, 5, 10, 15	Linear, degenerate, non-separable
<i>WFG4</i>	3, 5, 10, 15	Concave, multimodal
<i>WFG5</i>	3, 5, 10, 15	Concave, deceptive
<i>WFG6</i>	3, 5, 10, 15	Concave, nonseparable
<i>WFG7</i>	3, 5, 10, 15	Concave, parameter dependant biased
<i>WFG8</i>	3, 5, 10, 15	Concave, non-separable, parameter dependant biased
<i>WFG9</i>	3, 5, 10, 15	Concave, non-separable, deceptive, parameter dependant biased

Operation based Non-dominated Sorting Genetic Algorithm (NSGA-III-NE) [56], Improved many-objective Optimization Algorithm based on Decomposition (I-MOEA/D) [44], and Knee Point Driven Evolutionary Algorithm (KnEA) [31]. The results are evaluated and compared with two well-known performance measures such as Inverted Generational Distance (*IGD*) [36] and Hypervolume (*HV*) [38].

5.1 Benchmark test problems

All the algorithms have been tested over four DTLZ test problems (DTLZ1-DTLZ4) [32] and nine WFG test problems (WFG1-WFG9) [33]. For each benchmark test suite, the number of objectives is to be varied from 3 to 15, i.e., {3, 5, 10, 15}. Table 1 represents the description of these test problems. According to [32], the decision variables are set to $n = m + r - 1$ for DTLZ test problems. The value of r is fixed for DTLZ1 ($r = 5$) and for DTLZ2-DTLZ4 ($r = 10$). The decision variables are set to $n = k + l$ for WFG test problems, where variable $k = 2 \times (m - 1)$ and l is set to 20 as suggested by [33].

5.2 Parameter setting

The parameters of the above-mentioned algorithms are set as they are recommended in their original work. The parameters setting of these algorithms are reported in Table 2. In addition to these parameters, the population size and maximum iterations for all above-mentioned algorithms are set to 100 and 3000, respectively. Due to the stochastic nature of these algorithms, the results are averaged over 30 independent runs under 30 random seeds. The mean and median solutions obtained in final iteration are reported in tables. The simulations are carried out in Matlab R2017a environment operating on Core i5 processor and 2.40 GHz processor with 4 GB RAM.

Table 2 Parameters values of algorithms

#	Algorithms	Parameters	Values
1	<i>KnRVEA</i>	Simulated Binary Crossover probability	1
		Polynomial mutation probability	1/n
		Distribution index of crossover operator (n_c)	20
		Distribution index of mutation operator (n_m)	20
		Threshold (T)	0.5
1	<i>RVEA</i>	Simulated Binary Crossover probability	1
		Polynomial mutation probability	1/n
		Distribution index of crossover operator (n_c)	20
		Distribution index of mutation operator (n_m)	20
2	<i>HypE</i>	Sampling points	10,000
3	<i>NSGA-III-NE</i>	Simulated Binary Crossover probability	1
		Polynomial mutation prob.	1/n
		Distribution index of crossover operator (n_c)	30
		Distribution index of mutation operator (n_m)	20
4	<i>I-MOEA/D</i>	Simulated Binary Crossover probability	1
		Polynomial mutation prob.	1/n
		Distribution index of crossover operator (n_c)	20
		Distribution index of mutation operator (n_m)	20
5	<i>KnEA</i>	Threshold (T)	0.5

5.3 Experimentation 1: Performance evaluation on DTLZ test suite

Tables 3 and 4 show the values of *IGD* and *HV* obtained from above-mentioned six algorithms on DTLZ test suite with 3, 5, 10, and 15 objectives (i.e., 16 test instances). It can be seen from these tables that KnRVEA provides better values of *IGD* and *HV* for DTLZ1 test instances. After the proposed KnRVEA, RVEA and I-MOEA/D provide the better value of *IGD* and *HV* than the others. RVEA and I-MOEA/D are the second best performing algorithms. NSGA-III-NE performs almost similar to HypE algorithm in all the test instances of DTLZ1.

For DTLZ2 test instances, KnRVEA provides better value of *IGD* for 3- and 15-objective test instances. After the proposed KnRVEA, RVEA provides better results than the other algorithms for these test instances. For 5- and 10-objective test instances, KnRVEA is the second best algorithm in terms of *IGD*. For these instances, RVEA provides better results than KnRVEA. For *HV*, KnRVEA provides better results for all test instances of DTLZ2. While, the performance of RVEA and HypE algorithms is similar to each other for *HV* measure.

For DTLZ3 test instances, KnRVEA is significantly better than other competitor algorithms in terms of *IGD* and *HV*. For *IGD*, I-MOEA/D and NSGA-III-NE are the second

Table 3 The IGD values obtained by proposed KnRVEA and other competitor algorithm in DTLZ test suite

F	M	Analysis	KnRVEA	RVEA	HypE	NSGA-III-NE	I-MOEA/D	KnEA
DTLZ1	3	Mean	4.81E-05	4.69E-04	1.53E-04	4.90E-04	3.20E-04	4.78E-03
		Median	5.09E-05	4.99E-04	2.08E-01	1.40E-03	5.90E-04	2.31E-04
	5	Mean	2.72E-05	3.11E-05	1.85E-01	5.25E-04	2.65E-04	1.32E-03
		Median	3.17E-05	3.57E-05	2.20E-01	9.91E-04	2.96E-04	7.00E-02
	10	Mean	6.19E-05	1.50E-04	1.50E-01	2.25E-03	1.90E-03	4.47E-03
		Median	2.29E-04	6.69E-04	1.92E-01	3.53E-03	2.26E-03	5.11E-03
	15	Mean	2.00E-05	1.77E-04	1.82E-01	2.95E-03	2.90E-03	3.11E-03
		Median	1.14E-04	2.04E-04	2.60E-01	5.15E-03	4.25E-03	1.50E-06
DTLZ2	3	Mean	1.01E-04	1.71E-04	6.80E-02	1.35E-03	6.81E-04	4.12E-02
		Median	1.10E-04	1.90E-04	6.98E-02	1.42E-03	8.10E-04	3.81E-02
	5	Mean	1.03E-03	1.11E-04	2.80E-01	4.35E-03	1.18E-03	2.32E-03
		Median	1.91E-03	2.01E-04	2.92E-01	5.15E-03	1.40E-03	4.60E-04
	10	Mean	1.39E-03	1.11E-03	6.90E-01	1.40E-02	3.30E-03	1.15E-02
		Median	2.11E-02	2.17E-03	7.01E-01	1.61E-02	3.81E-03	4.62E-03
	15	Mean	2.93E-04	5.03E-04	6.35E-01	1.40E-02	4.60E-03	5.23E-03
		Median	3.67E-04	5.95E-04	7.62E-01	1.83E-02	5.90E-03	3.90E-03
DTLZ3	3	Mean	2.73E-05	1.11E-04	1.70E-02	9.80E-04	5.90E-04	2.35E-02
		Median	6.15E-05	2.20E-04	1.82E-02	4.11E-03	1.95E-03	2.79E-02
	5	Mean	2.39E-04	1.33E-02	1.19E-02	3.15E-03	6.25E-04	2.01E-02
		Median	1.16E-03	1.87E-02	1.93E-02	6.01E-03	1.36E-03	2.94E-02
	10	Mean	1.15E-03	2.01E-01	1.86E-02	8.90E-03	1.95E-03	2.38E-02
		Median	2.18E-03	2.71E-01	2.93E-02	1.25E-02	2.35E-03	2.93E-02
	15	Mean	2.97E-04	1.98E-01	4.35E-02	1.43E-02	5.90E-03	1.11E-02
		Median	5.93E-03	2.90E-01	6.93E-02	2.20E-02	7.50E-03	2.04E-02
DTLZ4	3	Mean	1.03E-04	5.03E-04	8.49E-02	2.94E-04	1.10E-04	2.23E-02
		Median	1.60E-04	6.84E-04	9.40E-02	6.11E-04	1.71E-04	2.48E-02
	5	Mean	6.72E-05	1.00E-01	2.80E-01	9.90E-04	1.11E-04	1.04E-02
		Median	7.16E-05	1.69E-01	2.93E-01	1.37E-03	1.37E-04	1.54E-02
	10	Mean	1.02E-03	1.94E-03	6.81E-01	5.80E-03	1.32E-03	2.30E-02
		Median	1.67E-03	2.31E-03	6.93E-01	6.43E-03	1.70E-03	2.83E-02
	15	Mean	1.13E-04	1.96E-04	6.08E-01	7.87E-03	1.50E-03	3.02E-02
		Median	2.01E-04	2.31E-04	6.41E-01	3.11E-02	1.90E-03	3.73E-02

The obtained best results are in bold

Table 4 The HV values obtained by proposed KnRVEA and other competitor algorithm in DTLZ test suite

F	M	Analysis	KnRVEA	RVEA	HypE	NSGA-III-NE	I-MOEA/D	KnEA	
DTLZ1	3	Mean	7.55E-02	4.69E-05	4.98E-03	3.15E-04	4.81E-02	4.11E-02	
		Median	2.81E-01	4.13E-06	2.57E-05	3.00E-04	4.11E-03	3.50E-03	
	5	Mean	5.64E-02	4.42E-04	3.49E-05	3.73E-04	3.09E-02	2.00E-02	
		Median	2.55E-01	2.03E-05	2.99E-05	1.63E-04	2.00E-03	1.05E-02	
	10	Mean	4.55E-02	1.39E-03	3.08E-03	2.15E-04	1.49E-02	4.11E-03	
		Median	1.11E-01	1.69E-04	4.18E-04	1.15E-05	2.80E-03	1.79E-04	
	15	Mean	3.19E-04	6.69E-05	4.18E-05	2.15E-05	1.81E-06	3.97E-07	
		Median	9.75E-05	3.00E-06	3.19E-07	3.97E-07	1.03E-06	2.97E-07	
	DTLZ2	3	Mean	3.51E-02	2.69E-05	3.99E-03	4.05E-04	3.85E-02	2.90E-02
			Median	2.87E-01	3.03E-06	2.67E-05	2.97E-04	3.19E-03	3.57E-02
		5	Mean	3.66E-01	3.40E-04	2.09E-05	2.77E-04	2.91E-02	3.87E-02
			Median	1.96E-02	2.03E-05	1.09E-05	1.67E-04	2.01E-03	1.07E-02
10		Mean	3.05E-02	1.19E-03	3.09E-03	3.17E-04	1.57E-02	3.10E-03	
		Median	2.12E-03	1.59E-04	3.18E-04	1.05E-05	2.73E-03	2.70E-04	
15		Mean	4.10E-04	5.60E-05	3.85E-05	3.37E-05	3.01E-06	2.97E-07	
		Median	4.55E-05	3.97E-06	2.19E-07	3.06E-07	1.19E-06	2.00E-07	
DTLZ3		3	Mean	7.06E-02	5.57E-03	3.98E-03	2.05E-04	5.83E-02	2.90E-02
			Median	3.83E-03	4.13E-04	3.07E-04	3.78E-05	3.11E-04	3.97E-04
		5	Mean	3.61E-01	3.47E-02	2.39E-03	1.77E-03	2.49E-02	1.37E-02
			Median	1.50E-02	2.57E-03	2.00E-04	1.95E-04	2.33E-03	1.69E-03
	10	Mean	5.50E-01	2.30E-02	3.18E-03	4.09E-03	2.33E-02	3.03E-02	
		Median	2.31E-02	1.23E-03	4.26E-04	3.37E-04	2.49E-03	2.70E-04	
	15	Mean	3.11E-02	4.60E-03	3.08E-03	2.67E-03	2.98E-02	1.23E-03	
		Median	1.70E-03	3.01E-04	2.10E-04	3.92E-04	2.50E-05	2.67E-04	
	DTLZ4	3	Mean	3.71E-02	3.48E-03	3.00E-03	2.02E-03	3.69E-04	4.57E-05
			Median	2.11E-03	4.49E-04	4.07E-04	2.84E-05	1.66E-04	2.39E-05
		5	Mean	3.60E-02	4.24E-03	2.49E-04	4.58E-03	2.43E-03	3.47E-04
			Median	1.51E-03	3.01E-04	1.90E-05	4.61E-04	5.17E-04	6.69E-05
10		Mean	9.50E-02	2.30E-04	3.57E-03	3.79E-03	2.08E-04	3.96E-03	
		Median	1.63E-03	1.63E-04	4.10E-04	1.15E-05	2.81E-04	1.39E-04	
15		Mean	4.49E-03	6.17E-05	3.58E-04	2.99E-05	3.02E-04	3.43E-05	
		Median	5.05E-04	1.00E-05	3.10E-06	4.69E-07	1.13E-05	2.90E-06	

The obtained best results are in bold

and third best algorithm after KnRVEA, respectively. For HV performance measure, NSGA-III-NE is the second best algorithm after KnRVEA.

For DTLZ4 test instances, the results obtained from KnRVEA are better than the competitor algorithms in terms of IGD and HV. The I-MOEA/D and NSGA-III-NE are the second and third best algorithms after KnRVEA for IGD and HV performance measures.

It has been observed from the results presented in Tables 3 and 4 that KnRVEA is able to find the optimal solution for most of test instances. For IGD, KnRVEA provides best results on 12 out of 16 test instances. For HV values, KnRVEA provides best results on all test instances.

For the visualization of solutions distribution, Figs. 3 and 4 show the non-dominated fronts obtained from KnRVEA and other competitor algorithms for 15- and 3-objective DTLZ test functions, respectively. For 15-objective DTLZ test functions, RVEA and I-MOEA/D provide uniform convergence but fails to reach some regions of Pareto front. HypE, NSGA-III-NE, and KnEA provide good convergence over all Pareto front. It can be seen from Fig. 3 that the non-dominated front obtained from KnRVEA provides better convergence and diversity than the other competitor algorithms. For 3-objective DTLZ test functions, it is observed that the Pareto front obtained from KnRVEA, RVEA, NSGA-III-NE, and I-MOEA/D has uniform convergence

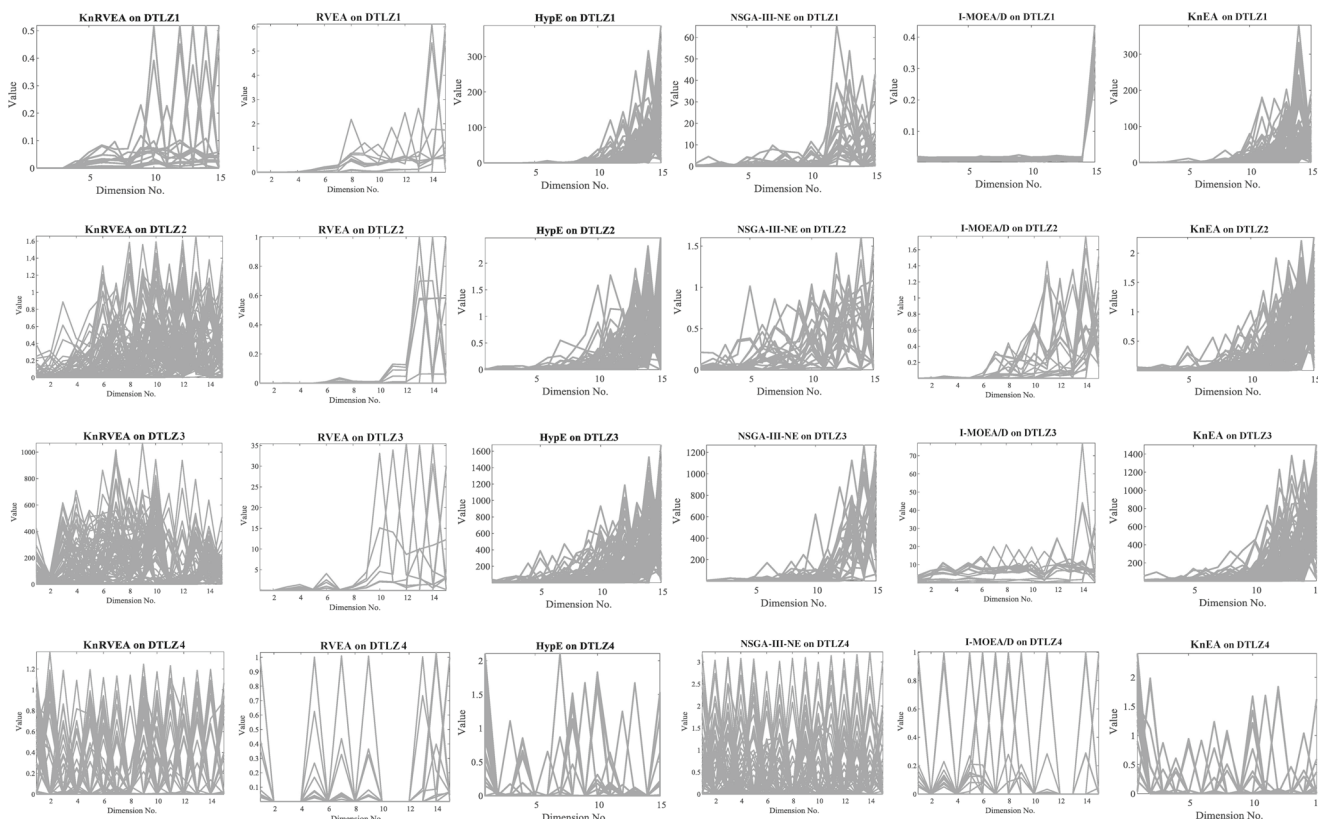


Fig. 3 The non-dominated fronts obtained from six algorithms on 15-objectives for DTLZ1, DTLZ2, DTLZ3, and DTLZ4 test instances

and better diversity (see Fig. 4). However, HypE and KnEA fail to provide a good coverage over the Pareto front.

5.4 Experimentation 2: Performance evaluation on WFG test suite

Table 5 shows the *IGD* values obtained from KnRVEA and other algorithms for WFG test suite with 3, 5, 10, and 15 objectives (36 test instances). For WFG1 test function, KnRVEA provides better results than the other five algorithms for all test instances. I-MOEA/D is the second best algorithm. The results obtained from KnEA are better than HypE algorithm.

For WFG2 benchmark test function, the best *IGD* value is obtained from KnRVEA for all test instances. RVEA and I-MOEA/D are the second and third best algorithms for this benchmark test function, respectively. KnEA outperforms HypE in terms of *IGD* for all test instances.

The results obtained from KnRVEA are better than other algorithms for WFG3 benchmark test function. For 5-, 10-, and 15-objective test instances, I-MOEA/D is the second best algorithm. For 3-objective test instance, RVEA is the second best performing algorithm after KnRVEA. For 3- and 5-objective test instances, HypE performs better than

KnEA. Whereas, KnEA performs better than HypE for 10- and 15-objective test instances.

For WFG4 test function, KnRVEA performs better results as compared to the other algorithms for all test instances. After KnRVEA, the results obtained from I-MOEA/D are better than others for 3-, 5-, and 10-objective test instances. For 15-objective test instance, RVEA is the second best algorithm. KnEA performs better than HypE algorithm.

The results obtained from KnRVEA for WFG5 test problem are superior than the other competitor algorithms for all test instances. For 3- and 5-objective test instances, I-MOEA/D provides better results than other algorithms except KnRVEA. For 10- and 15-objective test instances, RVEA is the second best algorithm after KnRVEA. The results obtained from HypE and KnEA are not competitive for this test problem.

WFG6 is a non-separable problem. For this problem, NSGA-III-NE outperforms the other algorithms in all test instances. The results obtained from RVEA and I-MOEA/D are very much similar. Therefore, these are the second best algorithms. KnRVEA is the third best algorithm for all test instances while the results obtained from KnEA are better than HypE.

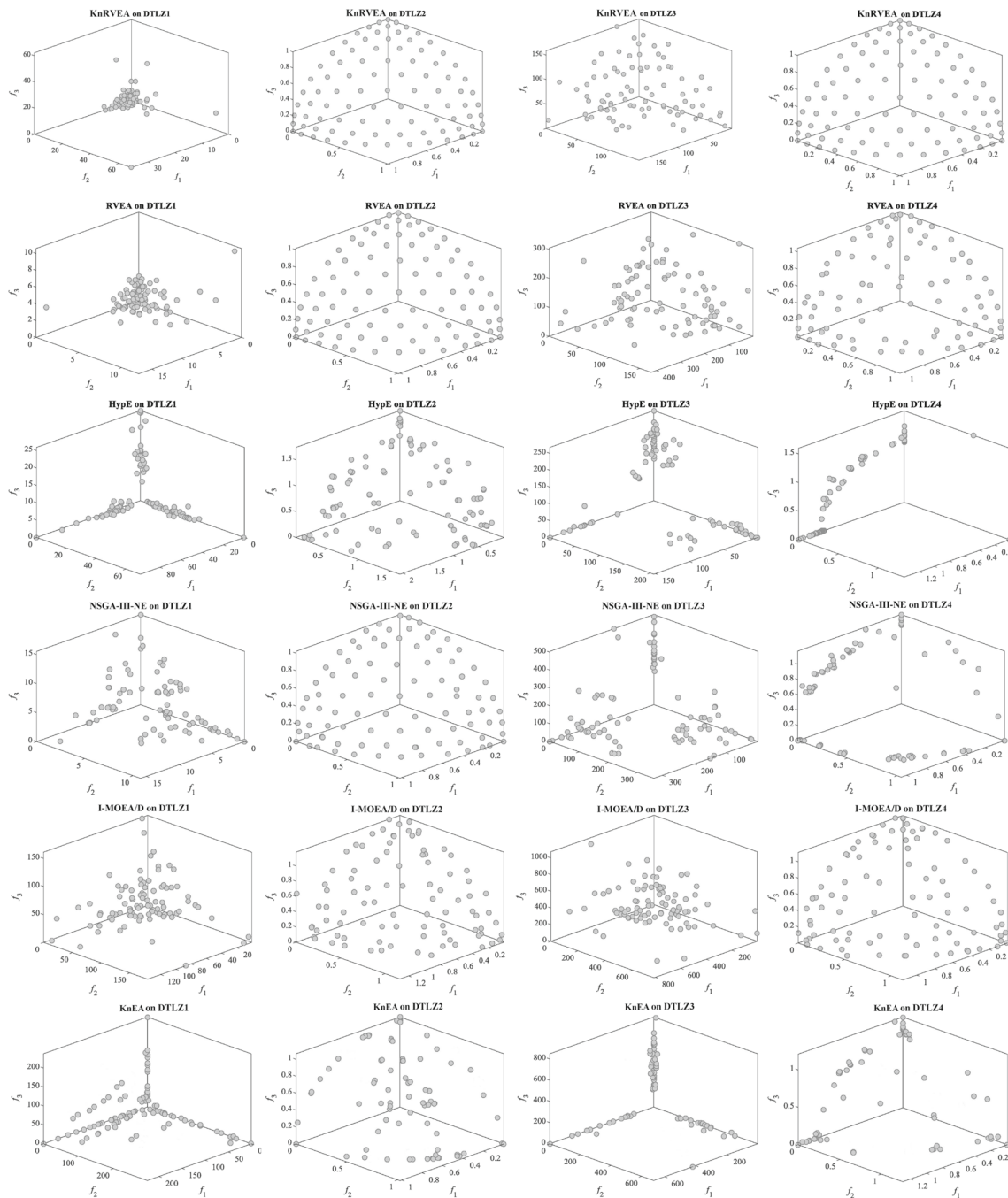


Fig. 4 The non-dominated solutions obtained from six algorithms on 3-objectives for DTLZ1, DTLZ2, DTLZ3, and DTLZ4 test instances

For WFG7 test problem, KnRVEA outperforms the other algorithm in terms of *IGD* for all test instances. For 3-objective test instance, RVEA is the second best algorithm. For 5-, 10-, and 15-objective test instances, I-MOEAD provides better results after KnRVEA. Whereas, KnEA performs better than HypE for all test instances.

For WFG8 test problem, KnRVEA performs better than the above-mentioned algorithms for all WFG8 test

instances. For 3-, 5-, and 10-objective test instances, I-MOEAD is the second best algorithm. For 15-objective test instance, RVEA is the second best optimization algorithm. KnEA performs better than HypE for all test instances.

For WFG9 test problem, KnRVEA outperforms the other competitor algorithms for all test instances. RVEA is the second best algorithm for all instances. KnEA provides better results than HypE for 5-, 10-, and 15-objective

Table 5 The IGD values obtained by proposed KnRVEA and other competitor algorithm in WFG test suite

F	M	Analysis	KnRVEA	RVEA	HypE	NSGA-III-NE	I-MOEA/D	KnEA	
WFG1	3	Mean	3.11E-05	3.19E-05	1.43E-04	2.90E-04	1.21E-04	4.51E-02	
		Median	3.79E-05	3.99E-05	2.61E-01	3.41E-03	2.93E-04	3.90E-02	
	5	Mean	2.93E-05	3.34E-05	1.99E-01	3.03E-04	2.38E-04	2.14E-02	
		Median	3.07E-05	3.68E-05	2.39E-01	6.67E-04	4.60E-04	2.33E-02	
	10	Mean	6.11E-05	1.51E-04	1.51E-01	2.24E-03	1.99E-03	3.67E-02	
		Median	1.19E-04	1.70E-04	1.69E-01	3.69E-03	2.99E-03	4.02E-02	
	15	Mean	2.39E-05	1.71E-04	1.12E-01	2.91E-03	2.00E-03	1.73E-02	
		Median	1.67E-04	2.39E-04	3.60E-01	4.15E-03	3.21E-03	2.18E-02	
	WFG2	3	Mean	1.11E-04	2.75E-04	4.81E-02	3.35E-03	3.01E-04	1.73E-02
			Median	2.04E-04	2.90E-04	5.98E-02	4.62E-03	4.12E-04	2.22E-02
		5	Mean	2.03E-04	3.11E-04	3.80E-01	3.45E-03	2.10E-03	1.25E-02
			Median	2.91E-04	3.67E-04	3.92E-01	4.13E-03	2.76E-03	1.59E-02
10		Mean	1.67E-04	2.61E-03	5.50E-01	2.42E-02	4.90E-03	3.22E-02	
		Median	2.64E-03	2.98E-03	6.00E-01	2.82E-02	5.23E-03	3.47E-02	
15		Mean	2.46E-04	4.03E-04	5.33E-01	2.40E-02	3.10E-03	3.42E-02	
		Median	3.44E-04	5.09E-04	6.49E-01	2.89E-02	4.40E-03	4.34E-02	
WFG3		3	Mean	2.24E-05	1.10E-04	1.60E-02	9.70E-04	5.00E-04	2.45E-02
			Median	6.05E-05	2.30E-04	1.88E-02	4.02E-03	1.06E-03	2.09E-02
		5	Mean	2.19E-04	1.13E-02	1.29E-02	3.25E-03	6.15E-04	2.61E-02
			Median	1.11E-03	1.81E-02	1.91E-02	5.00E-03	1.31E-03	3.93E-02
	10	Mean	2.15E-03	3.01E-01	4.86E-02	8.60E-03	3.97E-03	4.69E-02	
		Median	3.30E-03	3.70E-01	5.93E-02	8.98E-02	4.05E-03	4.57E-02	
	15	Mean	2.67E-04	2.96E-01	3.33E-02	2.13E-02	4.90E-03	2.52E-02	
		Median	4.67E-03	3.92E-01	5.92E-02	3.69E-02	6.54E-03	2.97E-02	
	WFG4	3	Mean	1.85E-04	4.96E-04	7.39E-02	3.64E-04	2.20E-04	4.42E-02
			Median	2.25E-04	5.88E-04	8.40E-02	5.01E-04	3.73E-04	5.70E-02
		5	Mean	5.67E-05	2.34E-01	3.20E-01	7.74E-04	3.53E-04	3.36E-02
			Median	6.35E-05	3.61E-01	4.70E-01	8.88E-03	4.67E-04	5.50E-02
10		Mean	1.12E-03	2.51E-03	5.81E-01	6.70E-03	2.33E-03	3.40E-02	
		Median	2.07E-03	4.32E-03	7.83E-01	7.53E-03	5.70E-03	4.74E-02	
15		Mean	1.76E-04	2.95E-04	5.28E-01	6.37E-03	2.52E-03	4.12E-02	
		Median	2.93E-04	3.33E-04	6.94E-01	4.67E-02	4.94E-03	4.99E-02	
WFG5		3	Mean	3.81E-05	4.69E-04	3.53E-04	3.50E-04	2.20E-04	2.55E-02
			Median	5.19E-05	5.99E-04	4.08E-01	3.84E-03	3.93E-04	3.15E-02
		5	Mean	3.12E-05	4.68E-05	2.57E-01	4.25E-04	3.73E-04	3.08E-02
			Median	3.85E-05	4.99E-05	2.98E-01	5.28E-04	4.60E-04	4.70E-02
	10	Mean	5.19E-05	2.98E-04	3.00E-01	3.63E-03	2.57E-03	2.74E-02	
		Median	1.22E-04	2.85E-04	3.92E-01	4.24E-03	3.20E-03	3.39E-02	
	15	Mean	2.11E-05	2.62E-04	2.17E-01	3.11E-03	3.55E-03	2.93E-02	
		Median	2.34E-04	3.04E-04	3.33E-01	5.00E-03	4.59E-03	3.95E-02	
	WFG6	3	Mean	2.61E-03	2.70E-04	5.35E-02	1.31E-04	5.00E-04	3.67E-02
			Median	3.80E-03	3.99E-04	6.18E-02	2.10E-04	6.13E-04	3.97E-02
		5	Mean	4.67E-03	1.85E-04	2.74E-01	1.78E-04	1.21E-03	1.07E-02
			Median	5.47E-03	2.79E-04	2.99E-01	2.31E-04	1.63E-03	1.68E-02
10		Mean	1.84E-02	1.34E-03	6.33E-01	2.39E-04	3.02E-03	2.22E-02	
		Median	2.92E-02	2.96E-03	7.52E-01	2.71E-03	4.01E-03	3.90E-02	
15		Mean	1.48E-02	5.25E-04	6.42E-01	2.11E-04	4.46E-03	2.00E-02	
		Median	2.83E-02	6.99E-04	7.02E-01	3.07E-04	5.01E-03	3.35E-02	

Table 5 (continued)

F	M	Analysis	KnRVEA	RVEA	HypE	NSGA-III-NE	I-MOEA/D	KnEA
WFG7	3	Mean	2.37E-05	1.67E-04	1.25E-02	7.75E-04	5.69E-04	2.11E-02
		Median	4.10E-05	2.34E-04	1.95E-02	8.23E-03	8.99E-03	2.23E-02
	5	Mean	1.19E-04	1.69E-02	2.19E-02	3.45E-03	5.97E-04	3.13E-02
		Median	2.16E-03	2.07E-02	2.93E-02	4.47E-03	6.98E-03	5.37E-02
	10	Mean	2.15E-03	3.71E-01	4.06E-02	4.67E-03	3.19E-03	4.11E-02
		Median	2.98E-03	4.96E-01	4.93E-02	5.37E-02	5.55E-03	4.07E-02
15	Mean	2.17E-03	2.67E-01	4.05E-02	2.33E-02	5.52E-03	1.37E-02	
	Median	2.81E-03	3.90E-01	5.49E-02	3.67E-02	6.50E-03	2.76E-02	
WFG8	3	Mean	1.19E-04	3.03E-04	7.67E-02	3.40E-04	2.00E-04	3.97E-02
		Median	2.06E-04	5.84E-04	8.40E-02	5.10E-04	3.71E-04	3.92E-02
	5	Mean	5.02E-05	1.57E-01	2.67E-01	9.96E-04	1.03E-04	2.47E-02
		Median	7.00E-05	2.61E-01	3.90E-01	9.01E-03	2.37E-04	3.51E-02
	10	Mean	2.12E-03	3.67E-03	6.81E-01	5.27E-03	3.30E-03	3.17E-02
		Median	2.60E-03	4.31E-03	7.92E-01	6.53E-03	4.71E-03	4.81E-02
15	Mean	1.11E-04	2.76E-04	5.08E-01	4.86E-03	2.41E-03	1.12E-02	
	Median	1.97E-04	3.31E-04	5.41E-01	5.10E-02	3.91E-03	3.73E-02	
WFG9	3	Mean	2.31E-04	3.70E-04	4.05E-02	4.51E-03	3.55E-04	4.60E-02
		Median	2.77E-04	4.99E-04	5.17E-02	4.87E-03	5.13E-04	6.90E-02
	5	Mean	1.08E-04	1.80E-04	2.67E-01	4.49E-03	1.70E-03	4.99E-02
		Median	2.01E-04	2.45E-04	2.78E-01	5.13E-03	1.88E-03	2.01E-02
	10	Mean	2.91E-04	2.34E-03	5.42E-01	2.84E-02	4.27E-03	6.09E-02
		Median	1.37E-03	4.96E-03	6.52E-01	3.52E-02	5.67E-03	4.99E-02
15	Mean	2.39E-04	4.25E-04	5.02E-01	3.68E-02	5.87E-03	3.57E-02	
	Median	3.17E-04	5.99E-04	6.02E-01	4.83E-02	5.99E-03	4.25E-02	

The obtained best results are in bold

test instances. Whereas, I-MOEA/D performs better than NSGA-III-NE for all test instances.

Figures 5, 6, 7, 8 and 9 show the non-dominated fronts obtained from KnRVEA and other competitor algorithms for 15- and 3-objective WFG benchmark test functions, respectively. For 15-objective WFG test functions, RVEA, NSGA-III-NE, and MOEA/DD provide uniform convergence while HypE and KnEA provide good convergence over all Pareto front. For 3-objective WFG test functions, KnRVEA, RVEA, and NSGA-III-NE algorithms provide good Pareto front while the Pareto fronts of HypE, I-MOEA/D, and KnEA provide good uniform distribution. The results reveal that the KnRVEA, RVEA, and NSGA-III-NE provide better coverage over the Pareto front. On the other hand, the performance of HypE, I-MOEA/D, and KnEA are challenging but they fail to provide better converge in most test instances. After analysing the results, it has been concluded that KnRVEA outperforms on high dimensional objective search space in terms of performance measures.

5.5 Experimentation 3: Wilcoxon signed-rank test

In the literature [51], it has been observed that *IGD* and *HV* do not provide any guarantee for better convergence and diversity because sometimes the obtained solutions are not close to the Pareto optimal front. To resolve this problem, Wilcoxon signed-rank test [53] has been conducted on the average value of *IGD* and *HV*. It is used to compare KnRVEA with other algorithms in pair-wise manner. The positive rank is given to proposed algorithm if it is better than the competitor algorithms with respect to a particular metric measure (i.e., *IGD* and *HV*). Otherwise, the negative rank is assigned. For comparison, a significance level is set to 0.10 and summed up all the positive and negative rank [53]. The *TH* should be less than or equal to 100 to reject the null hypothesis. The results of Wilcoxon test are tabulated in Table 6 where +, -, and = indicate that the performance of KnRVEA is superior (+), inferior (-), and equal (=) to the competitor algorithms, respectively. It

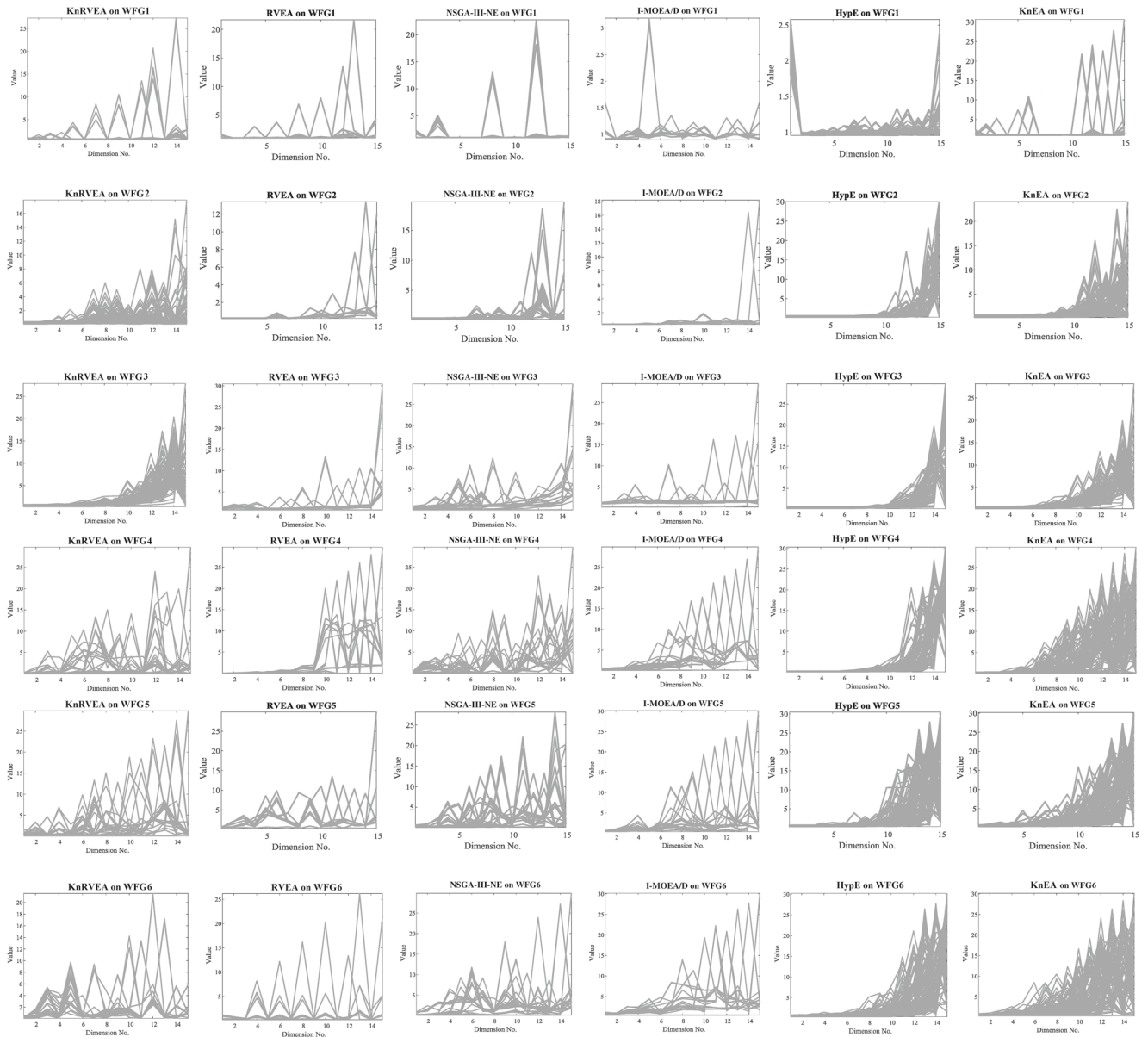


Fig. 5 The non-dominated fronts obtained from six algorithms on 15-objectives for WFG1, WFG2, WFG3, WFG4, WFG5, and WFG6 test instances

is observed from Table 6 that the KnRVEA outperforms over all the competitor algorithms in terms of *IGD* and *HV* performance measures except NSGA-III-NE.

5.6 Experimentation 4: Statistical significance test

Besides the basic statistical analysis (i.e., mean and median), ANOVA test has been performed for comparison of RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA, and proposed KnRVEA. ANOVA is used to test whether the results obtained from proposed KnRVEA differ from the results of other competitive algorithms in a statistical significant way. A *p*-value determines the statistical

significance level of KnRVEA. The 95% confidence interval is used for ANOVA testing. Therefore, algorithm is statistically significant if and only if the *p*-value is less than 0.05. ANOVA test results for DTLZ benchmark test suite on performance measures, i.e., *IGD* and *HV*, are reported in Tables 7 and 8. It is observed from these tables that KnRVEA is statistically different from the other algorithms. *p*-value obtained from KnRVEA is much smaller than 0.05. Table 9 shows the ANOVA test results for WFG test suite on performance measure *IGD*. The *p*-value obtained from KnRVEA is much smaller than 0.05 for all WFG test instances. The results demonstrate that the *p*-value obtained from KnRVEA is much smaller than 0.05

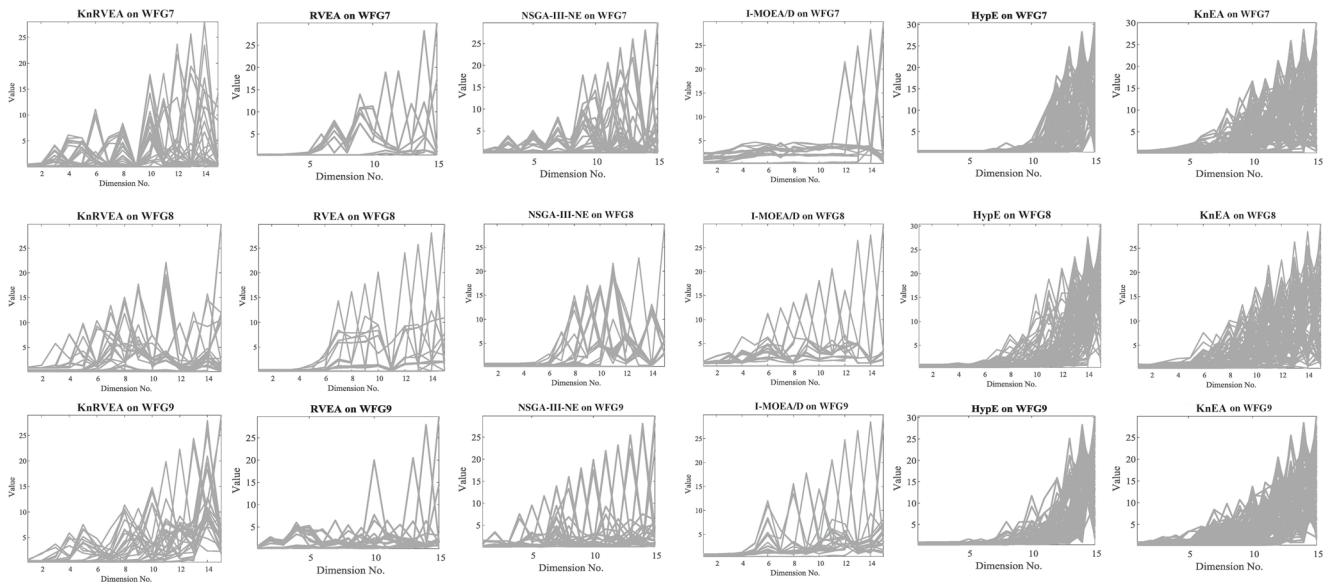


Fig. 6 The non-dominated fronts obtained from six algorithms on 15-objectives for WFG7, WFG8, and WFG9 test instances

for all benchmark test problems. Therefore, KnRVEA is statistically different from the other competitor algorithms.

5.7 Sensitivity analysis

The parameter tuning of many-objective algorithms is one of the major challenge. Different values of parameter might yield different results as reported from the exiting literature. KnRVEA algorithm involves three parameters namely Threshold (T), Crossover (n_c), and Mutation (n_m). The sensitivity investigation of these parameters has been discussed by varying their values.

1. Threshold (T): KnRVEA algorithm was run for different value of T . The values of T used in experimentation are 0.2, 0.5, 0.8, and 1.0. Figure 10a reveals that KnRVEA converges towards the optimum when the value of T is 0.5.
2. Crossover (n_c): To investigate the effect of crossover, KnRVEA algorithm was executed for 10, 20, 30, and 40. For brevity, the $DTLZ1$ test function is chosen for analysis. Figure 10b shows the effect of number of search agents on the performance of the above-mentioned algorithms. For most of these functions, it was found that KnRVEA provides best optimal solutions when the number of search agent is set to 20.
3. Mutation (n_m): KnRVEA algorithm was run for different values of parameter n_m . The values of n_m used in experimentation are 10, 20, 30, and 40. Figure 10c shows that the function $DTLZ1$ obtains best optimal solution when the value of n_m is set to 20.

5.8 Discussion

This subsection explains why KnRVEA provides better results than the other algorithms. KnRVEA maintains diversity among solutions. This is achieved by utilizing hypervolume indicator. It is evident from results that the solutions are very close to Pareto optimal solution. The solutions are distributed in whole region of Pareto front. Whereas, the above-mentioned algorithms do not maintain uniform Pareto front solutions (i.e., diversity). Besides this, KnRVEA also converges the solutions very fast. Figure 11 shows the evolutionary progress over number of generations. It can be seen that the number of non-dominated solutions decrease with an increase in number of objectives. This shows applicability of KnRVEA on many-objective optimization problems.

KnRVEA maintains the information about non-dominated solutions. This can be achieved by utilizing the knee adaptation strategy. This strategy adjusts the solutions according to objective functions. It ensures uniform distribution of solution, even if objective functions are not well normalized or Pareto fronts are in irregular shape.

6 Real-life applications

In this section, the performance of the proposed KnRVEA has been tested on three real-life problems such as Multi-objective Travelling Salesman Problem ($MOTSP$), Multi-objective 0/1 Knapsack Problem ($MOKP$), and Water Resource Planning Problem ($WRPP$). These problems are

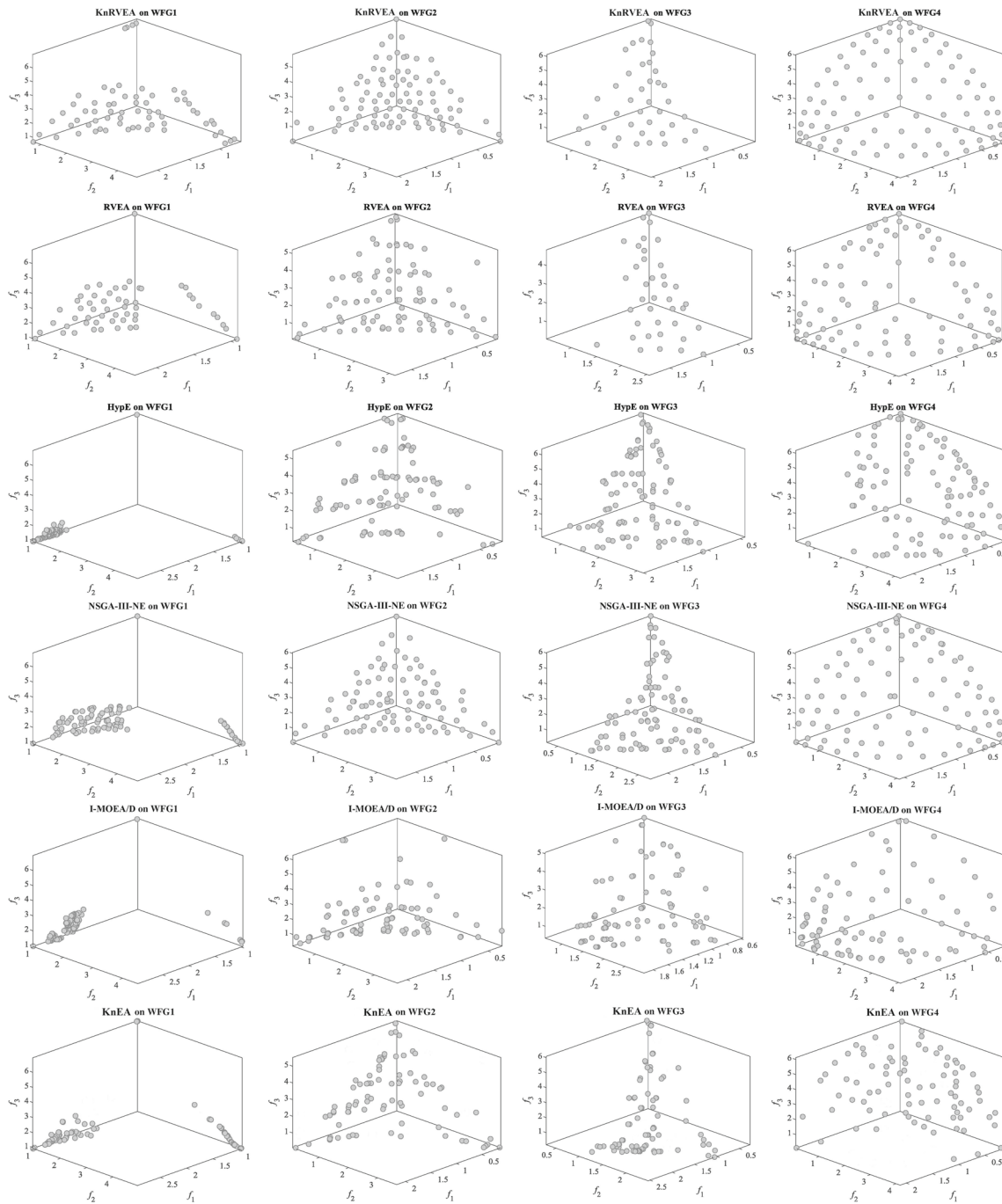


Fig. 7 The non-dominated solutions obtained from six algorithms on 3-objectives for WFG1, WFG2, WFG3, and WFG4 test instances

used to show the effectiveness of proposed KnRVEA in real-life problems.

6.1 Multi-objective travelling salesman problem

The mathematical formulation of multi-objective travelling salesman problem (*MOTSP*) is as follows [34]: Given a set n cities and p cost c_{ij}^k , $k = 1, 2, \dots, p$ (travel from city

i to j). The main objective of this problem is to find a tour, i.e., a cyclic permutation R of n cities.

$$\text{Minimize } \sum_{i=1}^{n-1} c_{R(i),R(i+1)}^k + c_{R(n),R(1)}^k, \quad k = 1, 2, \dots, p \tag{13}$$

In this paper, p is set to 3, 5, 10, and 15.

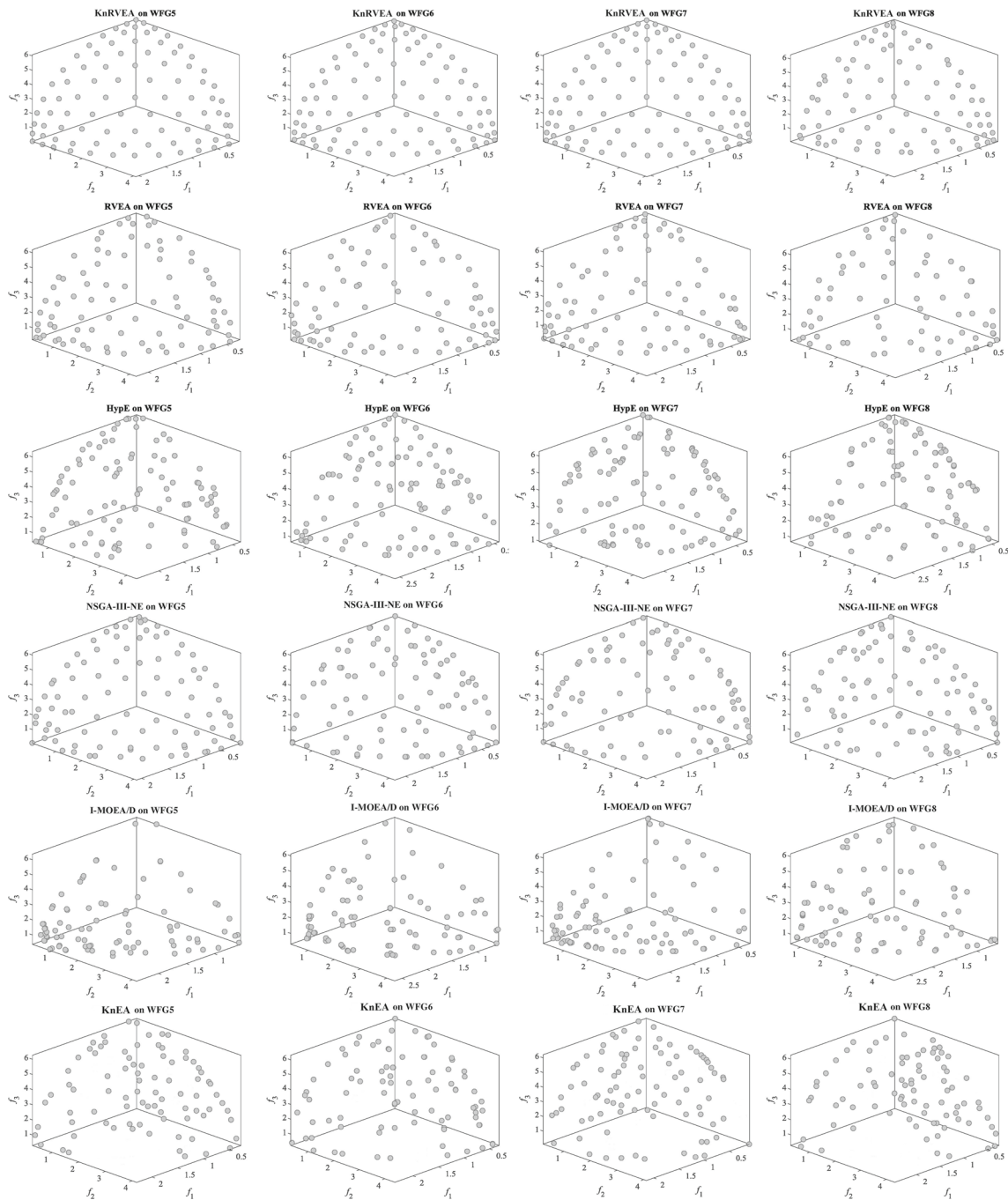


Fig. 8 The non-dominated solutions obtained from six algorithms on 3-objectives for WFG5, WFG6, WFG7, and WFG8 test instances

Table 10 shows *IGD* value obtained from KnRVEA and other competitor algorithms on *MOTSP*. It can be seen that KnRVEA performs better than other algorithms for all test instances (i.e., 3, 5, 10, 15). For 3- and 5-objective instances, KnRVEA provides best *IGD* value. RVEA is the second best algorithm for 3-objective test instance. For

5-objective test instance, I-MOEA/D is the second best performing algorithm. For 10- and 15-objective test instances, KnRVEA outperforms the other five algorithms. RVEA is the second best algorithm for these test instances. It is worth mentioning that KnRVEA consistently performing better than other algorithms for all four *MOTSP* instances.

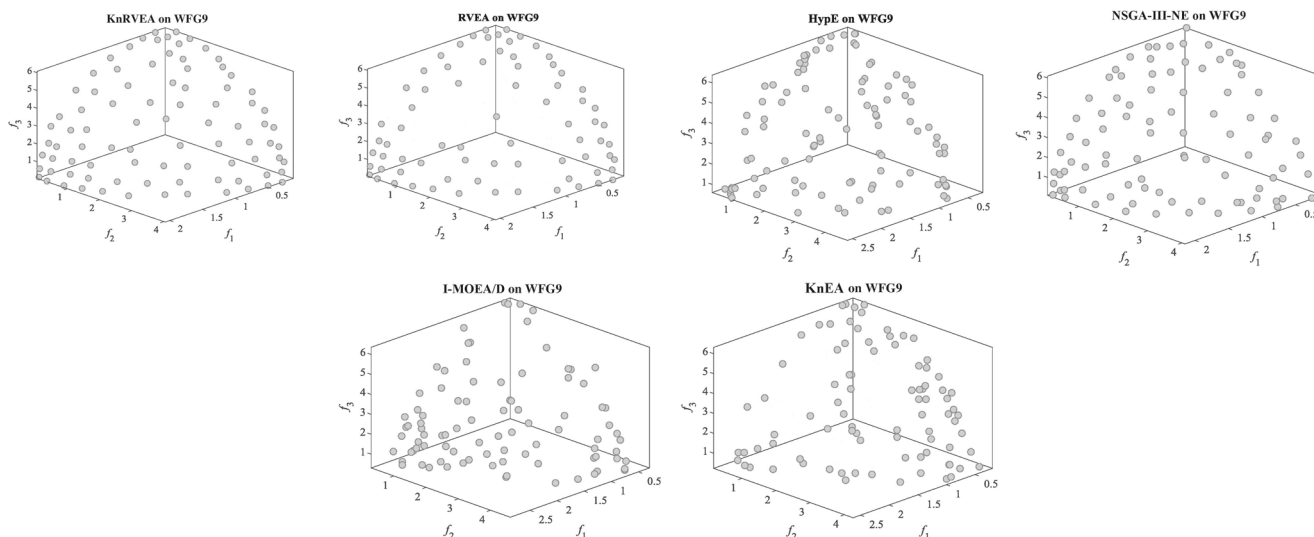


Fig. 9 The non-dominated solutions obtained from six algorithms on 3-objectives for WFG9 test instance

6.2 Multi-objective 0/1 knapsack problem

Multi-objective 0/1 knapsack problem (*MOKP*) is a combinatorial optimization problem. In multi-objective 0/1 knapsack problem, there are q knapsacks and a set of n items. The objective of this problem is to find the vector $z = (z_1, z_2, \dots, z_n) \in \{0, 1\}^n$, where $z_k = 1$ indicates the item k is in all knapsacks and $z_k = 0$ represents the item k is not in any knapsacks [38]. The mathematical formulation of multi-objective 0/1 knapsack problem is defined as follows [38]:

$$\text{Maximize } f_k(z) = \sum_{j=1}^n p_{kj} \times z_j, \quad k = 1, 2, \dots, q \quad (14)$$

Subject to:

$$\sum_{j=1}^n w_{kj} \times z_j \leq c_k \quad (15)$$

where p_{kj} is the profit of item j placed in knapsack k , w_{kj} is the weight of item j placed in knapsack k , and c_k is the capacity of knapsack k . z_q is the selected item q in the knapsack. According to [38], the value of p_{kj} and w_{kj} lies in the interval of [10, 100].

Table 11 shows the results obtained from KnRVEA and other competitive algorithms on *MOKP*. It is observed from table that KnRVEA provides better results than the other algorithms for 3- and 5-objective test instances. KnRVEA always provides lower *IGD* value than the other algorithms. For these instances, RVEA is the second best performing algorithm. For 10- and 15-objective test instances, KnRVEA provides lowest value of *IGD* as compared to other algorithms. RVEA and KnEA are the second best algorithm for these instances. The results

reveal that KnRVEA maintains a good balance between convergence and diversity.

6.3 Water resource planning problem

Water resource planning (*WRPP*) involves optimal planning for storm drainage systems in urban area which is described by Musselman and Talavage [35]. It consists of three variables such as local detention storage capacity (z_1), maximum treatment rate (z_2), and maximum allowable overflow rate (z_3). It uses five objectives functions ($f_1 - f_5$) and seven constraints ($g_1 - g_7$).

Table 12 shows the *IGD* values obtained from six algorithms on four *WRPP* instances. From this table, it is observed that KnRVEA provides better results than other algorithms for 3- and 5-objective test instances. The *IGD* value obtained from KnRVEA is much smaller than the others. RVEA is the second best algorithm for these test instances. For 10- and 15-objective test instances, the proposed KnRVEA always provides smaller value of *IGD*. The results reveal that KnRVEA is able to solve a problem with many objectives and constraints.

Table 6 Wilcoxon signed-rank test results between proposed KnRVEA and other algorithms based on average *IGD* and *HV*

Algorithms	<i>IGD</i>	<i>HV</i>
RVEA	=	+
HypE	+	+
NSGA-III-NE	+	-
I-MOEA/D	+	+
KnEA	+	+

Table 7 ANOVA results for IGD values on DTLZ benchmark test functions

F	<i>p</i> -value	KnRVEA	RVEA	HypE	NSGA-III-NE	I-MOEA/D	KnEA
<i>DTLZ1</i>	1.02E-14	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, I-MOEA/D
<i>DTLZ2</i>	1.92E-36	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, HypE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, I-MOEA/D
<i>DTLZ3</i>	3.71E-04	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEANSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, KnEA	KnRVEA, HypE, NSGA-III-NE, I-MOEA/D
<i>DTLZ4</i>	2.26E-11	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, I-MOEA/D

Table 8 ANOVA results for HV values on DTLZ benchmark test functions

F	<i>p</i> -value	KnRVEA	RVEA	HypE	NSGA-III-NE	I-MOEA/D	KnEA
<i>DTLZ1</i>	6.77E-11	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, NSGA-III-NE, I-MOEA/D	KnRVEA, RVEA, HypE, I-MOEA/D, KnEA	KnRVEA, HypE, NSGA-III-NE, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, I-MOEA/D
<i>DTLZ2</i>	1.02E-47	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, NSGA-III-NE, I-MOEA/D	KnRVEA, RVEA, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, HypE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, I-MOEA/D
<i>DTLZ3</i>	3.69E-17	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, HypE, KnEA	KnRVEA, RVEA, HypE, KnEA	KnRVEA, HypE, NSGA-III-NE, I-MOEA/D
<i>DTLZ4</i>	8.80E-23	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	HypE, NSGA-III-NE, KnEA	KnRVEA, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, I-MOEA/D, KnEA	KnRVEA, RVEA, NSGA-III-NE, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, I-MOEA/D

Table 9 ANOVA results for IGD values on WFG benchmark test functions

F	<i>p</i> -value	KnRVEA	RVEA	HypE	NSGA-III-NE	I-MOEA/D	KnEA
<i>WFG1</i>	2.02E-54	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, I-MOEA/D
<i>WFG2</i>	3.34E-26	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, NSGA-III-NE, I-MOEA/D, KnEA	RVEA, HypE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, KnEA	KnRVEA, RVEA, NSGA-III-NE, I-MOEA/D
<i>WFG3</i>	2.37E-31	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, HypE, NSGA-III-NE, KnEA	KnRVEA, RVEA, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, I-MOEA/D, KnEA	KnRVEA, HypE, NSGA-III-NE, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, I-MOEA/D
<i>WFG4</i>	3.39E-40	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	RVEA, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, I-MOEA/D, KnEA	KnRVEA, HypE, NSGA-III-NE, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, I-MOEA/D
<i>WFG5</i>	5.01E-19	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, NSGA-III-NE, I-MOEA/D, KnEA	RVEA, HypE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, I-MOEA/D
<i>WFG6</i>	3.34E-23	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, I-MOEA/D
<i>WFG7</i>	2.11E-06	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, KnEA	RVEA, HypE, NSGA-III-NE, I-MOEA/D
<i>WFG8</i>	1.32E-43	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, I-MOEA/D
<i>WFG9</i>	2.61E-39	RVEA, HypE, NSGA-III-NE, I-MOEA/D, KnEA	KnRVEA, HypE, I-MOEA/D, KnEA	KnRVEA, RVEA, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, I-MOEA/D, KnEA	KnRVEA, RVEA, HypE, NSGA-III-NE, KnEA	KnRVEA, RVEA, NSGA-III-NE, I-MOEA/D

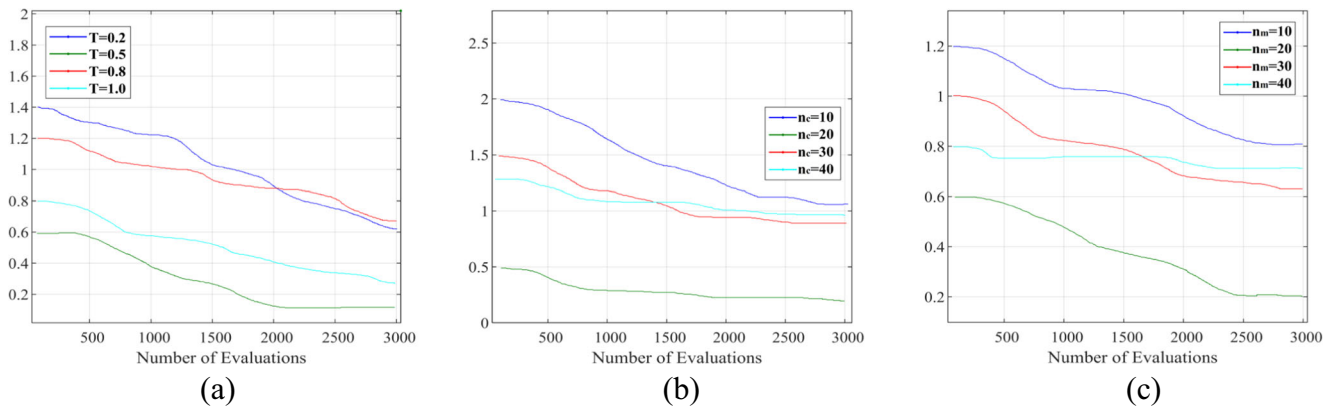


Fig. 10 Sensitivity analysis of proposed KnRVEA algorithm

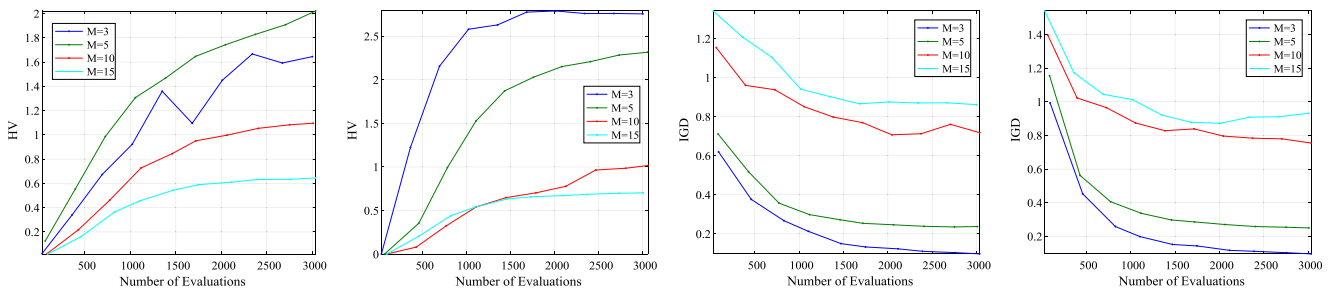


Fig. 11 The convergence analysis of non-dominated solutions on 3-, 5-, 10-, and 15-objectives for DTLZ2 and DTLZ4 test functions in terms of HV and IGD

Table 10 The IGD values obtained by proposed KnRVEA and other competitor algorithms on MOTSP

F	M	Analysis	KnRVEA	RVEA	HypE	NSGA-III-NE	I-MOEA/D	KnEA
3	Mean		6.11E-05	7.60E-05	2.51E-04	3.90E-04	2.10E-04	1.52E-02
			6.39E-05	3.11E-04	1.05E-03	2.52E-03	4.71E-03	2.90E-01
5	Mean		3.72E-05	2.01E-04	2.80E-02	4.23E-03	4.15E-05	1.30E-03
			4.10E-04	2.57E-03	4.25E-02	7.91E-02	6.76E-04	1.80E-02
10	Mean		9.09E-04	3.49E-03	2.70E-02	7.26E-02	2.20E-02	8.60E-01
			2.70E-03	6.61E-02	1.99E-01	4.00E-01	2.20E-01	9.89E-01
15	Mean		1.00E-05	4.37E-04	7.08E-03	3.49E-02	6.66E-02	2.06E-03
			4.14E-04	7.77E-03	4.39E-02	8.10E-02	9.05E-02	5.97E-02

The obtained best results are in bold

Table 11 The IGD values obtained by proposed KnRVEA and other competitor algorithms on MOKP

F	M	Analysis	KnRVEA	RVEA	HypE	NSGA-III-NE	I-MOEA/D	KnEA
3	Mean		6.81E-05	7.66E-05	2.66E-04	3.47E-04	2.22E-04	1.50E-02
			1.99E-04	2.90E-04	7.49E-03	3.39E-03	4.33E-03	9.95E-02
5	Mean		1.72E-05	4.10E-05	2.89E-02	4.65E-04	1.60E-03	6.33E-02
			3.13E-04	3.66E-04	2.26E-01	7.97E-03	2.77E-02	9.89E-02
10	Mean		1.11E-05	3.94E-04	5.40E-02	4.25E-03	3.47E-02	4.77E-02
			2.20E-04	2.79E-03	3.90E-01	5.55E-02	4.40E-02	5.80E-01
15	Mean		5.01E-02	8.78E-02	5.59E-01	4.47E-01	3.34E-01	7.07E-02
			7.74E-02	9.90E-02	7.77E-01	6.75E-01	4.26E-01	8.50E-01

The obtained best results are in bold

Table 12 The IGD values obtained by proposed KnRVEA and other competitor algorithms on *WRPP*

F	M	Analysis	KnRVEA	RVEA	HypE	NSGA-III-NE	I-MOEA/D	KnEA
3	<i>Mean</i>		8.80E-07	3.00E-06	4.37E-05	5.09E-04	6.49E-03	3.47E-05
	<i>Median</i>		7.00E-06	5.90E-05	3.11E-04	4.37E-03	3.19E-02	2.47E-04
5	<i>Mean</i>		2.12E-05	4.69E-04	5.05E-03	3.20E-04	6.37E-03	3.44E-03
	<i>Median</i>		3.00E-04	6.56E-04	2.47E-02	6.69E-03	3.72E-02	3.80E-02
10	<i>Mean</i>		9.10E-05	3.66E-04	3.49E-03	4.11E-02	2.91E-03	1.47E-03
	<i>Median</i>		6.20E-04	7.09E-04	1.92E-02	4.50E-01	1.16E-02	4.89E-02
15	<i>Mean</i>		1.00E-05	1.77E-04	1.81E-03	4.05E-02	2.91E-02	1.83E-03
	<i>Median</i>		1.02E-04	2.56E-03	2.60E-02	9.10E-02	7.25E-02	1.50E-02

The obtained best results are in bold

7 Conclusions

In this paper, a hybrid many-objective optimization algorithm, named KnRVEA, has been proposed. KnRVEA utilizes the features of knee points and reference vector adaptation strategies. The knee points are used to improve the search performance when the number of objectives increases. The knee points are used to increase the selection pressure for improving the convergence performance of proposed algorithm. The proposed knee adaption strategy is used to obtain a set of uniformly distributed solutions. KnRVEA has been implemented and tested on thirteen benchmark test functions. On comparing the results of KnRVEA with other algorithms, it has been observed that KnRVEA outperforms five recently developed MOEAs such as RVEA, HypE, NSGA-III-NE, I-MOEA/D, and KnEA. KnRVEA has further been applied on three well-known real-life problems. The results reveal that KnRVEA provides better results among all the competitive algorithms.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

- Ishibuchi H, Akedo N, Nojima Y (2015) Behavior of multiobjective evolutionary algorithms on manyobjective knapsack problems. *IEEE Trans Evol Comput* 19(2):264–283
- Jin Yaochu, Sendhoff B (2003) Connectedness, regularity and the success of local search in evolutionary multi-objective optimization. In: *Proceedings of the 2003 congress on evolutionary computation (CEC'03)*, vol 3, pp 1910–1917
- Deb K (2001) *Multi-objective optimization using evolutionary algorithms*. Wiley, New York
- Coello Coello CA, Lamont GB, Veldhuizen DAV (2007) *Evolutionary algorithms for solving multi-objective problems*, 2nd edn. Springer, New York
- Li K, Zheng J, Li M, Zhou C, Lv H (2009) A novel algorithm for nondominated hypervolume-based multiobjective optimization. In: *Proceedings of IEEE international conference system man cybernetics (SMC)*. San Antonio, pp 5220–5226
- Lygoe RJ, Cary M, Fleming PJ (2013) A real-world application of a many-objective optimisation complexity reduction process. In: *Proceedings of 7th international conference of evolution multi-criterion optimization (EMO)*. Sheffield, UK, pp 641–655
- Fu G, Kapelan Z, Kasprzyk JR, Reed P (2013) Optimal design of water distribution systems using many-objective visual analytics. *J Water Resour Plan Manag* 139(6):624–633
- Chikumbo O, Goodman ED, Deb K (2012) Approximating a multidimensional Pareto front for a land use management problem: a modified MOEA with an epigenetic silencing metaphor. In: *Proceedings of 2012 IEEE congress evolutionary computation (CEC)*. QLD, Brisbane, pp 1–9
- Deb K, Agrawal S, Pratap A, Meyarivan T (2002) A fast and elitist multi-objective genetic algorithm: NSGAII. *IEEE Trans Evol Comput* 6(2):182–197
- Emmerich M, Beume N, Naujoks B (2005) An EMO algorithm using the hypervolume measure as selection criterion. In: *Conference on evolutionary multi-criterion optimization (EMO 2005)*, volume 3410 of LNCS. Springer, pp 62–76
- Igel C, Hansen N, Roth S (2007) Covariance Matrix Adaptation for multi-objective optimization. *Evol Comput* 15(1):1–28
- Brockhoff D, Zitzler E (2007) Improving hypervolume-based multiobjective evolutionary algorithms by using objective reduction methods. In: *Congress on evolutionary computation (CEC 2007)*, pp 2086–2093
- Bader J, Zitzler E (2011) Hype: an algorithm for fast hypervolume-based many-objective optimization. *Evol Comput* 19(1):45–76
- Cheng R, Jin Y, Olhofer M, Sendhoff B (2016) A reference vector guided evolutionary algorithm for many-objective optimization. *IEEE Trans Evol Comput* 20(5):773–791
- While L, Hingston P, Barone L, Huband S (2006) A faster algorithm for calculating hypervolume. *IEEE Trans Evol Comput* 10(1):29–38
- Kollat J, Reed P (2007) A computational scaling analysis of multiobjective evolutionary algorithms in long-term groundwater monitoring applications. *Adv Water Resour* 30(3):408–419
- Dhiman G, Kumar V (2018) Multi-objective spotted hyena optimizer: a multi-objective optimization algorithm for engineering problems. *Knowl-Based Syst* 150:175–197
- Kollat JB, Reed P (2006) Comparison of multi-objective evolutionary algorithms for long-term monitoring design. *Adv Water Resour* 29(6):792–807
- Hadka D, Reed P (2012) Diagnostic assessment of search controls and failure modes in many-objective evolutionary optimization. *Evol Comput* 20(3):423–452. ISSN 1063-6560
- Hadka D, Reed P (2013) Borg: an auto-adaptive many-objective evolutionary computing framework. *Evol Comput* 21(2):231–259
- Bringmann K, Friedrich T, Neumann F, Wagner M (2011) Approximation guided evolutionary multi-objective optimization. In:

- International joint conference on artificial intelligence (IJCAI). AAAI Press, pp 1198–1203. ISBN 978-1-57735-514-4
22. Wagner M, Bringmann K, Friedrich T, Neumann F (2014) Efficient optimization of many objectives by approximation-guided evolution. *European Journal of Operational Research*, page in print. ISSN 0377–2217
 23. Zhang Q, Li H (2007) MOEA/D: a multiobjective evolutionary algorithm based on decomposition. *IEEE Trans Evol Comput* 11(6):712–731. ISSN 1089-778X
 24. Deb K, Jain H (2014) An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. *IEEE Trans Evol Comput* 18(4):577–601. ISSN 1089-778X
 25. Asafuddoula M, Ray T, Sarker R (2013) A decomposition based evolutionary algorithm for many objective optimization with systematic sampling and adaptive epsilon control. In: *Evolutionary multi-criterion optimization*, volume 7811 of LNCS. Springer, pp 413–427. ISBN 978-3-642-37139-4
 26. Asafuddoula M, Ray T, Sarker R (2015) A decomposition-based evolutionary algorithm for many objective optimization. *IEEE Trans Evol Comput* 19(3):445–460. ISSN 1089-778X
 27. Jain H, Deb K (2013) An improved adaptive approach for elitist nondominated sorting genetic algorithm for many-objective optimization. In: *Evolutionary multi-criterion optimization*, volume 7811 of LNCS. Springer, pp 307–321. ISBN 978-3-642-37139-4
 28. Jain H, Deb K (2014) An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part-II: handling constraints and extending to an adaptive approach, vol 18. ISSN 1089-778X
 29. Seada H, Deb K (2015) U-nsga-iii: a unified evolutionary optimization procedure for single, multiple, and many objectives: Proof-of-principle results. In: *Evolutionary multi-criterion optimization*, volume 9019 of LNCS. Springer, pp 34–49. ISBN 978-3-319-15891-4
 30. Wang H, Jiao L, Yao X (2014) An improved two-archive algorithm for many-objective optimization. *IEEE Trans Evol Comput* PP(99):1-1. ISSN 1089-778X
 31. Zhang X, Tian Y, Jin Y (2014) A knee point driven evolutionary algorithm for many-objective optimization. *IEEE Trans Evol Comput* PP(99):1-1. ISSN 1089-778X
 32. Deb K, Thiele L, Laumanns M, Zitzler E (2005) Scalable test problems for evolutionary multiobjective optimization. In: Abraham A, Jain L, Goldberg R (eds) *Evolutionary multiobjective optimization (advanced information and knowledge processing)*. Springer, London, pp 105–145
 33. Huband S, Hingston P, Barone L, While L (2006) A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Trans Evol Comput* 10(5):477–506
 34. Corne DW, Knowles JD (2007) Techniques for highly multiobjective optimisation: some nondominated points are better than others. In: *Proceedings of the 9th annual conference on genetic and evolutionary computation, GECCO*, pp 773–780
 35. Musselman K, Talavage J (1980) A trade-off cut approach to multiple objective optimization. *Oper Res* 28:1424–1435
 36. Bosman P, Thierens D (2003) The balance between proximity and diversity in multiobjective evolutionary algorithms. *IEEE Trans Evol Comput* 7(2):174–188
 37. Zou J, Li Q, Yang S, Bai H, Zheng J (2017) A prediction strategy based on center points and knee points for evolutionary dynamic multi-objective optimization. *Appl Soft Comput* 61:806–818
 38. Zitzler E, Thiele L (1999) Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE Trans Evol Comput* 3(4):257–271
 39. Li K, Deb K, Zhang Q, Kwong S (2015) An evolutionary many-objective optimization algorithm based on dominance and decomposition. *IEEE Trans Evol Comput* 19(5):694–716
 40. Ishibuchi H, Murata T (1998) A multi-objective genetic local search algorithm and its application to flowshop scheduling. *IEEE Trans syst Man Cybern Part C (Appl Rev)* 28(3):392–403
 41. Jaskiewicz A (2002) On the performance of multiple-objective genetic local search on the 0/1 knapsack problem - a comparative experiment. *IEEE Trans Evol Comput* 6(4):402–412
 42. Zitzler E, Laumanns M, Thiele L (2001) SPEA2: improving the strength Pareto evolutionary algorithm. *Tik Report*
 43. Zitzler E, Künzli S (2004) Indicator-based selection in multiobjective search. In: *Parallel problem solving from nature-PPSN VIII*. Springer, Heidelberg, pp 832–842
 44. Zheng W, Tan Y, Meng L, Zhang H (2018) An improved MOEA/d design for many-objective optimization problems. *Appl Intell* 48:1–23
 45. Basseur M, Burke EK Indicator-based multi-objective local search. In: *IEEE congress on evolutionary computation, CEC 2007*. IEEE, New York, pp 3100–3107
 46. Wagner T, Beume N, Naujoks B (2007) Pareto-, aggregation-, and indicator-based methods in many-objective optimization. In: *Evolutionary multi-criterion optimization*. Springer, Heidelberg, pp 742–756
 47. Emmerich M, Beume N, Naujoks B (2005) An emo algorithm using the hypervolume measure as selection criterion. In: *Evolutionary multi-criterion optimization*. Springer, Heidelberg, pp 62–76
 48. Gomez RH, Coello CC (2013) Mombi: a new metaheuristic for many-objective optimization based on the r2 indicator. In: *IEEE congress on evolutionary computation (CEC)*. IEEE, New York, pp 2488–2495
 49. Xiao JK, Li WM, Xiao XR, Lv CZ (2017) A novel immune dominance selection multi-objective optimization algorithm for solving multi-objective optimization problems. *Appl Intell* 46(3):739–755
 50. Li B, Li J, Ke T, Yao X (2015) Many-objective evolutionary algorithms: a survey. *ACM Comput Surv (CSUR)* 48(1):13:1–13:35
 51. Roy PC, Islam MM, Murase K, Yao X (2015) Evolutionary path control strategy for solving many-objective optimization problem. *IEEE Trans Cybern* 45(4):702–715
 52. Giagkiozis I, Purshouse RC, Fleming PJ (2013) Towards understanding the cost of adaptation in decomposition-based optimization algorithms. In: *Proceedings of the IEEE international conference on systems, man, and cybernetics*, pp. 615–620
 53. Corder GW, Foreman DI (2009) *Nonparametric statistics for NonStatisticians: a step-by-step approach*. Wiley, New York
 54. Walker DJ, Everson RM, Fieldsend JE (2013) Visualizing mutually nondominating solution sets in many-objective optimization. *IEEE Trans Evol Comput* 17(2):165–184
 55. Coello CAC (2009) Evolutionary multi-objective optimization: some current research trends and topics that remain to be explored. *Front Comput Sci China* 3(1):18–30
 56. Bi X, Wang C (2018) A niche-elimination operation based NSGA-III algorithm for many-objective optimization. *Appl Intell* 48:118–141
 57. Ngatchou P, Zarei A, El-Sharkawi A (2005) Pareto multi objective optimization. In: *Proceedings of the 13th international conference on, intelligent systems application to power systems*. Arlington, pp 84–91
 58. Cornell JA (2011) *Experiments with mixtures: designs, models, and the analysis of mixture data*. Wiley, New York, p 403

59. Deb K, Agrawal RB (1995) Simulated binary crossover for continuous search space. *Complex Syst* 9:115–148
60. Deb K, Goyal M (1996) A combined genetic adaptive search (geneAS) for engineering design. *Comput Sci Inf* 26:30–45
61. Zhang X, Tian Y, Cheng R, Jin Y (2014) An efficient approach to non-dominated sorting for evolutionary multi-objective optimization. *IEEE Trans Evol Comput* 19:201–213
62. Das I, Dennis JE (1998) Normal-boundary intersection: a new method for generating the pareto surface in nonlinear multicriteria optimization problems. *SIAM J Optim* 8(3):631–657
63. Dhiman G, Kumar V (2017) Spotted hyena optimizer: a novel bio-inspired based metaheuristic technique for engineering applications. *Adv Eng Softw* 114:48–70
64. Dhiman G, Kumar V (2019) Spotted hyena optimizer for solving complex and non-linear constrained engineering problems. In: *Harmony search and nature inspired optimization algorithms*. Springer, Singapore, pp 857–867
65. Dhiman G, Kumar V (2018) Emperor penguin optimizer: a bio-inspired algorithm for engineering problems. *Knowl-Based Syst* 159:20–50
66. Singh P, Dhiman G, Kaur A (2018) A quantum approach for time series data based on graph and Schrödinger equations methods. *Mod Phys Lett A* 33:1850208
67. Dhiman G, Kaur A (2017) Spotted hyena optimizer for solving engineering design problems. In: *International conference on machine learning and data science (MLDS)*. IEEE, pp 114–119
68. Dhiman G, Kumar V (2018) Astrophysics inspired multi-objective approach for automatic clustering and feature selection in real-life environment. *Modern Physics Letters B*, In Press
69. Singh P, Dhiman G (2017) A fuzzy-lp approach in time series forecasting. In: *International conference on pattern recognition and machine intelligence*. Springer, pp 243–253
70. Singh P, Dhiman G (2018) A hybrid fuzzy time series forecasting model based on granular computing and bio-inspired optimization approaches. *J Comput Sci* 27:370–385
71. Kaur A, Dhiman G (2019) A review on search-based tools and techniques to identify bad code smells in object-oriented systems. In: *Harmony search and nature inspired optimization algorithms*. Springer, Singapore, pp 909–921
72. Dhiman G, Kaur A (2018) Optimizing the design of airfoil and optical buffer problems using spotted hyena optimizer. *Designs* 2:28
73. Chandrawat RK, Kumar R, Garg BP, Dhiman G, Kumar S (2017) An analysis of modeling and optimization production cost through fuzzy linear programming problem with symmetric and right angle triangular fuzzy number. In: *Proceedings of sixth international conference on soft computing for problem solving*. Springer, Singapore, pp 197–211



Gaurav Dhiman received the Master Degree of Computer Applications from Thapar Institute of Engineering & Technology, Patiala, INDIA. He is currently working towards the Ph.D. degree at the Computer Science and Engineering Department, Thapar Institute of Engineering & Technology, Patiala, INDIA. He was selected as an outstanding reviewer from Knowledge-Based Systems (Elsevier). He has published more than 20 peer-review papers in SCI-

SCIE indexed journals. His research interests include Single-, Multi-, Many-objectives optimization (Bio-inspired, Evolutionary and Quantum), Soft computing (Type-1 and Type-2 fuzzy sets), Power systems, and Change detection using remotely sensed high-resolution satellite data. His research articles can be found in Knowledge-Based Systems (Elsevier), Advances in Engineering Software (Elsevier), Applied Intelligence (Springer), Journal of Computational Science (Elsevier), Applied Soft Computing (Elsevier), Swarm and Evolutionary Computation (Elsevier), and so on.