Information Sciences 474 (2019) 236–252

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

A scalarization-based dominance evolutionary algorithm for many-objective optimization

Burhan Khan*, Samer Hanoun, Michael Johnstone, Chee Peng Lim, Douglas Creighton, Saeid Nahavandi

Institute for Intelligent Systems Research and Innovation (IISRI), Deakin University, 75 Pigdons Road, Waurn Ponds, 3216, Australia

ARTICLE INFO

Article history:
Received 25 March 2016
Revised 11 September 2018
Accepted 17 September 2018
Available online 19 September 2018

Keywords:
Multi-objective
Many-objective
Optimization
Genetic algorithm
Evolutionary computation
Decomposition
Scalarization
Reference vectors

ABSTRACT

Classical Pareto-dominance based multi-objective evolutionary algorithms underperform when applied to optimization problems with more than three objectives. A class of multi-objective evolutionary algorithms introduced in the literature, utilizing pre-determined reference points acting as target vectors to maintain diversity in the objective space, has shown promising results. Inspired by this approach, we propose a scalarization-based dominance evolutionary algorithm (SDEA) that utilizes a reference point-based method and combine it with a novel sorting strategy that employs fitness values determined via scalarization. SDEA reduces computation complexity by eliminating the need for a Pareto-dominance approach to obtain non-dominated solutions. By means of a set of common benchmark optimization problems with 3- to 15-objectives, we compare the performance of SDEA with state-of-the-art many-objective evolutionary algorithms. The results indicate that SDEA outperforms existing algorithms in undertaking complex optimization problems with a high number of objectives, and has comparable outcomes over low-dimensional objective space benchmark problems.

1. Introduction

An optimization problem with more than three objectives is regarded as a many-objective optimization problem (MaOP) [36,37]. Currently, MaOPs are the subject of increased interest in the evolutionary multi-objective optimization (EMO) community. The trend towards MaOPs is encouraged by many real-world requirements, such as molecular design problem with 4 objectives [29], airfoil design problem with 6 objectives [44], flight control system design problem with 8 objectives [16], automotive engine calibration problem with 10 objectives [32], air traffic control tracking filters problem with 12 objectives [19], and complex industrial scheduling problem with 20 objectives [39]. Multi-objective evolutionary algorithms (MOEAs) based on the Pareto-dominance principle have been in existence for some time, such as non-dominated sorting genetic algorithm II (NSGA-II) [13], the strength Pareto evolutionary algorithm 2 (SPEA2) [49], and Pareto envelope-based selection algorithm II (PESA-II) [8]. These approaches are typically used to address 2- and 3-objective optimization problems. Many researchers have reported a myriad of challenges when applying these algorithms to MaOPs [30,36,37,42], which include ineffective convergence due to inefficient selection pressure, diversity preservation in a huge objective space, computationally expensive calculation of performance metrics, and visualization of the high-dimensional objective space.

* Corresponding author.
E-mail address: burhan.khan@deakin.edu.au (B. Khan).

https://doi.org/10.1016/j.ins.2018.09.031
0020-0255/

Considering these challenges, researchers in the EMO community are actively formulating improved optimization techniques. The experimental results using Pareto-dominance based MOEAs for undertaking MaOPs show the phenomenon of a poor selection pressure [30,42]. Due to the high-dimensional size of the MaOPs, the majority of the obtained solutions tend to be non-dominated when the solutions are ranked in accordance with the non-domination levels in the objective space. To overcome the selection pressure problem, researchers have proposed several methods to scale the Pareto-dominance technique to handle MaOPs, which include modified optimality techniques, new ranking methods, and other preference relation operators. Several modified forms of Pareto-dominance methods reduce the optimality conditions so as to tailor the optimality relation to MaOPs, e.g. ε-dominance [12] and α-dominance [22]. However, these variants of Pareto-dominance relation involve additional parameters and are usually problem specific. Another set of preference relation methods ranks the solutions in the objective space and then assigns the fitness value using a scalarization technique, e.g. p-optimality [24] and relation favor [15].

Solution proximity and diversity across the Pareto-front (PF) are two key criteria for multi-objective optimization algorithms. It is vital for an effective MOEA model to maintain a proper balance between proximity and diversity, in order to obtain a converged and diverse set of solutions with respect to the exact PF. Improved selection pressure contributes to convergence, but, due to the high-dimensional objective space, the solutions produced are usually far away from each other. These distant solutions are more likely to lead to worse solutions [37]. Therefore, it is essential for the diversity method to maintain variation and the proximity of the obtained solutions during the search process of MOEAs. Several studies focus on improving the diversity mechanism and improving the selection pressure [35,46]. While these existing methods are able to preserve the diversity of the obtained solutions in the objective space, their proximity to the PF cannot be guaranteed [17].

Indicator-based MOEAs are better alternatives to Pareto-dominance-based MOEAs for solving multi-objective optimization problems (MOPs) [5,48]. Indicator-based techniques determine the fitness of the solutions in the objective space using quality metrics. This approach is susceptible to the choice of the quality metric used for fitness estimation. The indicator-based evolutionary algorithm (IBEA) [48] formulates a general structure that incorporates the quality metrics in MOEAs [4]. Some of the notable MOEAs proposed in this vein use the popular S metric or hypervolume calculation metric. As an example, the S metric selection evolutionary multi-objective optimization algorithm (SMS-EMOA) [4] uses a hypervolume indicator in addition to the non-dominated sorting method proposed in [13]. To reduce the additional overhead due to the computation of the quality metric, SMS-EMOA adopts a steady-state selection scheme to produce only one individual in each generation. However, the performance of indicator-based MOEAs degrades with increasing number of objectives. Despite the computational overhead, researchers in the EMO community continue to propose faster ways for calculating the quality metrics. One such example is the hypervolume estimation algorithm (HypE) [3], which estimates the hypervolume metric using Monte Carlo simulation. However, MOPs with more than five objectives lead to a significant computational burden even with the use of faster methods [3,34].

Another effective approach to solving MOPs is decomposition of the objective space. Decomposition-based techniques solve multiple objectives simultaneously by converting the objective space to a set of multiple scalar problems using a scalarization technique. This framework was originally introduced in the multi-objective evolutionary algorithm based on decomposition (MOEA/D) [38]. Since then, variations have been proposed to improve on the limitations of MOEA/D [31,33,40]. However, the most prevailing weakness reported in the literature is a loss of diversity in the solutions when MOEA/D is applied to MaOPs [21,46].

Similar to decomposition-based MOEAs, reference point-based MOEAs use reference points or weight vectors as target vectors, as opposed to decomposition-based MOEAs which use weight vectors as directional vectors. Maintaining diversity is a key advantage in using reference points as target vectors. Having multiple target vectors in the objective space ensures the presence of solutions at the desired locations. A sufficiently diverse set of solutions comes at the price of reduced computation efficiency, owing to the increased number of pair-wise comparisons needed between the obtained solution points and the predetermined reference points. However, this approach is well suited for undertaking MaOPs [11]. A number of similar approaches exist [1,47], and more recently, reference point-based non-dominated sorting NSGA-III [11] and θ-dominance-based evolutionary algorithm (θ-DEA) [45] have been proposed to solving MaOPs. NSGA-III is an improved version of NSGA-II by using reference vectors with different selection mechanisms. On the other hand, θ-DEA has been proposed by considering both the positive and negative aspects of NSGA-III and MOEA/D models [45].

NSGA-III uses the Pareto-dominance principle in the fitness scheme combined with the idea of niching solutions close to the reference vectors. However, the Pareto-dominance scheme is known for its inability to converge efficiently in a high-dimensional objective space [30,42]. While MOEA/D is simple and effective owing to the use of a scalarization technique as the fitness function, it loses its diversity in high-dimensional and complex problems [21,46]. To harness the benefits of both MOEA approaches, θ-DEA adopts the fitness evaluation mechanism of MOEA/D and the reference vectors of NSGA-III to maintain diversity. It maintains a cluster of solutions allocated to the reference points, and the solutions within each cluster are evaluated using a penalty-based boundary intersection (PBI) [38] fitness scheme. However, θ-DEA still relies on the Pareto-dominance principle, in addition to the proposed θ-dominance, which is not efficient in tackling high-dimensional problems.

We propose a novel scalarization-based dominance evolutionary algorithm (SDEA) to address the shortcomings of existing MOEAs in undertaking MaOPs. The key contributions of the proposed SDEA model are three-fold: (i) it reduces the computational complexity and enables a faster execution with a lower demand on computational resources; (ii) it provides an improved convergence rate, as compared with those of existing models, in tackling MaOPs; (iii) it allows a decision maker

(DM) to incorporate their preferences in the search process and, at the same time, maintains a diverse set of solutions using the reference vectors.

Specifically, the computational complexity problem is overcome by proposing a novel ranking methodology based on scalarization, which is computationally inexpensive when compared with those from methods based on the Pareto-dominance principle. The proposed scalarization-based sorting method aims to increase the convergence rate. The proximity of the solutions to the reference vectors therein is systematically examined by replacing the commonly used Euclidean distance measure with an inner angle measure. This allows uniformly distributed reference vectors to be set by the DM, to incorporate their preferences into the search process.

This paper is organized as follows. The background and a review of MOPs, including relevant MOEAs to this research such as MOEA/D-PBI, NSGA-III, and θ-DEA, are presented in Section 2. The proposed SDEA model is explained in Section 3. An experimental study to evaluate the usefulness of SDEA is presented in Section 4. The results and discussion are given in Section 5. Conclusions are presented in Section 6.

2. Preliminaries

The basic concepts of MOPs are explained in this section, which is followed by a review of MOEA/D-PBI, NSGA-III, and θ-DEA algorithms, enabling readers to have a better understanding of the comparison results presented in Section 4.

2.1. Multi-objective optimization

A general form of MOP can be represented by Eqs. (1)–(4). Eq. (1) is a minimization function for M objectives where $x=(x_1, x_2,\ldots,x_n)$ represents the decision variables. Eq. (2) defines the lower and upper bounds for each decision variable in x. Eq (3) represents the inequality constraints, whereas Eq. (4) represents the equality constraints of the optimization problem.

$$min f(x) = f_1(x), f_2(x), f_3(x), \ldots, f_m(x) \tag{1}$$

$$x_i^l \le x_i \le x_i^u \quad \forall i = 1, 2, 3, \ldots, n \tag{2}$$

$$g_j(x) \ge 0 \quad \forall j = 1, 2, 3, \ldots, J \tag{3}$$

$$h_k(x) = 0 \quad \forall k = 1, 2, 3, \ldots, K \tag{4}$$

2.2. MOEA/D-PBI

A decomposition-based MOEA model was proposed in [38]. It divides a MOP into multiple single-objective optimization sub-problems using scalarization techniques such as weighted sum, Tchebycheff, and PBI methods [38]. Accordingly, a uniform set of weight vectors $\lambda=(\lambda_1, \lambda_2,\ldots,\lambda_N)^T$ is pre-determined. Then, a neighborhood matrix $B(i)=(i_1,i_2,\ldots,i_T)$ is populated by estimating the Euclidean distance for T neighboring weight vectors. This matrix allows comparison of the obtained fitness values with the neighboring weight vectors. The remaining part of the algorithm runs a choice of reproduction operator and a scalarization technique to obtain new solutions and evaluates the fitness of the newly obtained solution, respectively. However, for mating purposes, two solutions are randomly selected within the same neighborhood set $B(i)$. Inspired by the normal-boundary intersection method (NBI) [7] (explained in detail in Supplementary Materials), the authors suggested the PBI approach (as explained in Section 3.5) for fitness estimation.

2.3. NSGA-III

NSGA-III is an extension of its predecessor, NSGA-II. It addresses MaOPs with a significant modification to its selection mechanism [11]. NSGA-III adopts a set of pre-defined target reference points as a diversity preserving mechanism to overcome the shortcomings of the crowding distance measure when it is applied to MaOPs [11]. In general, NSGA-III relies on non-dominated Pareto solutions close to the reference points.

The first step in NSGA-III requires the generation of a set of uniformly spread reference points or user-provided aspiration points in the objective space. This is followed by producing an initial population of N number of randomly generated solutions. The parent population P_t is randomly selected. The subsequent process of producing the offspring population Q_t with simulated binary crossover (SBX) [9] and polynomial mutation [10] genetic operators is repeated until the stopping criterion is satisfied. The combined population R_t of size $2N$ using P_t and Q_t is classified as various non-dominated fronts $(F_1, F_2,\ldots,F_l)^T$ based on the Pareto-domination principle, where F_l is the last front. Thereafter, a non-dominated population S_t is formed by selecting the solutions from each front F_i one by one until $|S_t| \ge N$. The newly constructed population S_t is then assigned as the parent population P_{t+1} in the next generation if $|S_t| = N$. In a situation when $|S_t| > N$, S_t is filled till the second last front $S_t = F_{l-1}$. However, the remaining number of solutions, $k = N - |S_t|$, are chosen from the last front F_l to fill

in the empty spots in S_t, and the residual fronts are discarded. Since NSGA-III uses reference points to maintain diversity, it is necessary to normalize the solutions in the objective space and reference points to share a similar range such that the ideal point of the normalized objective space is a zero vector. The normalization process is conducted by erecting the reference lines from the origin through each reference point. Each solution in S_t is linked to the reference lines by comparing the minimum perpendicular distance to the reference line. The odds are high for more than one solution to be assigned to one reference line, while no solution may be associated with a reference line. Therefore, a niche-preserving mechanism is introduced, where the reference lines without any solutions are not considered in the existing generation, whereas those reference lines with multiple solutions are used to produce the next generation population P_{t+1}.

2.4. θ-DEA

The framework of θ-DEA is similar to NSGA-III, but with a different approach to obtaining PF based on the proposed θ-dominance concept, as opposed to Pareto-dominance in NSGA-III. Before estimating the fitness of the solutions using θ-dominance, the combined population R_t of current and offspring populations is sorted using non-dominated sorting. The PBI function is used as the scalarization function to determine the fitness of the solutions within each cluster. A major portion of the algorithm is similar to NSGA-III. Similarly, a set of reference points $\Lambda = (\lambda_1, \lambda_2, \ldots, \lambda_N)^T$ is initially computed using Das and Dennis's approach, where λ is an M-dimensional vector representing a point in the M-dimensional space. This is followed by initializing population P_0 and computing the ideal point z^* and nadir point z^{nad} from the initial population. Subsequently, the evolutionary process produces offspring solutions Q_t using a recombination operator which is applied until certain termination criteria are met. The obtained non-dominated solutions R_t after applying the recombination operator, are sorted into non-dominating fronts S_t using the non-dominating sorting method. The sorted solutions S_t are then normalized and clustered based on the Euclidean distance d between the solutions in S_t and reference points in Λ. Thereafter, the solutions are sorted into several fronts (F_1', F_2',…) based on θ-dominance sorting to obtain the next generation population P_{t+1}.

3. Proposed framework

In this section, we describe the proposed framework and the steps involved.

3.1. Algorithm overview

The proposed reference point-based SDEA framework is described in Algorithm 1. The framework commences by generating a set of pre-determined reference point's λ at step 2. This is followed by initializing the random population P_I in step 3 using the cardinality of reference points $|\lambda|$ as the population size N. Ideal point z^{ideal} and nadir point z^{nadir} are obtained in step 4 and step 5, respectively, by searching across objective values of every solution in population P_I. Furthermore, the values obtained from step 4 and step 5 are used in step 6, where the initial population P_I is normalized and translated to the first quadrant of the objective space. Step 7 prepares a list of archives $\vartheta_{\lambda,P}$ for the reference points in set λ, that holds the assigned solutions later. Steps 10–17 are repeated until the termination criterion is satisfied. Step 11 produces the offspring population P_{t+1} of size $2N$. Steps 12–14 normalize and translate the obtained offspring population P_{t+1} using the updated ideal point z^{ideal} and nadir point z^{nadir}. Thereafter, the obtained population P_{Norm} is ranked by assigning the respective reference point archives at step 15, and the resulting population P_t is obtained from the ranked fronts (F_1,F_2,…) at step 16, for use in the next generation.

Algorithm 2 assigns the solutions in the population P to the closest reference vector archive $\vartheta_{\lambda,P}$ and ranks the assigned solutions based on the scalarized non-domination levels, as shown in Fig. 3. Each reference vector ϑ_λ maintains an archive of the closest solutions. Steps 4–15 are repeated for each ith solution of P to find the nearby reference vector, based on the calculated inner angle $\angle\vartheta$ in steps 6 and 8, as illustrated in Eq. (5). Steps 4–6 assign the first solution P_1 to the first reference vector archive ϑ_P, whereas the remaining reference vectors are compared with the ith solution from steps 8–12. Subsequently, steps 9–12 update the minimum inner angle $\angle\vartheta_{min}$ of the ith solution in P, only when $\angle\vartheta_{min}$ is greater than the current inner angle $\angle\vartheta$ with respect to the jth reference vector of ϑ_λ. Steps 18–24 are repeated for each reference vector archive to rank the solutions in an ascending order into fronts (F_1,F_2,…) based on the calculated fitness using the scalarization technique at step 14.

Similar to the "Associate" procedure in NSGA-III, "Clustering operator" in crEA [6], and "Clustering" algorithm in θ-DEA, steps 6 and 8 in Algorithm 2 determine the proximity of the solutions based on the inner angle between the solution and the reference vector. Fig. 1, illustrates three different cases for measuring the proximity of a solution using angles and distances.

In Case 1, both solutions a and b have identical angles but dissimilar distances in the objective space. When comparing the solutions based on their corresponding distances to the reference vector, solution a dominates solution b, leading to the fact that solution b is discarded; however, based on angles, both solutions are non-dominating, and the first solution to appear in the list is maintained.

Case 2 highlights the scenario where solution b is favored over solution a based on having a smaller angle with respect to the reference vector; however, solution a is closer and dominates solution b. Using angles as the proximity measure, two

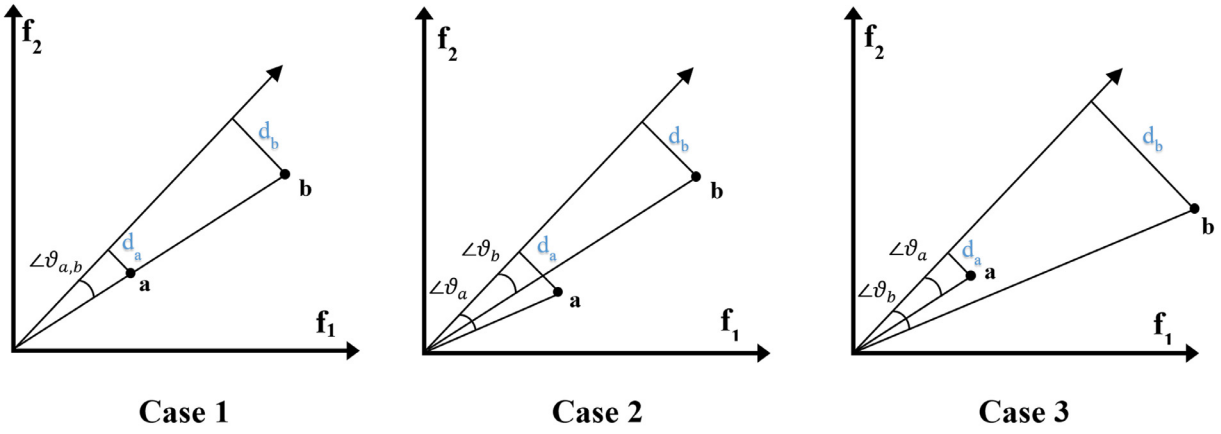

Fig. 1. Illustration of angle vs distance proximity to the reference vector.

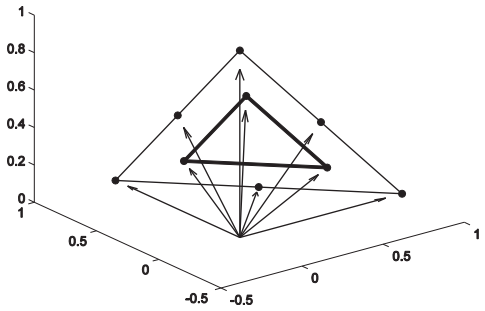

Fig. 2. Illustration of reference points generated in the 3D objective space for $H_1 = 2$ and $H_2 = 1$.

goals are achieved; solutions are maintained closer to the reference vectors and at the same time the dominated regions are exploited as well.

Finally, Case 3 illustrates the scenario where both distance and angle proximity measures favor solution a, due to the smaller angle $\angle\vartheta_a$ and the projected distance d_a.

$$\angle\vartheta = \cos^{-1}\left(\frac{\vec{\lambda}.\vec{f}}{\|\lambda\|.\|\vec{f}\|}\right) \tag{5}$$

Algorithm 3 returns the new population P_t for the reproduction purpose in step 10 of Algorithm 1. This new population is constructed by obtaining the solutions starting from the first front F_1 until the population size is reached.

3.2. Generation of the reference points

Decomposition and reference point-based approaches require a set of pre-determined reference points to decompose the objective space into several regions. Similar to MOEA/D, NSGA-III, and θ-DEA, our proposed SDEA model uses Das and Dennis's methodology for generating the direction vectors on a normalized hyper-plane. This mechanism produces reference points $\lambda = (\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_D)^T$, where the number of reference points or divisions D is subject to the positive integer H and dimensions of the optimization problem M in Eq. (6). It is important to note that the population size N is equivalent to D, which depends on the choice of the value used by the user for parameter H.

$$D = C_{M-1}^{H+M-1} \tag{6}$$

According to Eq. (6), the number of reference points increases exponentially with increasing number of objectives M and positive integer H. Therefore, we adopt a two-layered approach as proposed in [11] with smaller values of H_1 and H_2, as shown in Eq. (7) and Fig. 2.

$$D = C_{M-1}^{H_1+M-1} + C_{M-1}^{H_2+M-1} \tag{7}$$

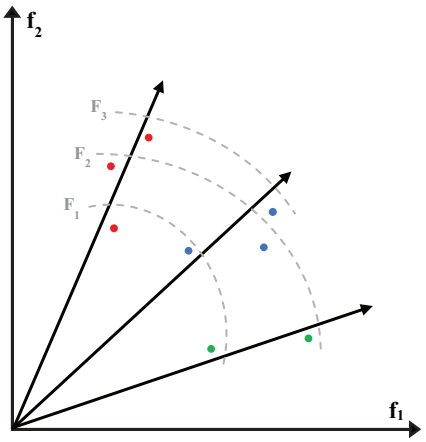

Fig. 3. Illustration of scalarization-based dominance. Each reference vector archive is represented by a distinct color for solutions and the dashed lines represent domination levels.

3.3. Offspring population

Traditional MOEAs face many challenges to solve many-objective optimization problems efficiently. This is due to solutions generated in a high-dimensional objective space reside far away from each other and may end up producing offspring solutions distant from the parent solutions. A more favorable approach is to produce offspring solutions close to the parent solutions [45]. Therefore, we use the SBX crossover and polynomial mutation recombination operators in SDEA. These genetic operators exploit the property of producing offspring solutions close to the parent solutions by setting the maximum value for the distribution index. It is implicit that the quality of solutions depends on the choice of the genetic operators used and the parameters employed.

3.4. Normalization and translation of objectives

Normalization of the objective space is required in SDEA to standardize the objective space pertaining to incongruent objectives, therefore supporting the optimization problems with different scaled objectives. In the normalization step, each objective value of population P is adaptively normalized at each generation using ideal point z^{ideal} and nadir point z^{nadir}, as illustrated in Eq. (8). Translation is preceded by normalization of objectives, where all solutions in P are translated to the first quadrant by subtracting z^{ideal} from the objective value of each solution of P in the objective space. This step allows formulating z^{ideal} to a zero origin vector.

$$\tilde{f}_i(x) = \frac{f_i(x) - z_i^{ideal}}{z_i^{nadir} - z_i^{ideal}} \tag{8}$$

Ideal point z^{ideal} is identified as the minimum of each objective however, estimation of z^{nadir} is not an easy task and requires knowledge of the attainable objective space. In this study, we adopt the methodology proposed in [11], for computation of extreme points to estimate z^{nadir}, with a minor modification of using normalized values in the achievement scalarizing function.

3.5. Fitness scheme

The proposed scalarization-based dominance methodology generally works with the scalar fitness values obtained from any scalarization technique. A number of well-known scalarization techniques are presented in Supplementary Materials. It has been suggested in [38] that out of the well-known scalarization techniques, the PBI method is suitable for MaOPs.

The PBI method allows the DM to deal with the degree of enforcing an equality constraint by introducing the penalty parameter θ into NBI, as given in Eq. (9) and illustrated in Fig. 4. The lower value of θ reduces the equality constraint, and the opposite is true for increasing θ. However, the obvious disadvantage of this method is the added complexity of finding the optimum value of θ.

$$PBI = d_1 + \theta d_2 \quad \text{where } d_1 = \frac{\left\| \tilde{f}_i(x)^T \lambda_i \right\|}{\|\lambda_i\|} \text{ and } d_2 = \left\| \tilde{f}_i(x) - d_1 \left(\frac{\lambda_i}{\|\lambda_i\|} \right) \right\| \tag{9}$$

According to the PBI method, the penalized distance θd_2 of perpendicular projection of the individual solution onto the reference vector and the distance d_1 along with the reference vector to the ideal point $z^{ideal}$are computed, where θ is the penalty parameter pre-defined by the user.

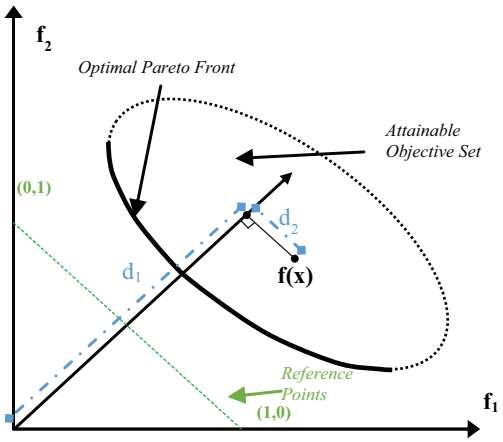

Fig. 4. Illustration of PBI approach in 2-dimensional objective space.

Table 1
Summary of differences between MOEAs.

Number	Similarities / Dissimilarities	MOEA/D	NSGA-III	θ-DEA	SDEA
i.	Diversity maintenance using reference vectors	✗	√	√	√
ii.	Fitness estimation using scalarization	√	✗	√	√
iii.	Use of Pareto-dominance principle	√	√	√	✗
iv.	Solution proximity to the reference vector	✗	Distance-based	Distance-based	Angle-Based
v.	Neighborhood selection	√	✗	✗	✗
vi.	Dominance criterion	Decomposition	Pareto-dominance	θ-dominance	Scalarization-based dominance
vii.	Computational complexity	$O(MNT)$ [41]	$O(N^2 logN)$ [11]	$O(MN^2)$ [45]	$O(M2N^2)$

In this study, we use the PBI method as the scalarization technique and set the suggested penalty parameter $\theta = 5$ according to Yuan et al. [45]. Further details regarding the choice of penalty parameters and their effect on the convergence rate are given in [45].

3.6. Computational complexity

A major portion of the computational complexity of the proposed algorithm for a single iteration is contributed to the assignment method in Algorithm 2. In this method, each solution in population P is assigned to a reference point using the fitness function, which requires $O(NM|P|)$ computation, where M is the number of objectives and N is the number of reference points. Since the proposed algorithm does not use non-dominated sorting of solutions at the PFs, a portion of the computational complexity of $O(M(2N)^2)$ [13] is avoided [11]. Moreover, the normalization step involving the determination of the ideal point and nadir point requires only $O(NM)$ for each generation. Considering the described computational complexities, the overall worst-case scenario complexity of SDEA is $O(M2N^2)$.

3.7. Difference between MOEAs

In this section, we present similarities and dissimilarities between the four main MOEAs (MOEA/D, NSGA-III, θ-DEA, and SDEA) considered in this study for comparison. Some of the major differences amongst these MOEAs are detailed in Table 1.

I. The use of reference points is common amongst all four MOEAs (MOEA/D, NSGA-III, θ-DEA, SDEA) in comparison. However, the logic behind the use of the reference points varies between these MOEAs. For instance, MOEA/D uses reference points as weight vectors to achieve two tasks; defines multiple scalar sub-problems by abandoning Pareto-dominance principle in the selection and restricts mating selection to solutions belonging to the neighboring reference points. On the other hand, reference-based MOEAs use reference points as target vectors to preserve diversity when tackling high-dimensional optimization problems.
II. NSGA-III uses the reference points only to maintain diversity. However, MOEA/D, θ-DEA, and SDEA also use these reference points as the mean of estimating fitness values in scalarization techniques.
III. MOEA/D, NSGA-III, and θ-DEA use Pareto-dominance principle to sort solutions into non-domination levels and discard the dominating solutions at every generation. However, SDEA abandons the Pareto-dominance principle and relies completely on the proposed scalarization-based dominance methodology that ranks the solutions based on the fitness values. This approach benefits SDEA in following ways.
    a Reduces the complexity added by the non-dominating sorting method at every generation.

Table 2
Reference points for the adopted benchmark test problems.

	DTLZ 1	DTLZ 2-4	WFG 1-9
z^{ideal}	(0.0,...,0.0)	(0.0,...,0.0)	(0.0,...,0.0)
z^{nadir}	(0.5,...,0.5)	(1.0,...,1.0)	(2,4,...,2M)

b Avoids the hazard of low selection pressure in high-dimensional optimization problems posed by Pareto-dominance principle.

IV. To preserve diversity in high-dimensional problems, NSGA-III and θ-DEA use Euclidean-based distance calculation method to assign solutions to their respective closest reference vector. However, SDEA proposes to calculate the proximity of solutions based on the inner angle between the solution and reference vector. Proximity estimation using angle calculation is better than Euclidean distance estimations for optimization problems that require normalization of objective space.

V. Unlike MOEA/D, other MOEAs in comparison do not adopt neighborhood mating selection restriction approach.

VI. All four MOEAs use separate methodologies to rank the solutions. SDEA proposes novel scalarization-based dominance methodology that eludes the concept of distinguishing between the solutions based on Pareto-dominance principle. Hence, SDEA establishes sort levels based on the fitness of the solutions obtained from any scalarization approach.

4. Experimental study

This section presents a detailed analysis of the proposed SDEA model using many-objective optimization benchmark problems. The results are compared with those from the most recent state-of-the-art multi-objective and many-objective optimization algorithms including MOEA/D-PBI, NSGA-III, and θ-DEA. As these algorithms share a similar nature of using reference points as reference directions, we adopt similar performance metrics and parameter settings to those given in [11,45], in order to have a fair comparison between SDEA and the aforementioned MOEAs.

4.1. Performance metrics

Performance metrics are used to compare optimization algorithms by evaluating the obtained Pareto-optimal solutions on known benchmark MOPs. These test problems vary in complexity and number of variables and objectives, providing a thorough qualitative analysis. The performance analysis results are compared with those from other MOEAs using similar parameter settings. As suggested in [11], we can locate the exact PF solutions precisely along the line of the reference vectors if the exact PF is known. Let $\Lambda = (\lambda_1, \lambda_2, \ldots, \lambda_N)^T$ be the set of N target reference points located at the exact PF surface. Then, $F = (f_1, f_2, \ldots f_N)^T$ is the obtained set of Pareto-optimal solutions for $j = 1, 2, \ldots, N$.

The inverse generational distance (*IGD*) [50] metric assesses both convergence and diversity of the obtained solutions with respect to the exact PF. In Eq. (10), the average sum of the minimum Euclidean distance is calculated between each obtained solution, v_j, and the nearest reference point, λ_i. A smaller IGD value indicates a better performance. One of the major disadvantages of this metric is the requirement of the exact PF.

$$IGD(F, \Lambda) = \frac{\sum_{i=1}^{|\lambda|} \min d\left(\lambda_i, f_j\right)}{|\lambda|} \tag{10}$$

Hypervolume (*HV*) computes the M-dimensional volume of the region enclosed by N number of solutions, $(f_1, f_2, \ldots f_N)^T$, obtained from an MOEA, with respect to the nadir point, z^{nadir}, in the objective space. As illustrated in Eq. (11), a larger value of this metric indicates dominance with respect to convergence and diversity. The main advantage of this metric is that, unlike IGD, knowledge of the exact PF is not required [35]. However, it is well-known that it requires a higher computational cost for optimization problems with more than 5 objectives [3,34].

$$HV = volume\left(\bigcup_{i=1}^{N} f_i\right) \quad \text{w.r.t.} z^{nadir} \tag{11}$$

Similar to [45], we set z^{nadir} to 1.1 z^{nadir} and use the z^{ideal} and z^{nadir} values from Table 2, to estimate hypervolume by adopting a fast hypervolume calculation metric [43], for up to 10 objectives. The slight increase in the reference point allows a better balance between the effects of convergence and diversity [2,23]. The steps taken to match the results of the study in [45] before calculating hypervolume are as follows.

- Discard points that do not dominate nadir point $1.1z^{nadir}$
- Normalize PF using z^{nadir} and z^{ideal}
- Invert normalized PF with 1.1 to make the problem a maximization problem for *HV* calculation

Table 3
Properties of the DTLZ1-4 and WFG1-9 benchmark problems.

Property				
Problem	Separability	Modality	Bias	Geometry
DTLZ1	Separable	Multimodal	Non-biased	Linear
DTLZ2	Separable	Unimodal	Non-biased	Concave
DTLZ3	Separable	Multimodal	Non-biased	Non-convex
DTLZ4	Separable	Unimodal	Biased (f_M-f_1 plane)	Non-convex
WFG1	Separable	Unimodal	Flat, polynomial	Mixed, convex
WFG2	Non-separable	Unimodal (f_1:M-1), multimodal (f_1:M)	Non-biased	Disconnected, convex
WFG3	Non-separable	Unimodal	Non-biased	Linear, degenerate
WFG4	Non-separable	Multimodal	Non-biased	Concave
WFG5	Separable	Deceptive	Non-biased	Concave
WFG6	Non-separable	unimodal	Non-biased	Concave
WFG7	Separable	Unimodal	Biased (parameter dependent)	Concave
WFG8	Non-separable	Unimodal	Biased (parameter dependent)	Concave
WFG9	Non-separable	Deceptive, multimodal	Biased (parameter dependent)	Concave

Table 4
Population size determined using divisions (H_1 and H_2) [11,45].

Objectives (M)	3	5	8	10	15
Divisions (H_1,H_2)	12,0	6,0	3,2	3,2	2,1
Population size (N)	91	210	156	275	135

Table 5
MOEA parameters [11,45].

Parameter Name		Value
Crossover probability		1
Distribution	SBX crossover	30
index	Polynomial mutation	20

Table 6
Number of generations for the benchmark test problems [11,45].

Problem\Objectives (M)	$M=3$	$M=5$	$M=8$	$M=10$	$M=15$
DTLZ1	400	600	750	1000	1500
DTLZ2	250	350	500	750	1000
DTLZ3	1000	1000	1000	1500	3000
DTLZ4	600	1000	1250	2000	3000
WFG 1-9	400	600	750	1000	1500

Table 7
IGD comparison values with NSGA-III on minimum, median, and maximum values.

M	DTLZ1		DTLZ2		DTLZ3		DTLZ4	
	SDEA	NSGA-III	SDEA	NSGA-III	SDEA	NSGA-III	SDEA	NSGA-III
3	7.512E−04	**4.880E−04**	**9.556E−04**	1.262E−03	**7.697E−04**	9.751E−04	**1.547E−04**	2.915E−04
	3.521E−03	**1.308E−03**	1.649E−03	**1.357E−03**	**3.862E−03**	4.007E−03	**2.066E−04**	5.970E−04
	3.959E−02	**4.880E−03**	8.697E−03	**2.114E−03**	3.952E−02	**6.665E−03**	**4.768E−04**	4.286E−01
5	1.030E−03	**5.116E−04**	**2.966E−03**	4.254E−03	**1.891E−03**	3.086E−03	**2.929E−04**	9.849E−04
	1.334E−03	**9.799E−04**	**4.359E−03**	4.982E−03	**3.791E−03**	5.960E−03	**3.506E−04**	1.255E−03
	3.232E−02	**1.979E−03**	1.076E−02	**5.862E−03**	1.361E−02	**1.196E−02**	**7.643E−04**	1.721E−03
8	4.159E−03	**2.044E−03**	**8.075E−03**	1.371E−02	**1.093E−02**	1.244E−02	**2.145E−03**	5.079E−03
	4.804E−03	**3.979E−03**	**9.189E−03**	1.571E−02	**1.827E−02**	2.375E−02	**2.876E−03**	7.054E−03
	1.037E−02	**8.721E−03**	**1.279E−02**	1.811E−02	**7.071E−02**	9.649E−02	**2.626E−01**	6.051E−01
10	4.598E−03	**2.215E−03**	**7.502E−03**	1.350E−02	**6.450E−03**	8.849E−03	**2.567E−03**	5.694E−03
	5.088E−03	**3.462E−03**	**8.866E−03**	1.528E−02	**8.368E−03**	1.188E−02	**2.988E−03**	6.337E−03
	1.267E−02	**6.869E−03**	**1.537E−02**	1.697E−02	8.920E−02	**2.083E−02**	**3.516E−03**	1.076E−01
15	5.443E−03	**2.649E−03**	**9.262E−03**	1.360E−02	**4.714E−03**	1.401E−02	**3.664E−03**	7.110E−03
	7.560E−03	**5.063E−03**	**1.082E−02**	1.726E−02	**6.793E−03**	2.145E−02	**5.145E−03**	3.431E−01
	2.457E−02	**1.123E−02**	**1.270E−02**	2.114E−02	**1.810E−02**	4.195E−02	**1.511E−01**	1.073E+00

Table 8
IGD comparison values with MOEA/D-PBI on minimum, median, and maximum values.

M	DTLZ1		DTLZ2		DTLZ3		DTLZ4	
	SDEA*	MOEA/D-PBI	SDEA*	MOEA/D-PBI	SDEA*	MOEA/D -PBI	SDEA*	MOEA/D -PBI
3	6.861E−04	**4.095E−04**	6.927E−04	**5.432E−04**	**6.352E−04**	9.773E−04	**1.290E−04**	2.929E−01
	1.933E−03	**1.495E−03**	8.586E−04	**6.406E−04**	**2.115E−03**	3.426E−03	**1.500E−04**	4.280E−01
	1.508E−02	**4.743E−03**	1.740E−03	**8.006E−04**	1.264E−02	**9.113E−03**	**1.687E−04**	5.234E−01
5	7.165E−04	**3.179E−04**	1.800E−03	**1.219E−03**	1.148E−03	**1.129E−03**	**2.376E−04**	1.080E−01
	8.356E−04	**6.372E−04**	2.107E−03	**1.437E−03**	**1.751E−03**	2.213E−03	**2.893E−04**	5.787E−01
	2.178E−03	**1.635E−03**	2.426E−03	**1.727E−03**	**4.012E−03**	6.147E−03	**4.366E−04**	7.348E−01
8	3.935E−03	**3.914E−03**	5.153E−03	**3.097E−03**	7.771E−03	**6.459E−03**	**2.189E−03**	5.298E−01
	4.426E−03	6.106E−03	5.874E−03	**3.763E−03**	**1.145E−02**	1.948E−02	**2.599E−03**	8.816E−01
	6.522E−03	8.537E−03	6.481E−03	**5.198E−03**	**1.596E−02**	1.123E + 00	**3.350E−03**	9.723E−01
10	4.125E−03	**3.872E−03**	5.285E−03	**2.474E−03**	4.941E−03	**2.791E−03**	**2.533E−03**	3.966E−01
	4.649E−03	5.073E−03	5.782E−03	**2.778E−03**	6.036E−03	**4.319E−03**	**2.817E−03**	9.203E−01
	5.431E−03	6.130E−03	6.651E−03	**3.235E−03**	**8.447E−03**	1.010E + 00	**3.285E−03**	1.077E + 00
15	**5.664E−03**	1.236E−02	5.493E−03	**5.254E−03**	**3.875E−03**	4.360E−03	**3.572E−03**	5.890E−01
	7.138E−03	1.431E−02	6.365E−03	**6.005E−03**	**5.475E−03**	1.664E−02	**5.060E−03**	1.133E + 00
	9.332E−03	1.692E−02	**8.257E−03**	9.409E−03	**8.168E−03**	1.260E + 00	**8.118E−03**	1.249E + 00

Table 9
IGD comparison values with θ-DEA on minimum, median, and maximum values.

M	DTLZ1		DTLZ2		DTLZ3		DTLZ4	
	SDEA	θ-DEA	SDEA	θ-DEA	SDEA	θ-DEA	SDEA	θ-DEA
3	7.512E−04	**5.665E−04**	**9.556E−04**	1.042E−03	**7.697E−04**	1.343E−03	**1.547E−04**	1.866E−04
	3.521E−03	**1.307E−03**	1.649E−03	**1.569E−03**	3.862E−03	**3.541E−03**	**2.066E−04**	2.506E−04
	3.959E−02	**9.449E−03**	8.697E−03	**5.497E−03**	3.952E−02	**5.528E−03**	**4.768E−04**	5.320E−01
5	1.030E−03	**4.432E−04**	2.966E−03	**2.720E−03**	**1.891E−03**	1.982E−03	2.929E−04	**2.616E−04**
	1.334E−03	**7.328E−04**	4.359E−03	**3.252E−03**	**3.791E−03**	4.272E−03	**3.506E−04**	3.790E−04
	3.232E−02	**2.138E−03**	1.076E−02	**5.333E−03**	**1.361E−02**	1.911E−02	7.643E−04	**4.114E−04**
8	4.159E−03	**1.982E−03**	8.075E−03	**7.786E−03**	1.093E−02	**8.769E−03**	**2.145E−03**	2.780E−03
	4.804E−03	**2.704E−03**	9.189E−03	**8.990E−03**	1.827E−02	**1.535E−02**	**2.876E−03**	3.098E−03
	1.037E−02	**4.620E−03**	1.279E−02	**1.140E−02**	7.071E−02	**3.826E−02**	2.626E−01	**3.569E−03**
10	4.598E−03	**2.099E−03**	**7.502E−03**	7.558E−03	6.450E−03	**5.970E−03**	**2.567E−03**	2.746E−03
	5.088E−03	**2.448E−03**	8.866E−03	**8.809E−03**	8.368E−03	**7.244E−03**	**2.988E−03**	3.341E−03
	1.267E−02	**3.935E−03**	1.537E−02	**1.020E−02**	8.920E−02	**2.323E−02**	**3.516E−03**	3.914E−03
15	5.443E−03	**2.442E−03**	9.262E−03	**8.819E−03**	**4.714E−03**	9.834E−03	**3.664E−03**	4.143E−03
	7.560E−03	8.152E−03	**1.082E−02**	1.133E−02	**6.793E−03**	1.917E−02	**5.145E−03**	5.904E−03
	2.457E−02	2.236E−01	**1.270E−02**	1.484E−02	**1.810E−02**	6.210E−01	1.511E−01	**7.680E−03**

4.2. Benchmark problems

The optimization problems introduced in this section are widely used in the literature to evaluate both MOPs and MaOPs. The Deb-Thiele-Laumanns-Zitzler (DTLZ) [14] and Walking Fish Group (WFG) [20] suites of benchmark test problems are scalable to any number of objectives and variables. In this study, we used DTLZ1-4 for $M = (3,5,8,10,15)$ objectives and decision variables n based on $M + K$-1, where K for DTLZ1 is set to 5 and 10 for DTLZ2-4. The WFG suite is evaluated for $M = (3,5,8,10)$ objectives with 24 decision variables, where the position parameter is set to M-1 and the remaining is set as the distance parameter. To have a fair comparison with MOEA/D-PBI, NSGA-III, and θ-DEA MOEAs, the settings of decision variables are adopted similar to the studies in [11,45]. Table 3 [14,20] summarizes the properties of DTLZ1-4 and WFG1-9 problems, based on separability, modality, bias, and geometry. The separability characteristic of the optimization problem represents the parameters that can be optimized independently. On the contrary, the non-separable problems represent more real-world like problems with interdependent parameters. The modality characteristic exemplifies the challenge of having multiple local optima in a problem, whereby MOEAs are likely to be stuck in these local optima. The bias property, as the name suggests, highlights the tendency of obtaining high-density solutions in one or multiple planes of the problem. Finally, the geometry property represents the shape of the exact PF.

The considered DTLZ suite problems are all characterized as separable. The exact PF for DTLZ1 is a linear hyperplane with a value of 0.5 for each objective and encompasses $(11^k - 1)$ local optima. The exact PFs for DTLZ2-4 are similar. They are a unit concave sphere. It is suggested in [35] that DTLZ2 is useful for evaluating the scalability of MOEAs in tackling MaOPs. DTLZ3 combines the multi-modality characteristics of DTLZ1 and the exact PF of DTLZ2. To explore the spread of the solutions produced by MOEAs, DTLZ4 is biased towards one of its objective planes.

The WFG suite comprises nine scalable problems with a wide range of complexities. The WFG1, WFG5, and WFG7 problems are relatively simple problems with separable parameters and no local optima. The shapes of WFG1-3 are a com-

Table 10
IGD comparison values with θ-DEA* on minimum, median, and maximum values.

M	DTLZ1		DTLZ2		DTLZ3		DTLZ4	
	SDEA*	θ-DEA*	SDEA*	θ-DEA*	SDEA*	θ-DEA*	SDEA*	θ-DEA*
3	6.861E−04	**3.006E−04**	**6.927E−04**	7.567E−04	**6.352E−04**	8.575E−04	**1.290E−04**	1.408E−04
	1.933E−03	**9.511E−04**	**8.586E−04**	9.736E−04	**2.115E−03**	3.077E−03	**1.500E−04**	1.918E−04
	1.508E−02	**2.718E−03**	1.740E−03	**1.130E−03**	1.264E−02	**5.603E−03**	**1.687E−04**	5.321E−01
5	7.165E−04	**3.612E−04**	**1.800E−03**	1.863E−03	1.148E−03	**8.738E−04**	**2.376E−04**	2.780E−04
	8.356E−04	**4.259E−04**	**2.107E−03**	2.146E−03	**1.751E−03**	1.971E−03	**2.893E−04**	3.142E−04
	2.178E−03	**5.797E−04**	2.426E−03	**2.288E−03**	**4.012E−03**	4.340E−03	4.366E−04	**3.586E−04**
8	3.935E−03	**1.869E−03**	**5.153E−03**	6.120E−03	7.771E−03	**6.493E−03**	**2.189E−03**	2.323E−03
	4.426E−03	**2.061E−03**	**5.874E−03**	6.750E−03	1.145E−02	**1.036E−02**	**2.599E−03**	3.172E−03
	6.522E−03	**2.337E−03**	**6.481E−03**	7.781E−03	1.596E−02	**1.549E−02**	**3.350E−03**	3.635E−03
10	4.125E−03	**1.999E−03**	**5.285E−03**	6.111E−03	**4.941E−03**	5.074E−03	**2.533E−03**	2.715E−03
	4.649E−03	**2.268E−03**	**5.782E−03**	6.546E−03	**6.036E−03**	6.121E−03	**2.817E−03**	3.216E−03
	5.431E−03	**2.425E−03**	**6.651E−03**	7.069E−03	8.447E−03	**7.243E−03**	**3.285E−03**	3.711E−03
15	5.664E−03	**2.884E−03**	**5.493E−03**	7.269E−03	**3.875E−03**	7.892E−03	**3.572E−03**	4.182E−03
	7.138E−03	**3.504E−03**	**6.365E−03**	8.264E−03	**5.475E−03**	9.924E−03	**5.060E−03**	5.633E−03
	9.332E−03	**3.992E−03**	**8.257E−03**	9.137E−03	**8.168E−03**	1.434E−02	8.118E−03	**6.562E−03**

Table 11
Comparison of SDEA with θ-DEA, MOEA/D-PBI and NSGA-III on the average hypervolume values.

Problem	M	Normalized			Non-normalized		
		NSGA-III	θ-DEA	SDEA	θ-DEA*	MOEA/D-PBI	SDEA*
DTLZ1	3	1.117600^	1.116137	**1.118630**	**1.118329**	1.116679	1.117332^
	5	1.577027^	1.576983	**1.577765**	**1.577892**	1.577632	1.577872^
	8	2.137837	2.137924^	**2.137962**	**2.137998**	2.136337	2.137947^
	10	**2.592792**	2.592719	2.592722^	**2.592696**	2.592233	2.592674^
DTLZ2	3	0.743523^	**0.743778**	0.742935	**0.744320**	0.744137^	0.743913
	5	1.303638	**1.306928**	1.306021^	**1.307368**	1.307343^	1.306870
	8	1.969096	**1.977904**	1.977325^	**1.978469**	1.978216	1.978219^
	10	2.508717	**2.514259**	2.514003^	2.514485^	**2.515040**	2.514349
DTLZ3	3	0.737407^	0.736938	**0.737497**	**0.738977**	0.736044	0.738183^
	5	1.301481	1.303987^	**1.305050**	1.305846^	1.303168	**1.306134**
	8	1.954336	**1.968943**	1.967863^	1.970805^	1.251873	**1.971727**
	10	2.508879	2.512662^	**2.512941**	2.514027^	2.406221	**2.514105**
DTLZ4	3	0.744634^	0.729265	**0.744812**	0.602951^	0.406020	**0.744835**
	5	1.308698	**1.308945**	1.308772^	**1.308934**	1.205512	1.308762^
	8	1.980236^	**1.980779**	1.976670	1.977231^	1.826489	**1.980730**
	10	2.515172	**2.515436**	2.515355^	**2.515468**	2.423727	2.515356^

bination of several geometries. WFG2-6 are characterized as non-separable, while WFG4-9 share the same concave exact PF.

4.3. Parameter settings

The parameter settings used by the three algorithms (MOEA/D-PBI, NSGA-III, and θ-DEA) published in [11,45] are adopted in our experiments for performance comparison with SDEA. Specifically, Table 4 shows the population sizes, Table 5 shows the main parameters (e.g. crossover and mutation operators), and Table 6 shows the number of generations. The use of the same parameter settings provides a fair comparison and analysis of the proposed SDEA model with those of MOEA/D-PBI, NSGA-III, and θ-DEA.

5. Results & discussion

In this section, we present the results of our proposed SDEA model with three state-of-the-art MOEA models, namely NSGA-III, θ-DEA, and MOEA/D-PBI, for both DTLZ and WFG benchmark optimization problems, as published in [11,45]. The statistical hypothesis test results between SDEA and NSGA-III, θ-DEA, and MOEA/D-PBI, are presented in Supplementary Materials. In addition, an extended performance comparison between SDEA and six other multi- and many-objective optimization methods are included in Supplementary Materials.

5.1. Comparisons on DTLZ test suite

Tables 7–10 presents the comparison results of the minimum, median, and maximum IGD values for the DTLZ suite of MOPs for 20 runs. The normalized version of the proposed SDEA model is compared with NSGA-III and θ-DEA, whereas the

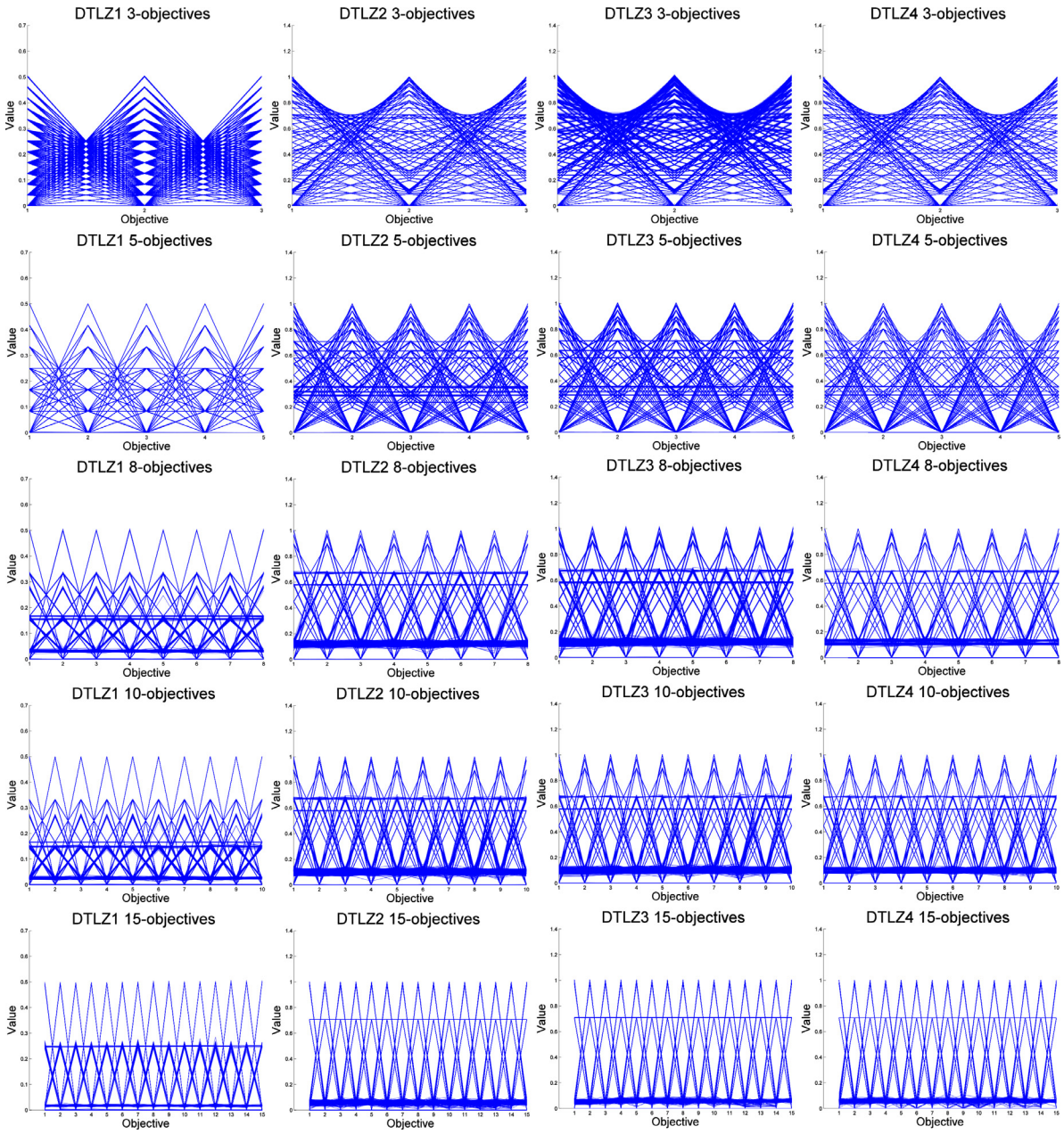

Fig. 5. Parallel coordinates of 3, 5, 8, 10, and 15 objectives (top to bottom) for DTLZ1-4 problems from left to right for the SDEA.

non-normalized version θ-DEA* and MOEA/D-PBI are compared with the non-normalized version of SDEA (SDEA*). The IGD and hypervolume values used in this study (Tables 7 –12) are taken from [11,45]. Better results are highlighted in bold.

We first consider the evaluation of SDEA for DTLZ1 instances, as presented in Tables 7 –11. The comparison results of SDEA with MOEA/D-PBI for instances of higher dimensions in Table 8 illustrate promising performance. However, SDEA shows poorer performance for all instances of DTLZ1 when compared with NSGA-III, θ-DEA, and θ-DEA* as in Tables 7,9, and 10, respectively. The inferior but comparable results are due to the inability of SDEA to converge faster in optimization problems with less number of objectives but large attainable objective space. In addition to the challenge of multiple local optima of DTLZ1, a second factor is the relatively larger attainable objective space. In other words, SDEA requires slightly more generations of problems with a larger and multi-modal attainable objective space to compete with the outcome of other MOEAs.

Comparing with DTLZ1, DTLZ2 is a simpler optimization problem with a smaller attainable objective space and no local optima. SDEA produce better results for all instances of DTLZ2 in comparison with NSGA-III and θ-DEA*, as illustrated in

Table 12
Comparison of SDEA with θ-DEA MOEA/D-PBI and NSGA-III on average hypervolume values.

Problem	M	Normalized			Non-normalized		
		NSGA-III	θ-DEA	SDEA	θ-DEA*	MOEA/D-PBI	SDEA*
WFG1	3	0.669729^	**0.704526**	0.574932	**0.697356**	0.657143^	0.168331
	5	0.859552	1.138794^	**1.153865**	1.236030^	**1.349888**	0.450609
	8	1.424963	**1.875997**	1.828494^	**1.905395**	1.755326^	1.656475
	10	2.249535	**2.364268**	2.333173^	**2.386742**	1.799394	2.309242^
WFG2	3	1.226956^	**1.227226**	1.212724	**1.221941**	1.111085	1.156610^
	5	**1.598410**	1.597188^	1.595842	1.564818^	1.520168	**1.579060**
	8	**2.136525**	2.124411	2.125146^	2.055014^	2.016854	**2.064650**
	10	**2.588104**	2.578311	2.578505^	2.491268^	2.459026	**2.492160**
WFG3	3	**0.819758**	0.814962^	0.806118	**0.798550**	0.757034^	0.597464
	5	1.013941	**1.028412**	1.024142^	**0.999933**	0.906075	0.997177^
	8	**1.221543**	1.147203	1.219939^	1.174530^	0.770754	**1.185483**
	10	1.567908^	**1.573090**	1.537988	**1.359598**	0.524917	1.351781^
WFG4	3	0.728892^	**0.729664**	0.706272	**0.720486**	0.685079	0.698463^
	5	1.285072^	**1.286861**	1.280045	**1.259362**	1.161435	1.254730^
	8	1.962156^	**1.964648**	1.962439	1.858132^	1.188847	**1.866635**
	10	2.502319	**2.504065**	2.502645^	2.232877^	1.432285	**2.251835**
WFG5	3	**0.687220**	0.687005^	0.675675	**0.676813**	0.656189	0.660409^
	5	1.222480^	**1.222746**	1.220825	**1.190345**	1.120619	1.188385^
	8	1.850281^	**1.850361**	1.849163	1.727167^	1.279934	**1.738739**
	10	**2.346581**	2.346521^	2.345897	2.092514^	1.541144	**2.094121**
WFG6	3	0.685939^	**0.690060**	0.664554	**0.679787**	0.654956^	0.641206
	5	1.219001^	**1.223099**	1.217614	**1.189960**	1.041593	1.188340^
	8	**1.843340**	1.841974^	1.835953	1.727171^	0.698152	**1.740619**
	10	2.326666	2.333417^	**2.334227**	2.011900^	0.811370	**2.071807**
WFG7	3	0.729030^	**0.731157**	0.695615	**0.722678**	0.619351	0.654220^
	5	1.291999^	**1.295864**	1.291682	**1.263840**	1.073783	1.255265^
	8	1.971529^	**1.973601**	1.971354	1.843617^	0.813288	**1.873784**
	10	2.507511	**2.508710**	2.508074^	2.280858^	0.950840	**2.285663**
WFG8	3	0.665932^	**0.666959**	0.645084	**0.655503**	0.633207	0.654220^
	5	1.18226^	**1.183904**	1.180984	1.147459^	0.968246	**1.255265**
	8	1.759882	1.768213^	**1.770010**	1.714995^	0.326124	**1.873784**
	10	2.280276	**2.297054**	2.292943^	2.198196^	0.255629	**2.246438**
WFG9	3	0.670081^	**0.680306**	0.637875	**0.671742**	0.564686	0.617136^
	5	1.212266^	**1.224104**	1.209240	**1.159929**	1.028928	1.143621^
	8	1.803989^	**1.842840**	1.823158	1.627886^	0.882226	**1.631428**
	10	2.32670	**2.364149**	2.361853^	**1.956897**	1.095281	1.948095^

Tables 7 and 9, except for the 10-objective problem in Table 10. However, SDEA produces inferior results in comparison with MOEA/D-PBI and θ-DEA, as deduced from Tables 8 and 9.

The combined complexities of DTLZ1 and DTLZ2 are found in DTLZ3, where the characteristics of a large attainable objective space and local optima are like those of DTLZ1, and the shape of the exact PF is similar to that of DTLZ2. SDEA performs better in comparison with NSGA-III and θ-DEA, in most instances. SDEA* clearly struggles in comparison with θ-DEA* and MOEA/D-PBI at some instance of the DTLZ3 problem. However, the overall outcome of SDEA over DTLZ3 is outstanding, and can still be related to DTLZ1 since SDEA* fails to obtain better results in some instances.

The exact PF shape of DTLZ4 is also like that of DTLZ2 and DTLZ3, but with an additional complexity of bias in the attainable objective space. The aim is to investigate the capability of MOEAs in finding the distribution of solutions. SDEA performs exceptionally well over DTLZ4, as illustrated in Tables 7–10.

Table 11 shows the average hypervolume results of 20 runs from SDEA by isolating the outcomes of the normalized and non-normalized versions. The highlighted values are the best results. The caret highlights the second best result out of three comparisons pertaining to each normalized and non-normalized version of MOEAs. The results show a competitive performance of SDEA. The results of other MOEAs are not significantly better, although θ-DEA for most of the cases in normalized and non-normalized versions performs better. However, SDEA outperforms NSGA-III and MOEA/D-PBI in most instances. As shown in Table 11, SDEA performs better in 21 out of 32 comparisons with other normalized MOEAs. However, SDEA* demonstrate comparable results for 18 out of 32 better results.

Fig. 5 depicts the parallel coordinate graphs of the PFs of DTLZ1-4 obtained from SDEA for 3, 5, 8, 10, and 15 objectives problems. These graphs clearly indicate convergence and uniformly distributed solutions along each objective.

5.2. Comparisons using the WFG test suite

Similar to Tables 11 and 12 depicts the experimental outcomes of SDEA for both normalized and non-normalized versions using the WFG test suite. The WFG test suite poses a more challenging set of optimization problems. In addition to

Algorithm 1 SDEA framework.

1. $M \leftarrow$ Number of Objectives
2. $\lambda \leftarrow$ *GenerateReferencePoints*(M)
3. $P_l \leftarrow$ *InitializePopulation*($|\lambda|$)
4. $z^{ideal} \leftarrow$ *FindIdealPoint*(P_l)
5. $z^{nadir} \leftarrow$ *FindNadirPoint*(P_l)
6. $P_t \leftarrow$ *Normalization* (z^{ideal}, z^{nadir}, P_l)
7. $\vartheta_{\lambda,P} \leftarrow$ *GetReferenceVectorArchives*(λ)
8. $t \leftarrow 1$
9. **do**
10. $O_{t+1} \leftarrow$ *RunEvolutionaryAlgorithm*(P_t)
11. $P_{t+1} \leftarrow O_{t+1} \cup P_t$
12. $z^{ideal} \leftarrow$ *UpdateIdealPoint*(O_{t+1})
13. $z^{nadir} \leftarrow$ *UpdateNadirPoint*(O_{t+1}))
14. $P_{Norm} \leftarrow$ *Normalization*(z^{ideal}, z^{nadir}, P_{t+1})
15. $[F_1, F_2, \ldots] \leftarrow$ *ScalarizationDominanceMethod*(P_{Norm}, ϑ_λ)
16. $P_t \leftarrow$ *GetPopulation*($[F_1, F_2, \ldots]$)
17. $t \leftarrow t + 1$
18. **while** the termination condition is not met;

Algorithm 2 ScalarizationDominanceMethod.

Input: P, ϑ_λ
Output: $[F_1, F_2, \ldots]$
1. $N \leftarrow$ *Size*(P)
2. $R \leftarrow$ *Size*(ϑ_λ)
3. **for** $i = 1 : N$ **do**
4. $j \leftarrow 1$
5. $ind \leftarrow 1$
6. $\angle\vartheta_{min} \leftarrow$ *CalculateAngle*($\vartheta_{\lambda_j}, P_i$)
7. **for** $j = 2 : R$ **do**
8. $\angle\vartheta \leftarrow$ *CalculateAngle*($\vartheta_{\lambda_j}, P_i$)
9. **if** $\angle\vartheta_{min} > \angle\vartheta$ **then**
10. $\angle\vartheta_{min} \leftarrow \angle\vartheta$
11. $ind \leftarrow j$
12. **end if**
13. **end for**
14. $P_{i_{fitness}} \leftarrow$ *Scalarization*($\vartheta_{\lambda_{ind}}$, P_i)
15. $\vartheta_{P_{ind}} \leftarrow \vartheta_{P_{ind}} \cup P_i$
16. **end for**
17. **for** $i = 1 : R$ **do**
18. $N \leftarrow$ *Size*($\vartheta_{\lambda_i,P}$)
19. **if** $N > 0$ **then**
20. $\vartheta_{\lambda,P} \leftarrow$ *AscendingSort*($\vartheta_{\lambda_i,P}$)
21. **for** $j = 1 : N$ **do**
22. $F_j = \vartheta_{P_{i,j}}$
23. **end for**
24. **end if**
25. **end for**

Algorithm 3 *GetPopulation*.

Input: $[F_1, F_2, \ldots]$
Output: P_t
1. $R \leftarrow$ *Size*(ϑ_λ)
2. $ind \leftarrow 1$
3. $N \leftarrow R$
4. **while** $R > 0$ **do**
5. $length = R$
6. **if** $R >$ *Size*(F_{ind}) **then**
7. $length =$ *Size*(F_{ind})
8. **end if**
9. **for** $i = 1 : length$ **do**
10. $P_t \leftarrow P_t \cup F_{ind_i}$
11. **end for**
12. $R \leftarrow N$
13. $R \leftarrow R$ - *Size*(P_t)
14. **if** $R > 0$ **then**
15. ind++
16. **end if**
17. **end while**

difficulties posed by the DTLZ suite, some of the complexities of the WFG test suite include degenerated, non-separable, scaled, deceptive, mixed, and discontinued PF, as revealed in Table 3. As shown in Table 12, the non-normalized version of SDEA performs better in 47 out of 72 problems as compared with those from both θ-DEA* and MOEA/D-PBI. However, a relatively poor performance is observed by the normalized version of SDEA, with only 16 out of 72 results better than those from θ-DEA and NSGA-III. This decline in performance is owing to SDEA not having an efficient mechanism to estimate the z^{nadir} point of the problem during the normalization step.

The attainable objective space of WFG1 is biased and is composed of convex and concave geometries. According to Table 12, SDEA performs well in comparison with NSGA-III, θ-DEA*, and MOEA/D-PBI. In addition, it is palpable that the performance of SDEA for WFG1 improves with increasing number of objectives. However, it is evident from Table 12 that SDEA* fails to converge for 3- and 5-objective instances. The exact PF shape of WFG2 is disconnected, and the problem is also composed of non-separable variables. The WFG2 problem is complex in nature with disconnected PFs. As opposed to SDEA, SDEA* produce an exceptionally good outcome in comparison with both MOEA/D-PBI and θ-DEA*. As opposed to WFG2, WFG3 consists of connected linear and degenerated exact PF. Once again, the normalized version of SDEA produces comparable results. However, SDEA* outperforms both MOEA/D-PBI in most cases and θ-DEA* in an 8-objective instance.

The problems WFG4 to WFG9 share the same concave exact PF, which represents a portion of the hyper-ellipse with radii 2i, $i = (1,2,\ldots,M)$, but with different properties in the decision space. WFG4 consists of multiple local optima, which presents a challenge to MOEAs. Nevertheless, SDEA* outperforms all algorithms in most instances due to the smaller attainable objective space. As a deceptive problem, WFG5 poses a challenge for the normalized version of SDEA. This can be deduced from the results in Table 12, where SDEA exhibits an ordinary performance in all instances of WFG5 except the comparison between SDEA* and MOEA/D-PBI, and the comparison in higher dimensions between SDEA* and θ-DEA*. WFG6 is non-separable and has a reduced attainable objective space. Similar to WFG5, SDEA's performance is competitive. However, SDEA* obtains comparable results in most low dimensional objective instances and better results in high-dimensional objective instances. WFG7 is a unimodal, separable, and biased optimization problem. Like the outcome of WFG6, SDEA* remains dominant over MOEA/D-PBI, but is comparable with θ-DEA* in most instances. The results of SDEA* on WFG8 are outstanding and satisfactory on WFG9, both problems are non-separable and biased, with WFG9 being deceptive too. According to Table 12, due to the lack of better z^{nadir} estimation mechanism, SDEA does not outperform other MOEAs. However, it displays a highly competitive nature. On the contrary, SDEA* presents outstanding outcomes for higher dimensions of all WFG problems.

6. Conclusions

In this paper, we have proposed a scalarization-based dominance evolutionary algorithm SDEA for addressing MaOPs. The major contributions of the SDEA are reduced computational complexity, diversity sustenance, and better convergence capability when compared with existing MOEAs in undertaking MaOPs.

SDEA replaces the computationally expensive non-dominated sorting, typically found in existing MOEAs, with a novel and more efficient sorting method based on fitness values obtained from scalarization. Additionally, it utilizes reference points in the decomposed objective space to act as target vectors for the solutions for sustaining their diversity and providing improved convergence rates.

To evaluate SDEA, IGD and hypervolume indicators have been used for comparison with state-of-the-art MOEAs in solving the DTLZ and WFG benchmark optimization problems. Results demonstrated the ability of SDEA to obtain a well-converged and well-distributed set of solutions. The performance of SDEA is comparable to state-of-the-art MOEAs in solving optimization problems with low-dimensional objective spaces and relatively less complexity. SDEA performs better when applied to optimization problems with high-dimensional objective spaces as the compared methods mainly rely on the non-dominated sorting of solutions based on the Pareto-dominance principle.

In our future work, we will explore the potential of information sharing between the neighboring reference points [26,27] and derive a scalarization technique based on angle proximity. Furthermore, we will test the proposed framework with real-world case studies [18,25,28].

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.ins.2018.09.031.

References

[1] Z. Aimin, Z. Qingfu, J. Yaochu, B. Sendhoff, Combination of EDA and DE for continuous biobjective optimization, in: *Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence). IEEE Congress* on, 2008, pp. 1447–1454.
[2] A. Auger, J. Bader, D. Brockhoff, E. Zitzler, Theory of the hypervolume indicator: optimal μ-distributions and the choice of the reference point, in: Proceedings of the tenth ACM SIGEVO workshop on Foundations of genetic algorithms, Orlando, Florida, USA, 2009.
[3] J. Bader, E. Zitzler, Hype: an algorithm for fast hypervolume-based many-objective optimization, Evol. Comput. 19 (1) (2011) 45–76.
[4] N. Beume, B. Naujoks, M. Emmerich, SMS-EMOA: multiobjective selection based on dominated hypervolume, Eur. J. Oper. Res. 181 (3) (2007) 1653–1669.
[5] D. Brockhoff, T. Friedrich, F. Neumann, Analyzing hypervolume indicator based algorithms, in: G. Rudolph, T. Jansen, N. Beume, S. Lucas, C. Poloni (Eds.), Parallel Problem Solving from Nature – PPSN X, Lecture Notes in Computer Science, 5199, Springer Berlin Heidelberg, 2008, pp. 651–660.

[6] L. Cai, S. Qu, Y. Yuan, X. Yao, A clustering-ranking method for many-objective optimization, Appl. Soft Comput. 35 (Suppl C) (2015) 681–694.
[7] I. Das, J.E. Dennis, Normal-boundary intersection: a new method for generating the pareto surface in nonlinear multicriteria optimization problems, SIAM J. Optim. 8 (3) (1998) 631–657.
[8] DavidW. Corne, N.R. J., JoshuaD. Knowles, MartinJ. Oates, J Martin, PESA-II: region-based selection in evolutionary multiobjective optimization, in: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2001), 2001.
[9] K. Deb, R.B. Agrawal, Simulated binary crossover for continuous search space, Complex Syst. 9 (2) (1995) 115–148.
[10] K. Deb, M. Goyal, A combined genetic adaptive search (GeneAS) for engineering design, Comput. Sci. Inform. 26 (4) (1996) 30–45.
[11] K. Deb, H. Jain, An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints, Evol. Comput., IEEE Trans. 18 (4) (2014) 577–601.
[12] K. Deb, M. Mohan, S. Mishra, Evaluating the ϵ-domination based multi-objective evolutionary algorithm for a quick computation of pareto-optimal solutions, Evol. Comput. 13 (4) (2005) 501–525.
[13] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, Evol. Comput., IEEE Trans. 6 (2) (2002) 182–197.
[14] K. Deb, L. Thiele, M. Laumanns, E. Zitzler, Scalable test problems for evolutionary multiobjective optimization, in: A. Abraham, L. Jain, R. Goldberg (Eds.), Evolutionary Multiobjective Optimization, Advanced Information and Knowledge Processing, Springer London, 2005, pp. 105–145.
[15] N. Drechsler, R. Drechsler, B. Becker, Multi-objective optimisation based on relation favour, in: E. Zitzler, L. Thiele, K. Deb, C. Coello Coello, D. Corne (Eds.), Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science, 1993, Springer Berlin Heidelberg, 2001, pp. 154–166.
[16] P. Fleming, R. Purshouse, R. Lygoe, Many-objective optimization: an engineering design perspective, in: C. Coello Coello, A. Hernández Aguirre, E. Zitzler (Eds.), Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science, 3410, Springer Berlin Heidelberg, 2005, pp. 14–32.
[17] D.B. Fogel, in: Evolutionary Computation: Toward a New Philosophy of Machine Intelligence, IEEE Press, 1995, p. 272.
[18] S. Hanoun, B. Khan, M. Johnstone, S. Nahavandi, D. Creighton, An effective heuristic for stockyard planning and machinery scheduling at a coal handling facility, in: 2013 11th IEEE International Conference on Industrial Informatics (INDIN), 2013, pp. 206–211.
[19] J.G. Herrero, A. Berlanga, J.M.M. Lopez, Effective evolutionary algorithms for many-specifications attainment: application to air traffic control tracking filters, Evol. Comput., IEEE Trans. 13 (1) (2009) 151–168.
[20] S. Huband, L. Barone, L. While, P. Hingston, A scalable multi-objective test problem toolkit, in: C.A. Coello Coello, A. Hernández Aguirre, E. Zitzler (Eds.), Evolutionary Multi-Criterion Optimization: Third International Conference, EMO 2005, Guanajuato, Mexico, March 9-11, 2005. Proceedings, Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 280–295.
[21] L. Hui, Z. Qingfu, Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II, IEEE Trans. Evol. Comput. 13 (2) (2009) 284–302.
[22] K. Ikeda, H. Kita, S. Kobayashi, Failure of Pareto-based MOEAs: does non-dominated really mean near to optimal? in: Evolutionary Computation, 2001. Proceedings of the 2001 Congress on, 2, 2001, pp. 957–962.
[23] H. Ishibuchi, Y. Hitotsuyanagi, N. Tsukamoto, Y. Nojima, Many-objective test problems to visually examine the behavior of multiobjective evolution in a decision space, in: Proceedings of the 11th International Conference on Parallel Problem Solving From Nature: Part II, Kraków , Poland, 2010.
[24] E.C. Jara, Multi-objective optimization by using evolutionary algorithms: the p-optimality criteria, Evol. Comput., IEEE Trans. 18 (2) (2014) 167–179.
[25] B. Khan, A. Bhatti, M. Johnstone, S. Hanoun, D. Creighton, S. Nahavandi, Optimal feature subset selection for neuron spike sorting using the genetic algorithm, in: S. Arik, T. Huang, K.W. Lai, Q. Liu (Eds.), Neural Information Processing: 22nd International Conference, ICONIP 2015, Istanbul, Turkey, November 9-12, 2015, Proceedings, Part II, Springer International Publishing, Cham, 2015, pp. 364–370.
[26] B. Khan, M. Johnstone, S. Hanoun, C.P. Lim, D. Creighton, S. Nahavandi, Improved NSGA-III using neighborhood information and scalarization, in: 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2016, pp. 003033–003038.
[27] B. Khan, S. Hanoun, M. Johnstone, C.P. Lim, D. Creighton, S. Nahavandi, A new decomposition-based evolutionary framework for many-objective optimization, in: 2017 Annual IEEE International Systems Conference (SysCon), 2017, pp. 1–7.
[28] B. Khan, S. Hanoun, M. Johnstone, C.P. Lim, D. Creighton, S. Nahavandi, Multi-objective job shop scheduling using I-NSGA-III, in: 2018 Annual IEEE International Systems Conference (SysCon), 2018, pp. 1–5.
[29] J. Kruisselbrink, M.M. Emmerich, T. Bäck, A. Bender, A. Ijzerman, E. van der Horst, Combining aggregation with pareto optimization: a case study in evolutionary molecular design, in: M. Ehrgott, C. Fonseca, X. Gandibleux, J.-K. Hao, M. Sevaux (Eds.), Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science, 5467, Springer Berlin Heidelberg, 2009, pp. 453–467.
[30] M. Li, S. Yang, X. Liu, R. Shen, A Comparative Study on Evolutionary Algorithms for Many-Objective Optimization, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 7811, 2013, pp. 261–275.
[31] K. Liangjun, Z. Qingfu, R. Battiti, MOEA/D-ACO: a multiobjective evolutionary algorithm using decomposition and antcolony, Cybern., IEEE Trans. 43 (6) (2013) 1845–1859.
[32] R. Lygoe, M. Cary, P. Fleming, A real-world application of a many-objective optimisation complexity reduction process, in: R. Purshouse, P. Fleming, C. Fonseca, S. Greco, J. Shaw (Eds.), Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science, 7811, Springer Berlin Heidelberg, 2013, pp. 641–655.
[33] X. Ma, F. Liu, Y. Qi, M. Gong, M. Yin, L. Li, L. Jiao, J. Wu, MOEA/D with opposition-based learning for multiobjective optimization problem, Neurocomputing 146 (2014) 48–64.
[34] A. Menchaca-Mendez, C. Coello Coello, GD-MOEA: a new multi-objective evolutionary algorithm based on the generational distance indicator, in: A. Gaspar-Cunha, C. Henggeler Antunes, C.C. Coello (Eds.), Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science, 9018, Springer International Publishing, 2015, pp. 156–170.
[35] L. Miqing, Y. Shengxiang, L. Xiaohui, Shift-based density estimation for pareto-based algorithms in many-objective optimization, Evol. Comput., IEEE Trans. 18 (3) (2014) 348–365.
[36] R.C. Purshouse, P.J. Fleming, Evolutionary many-objective optimisation: an exploratory analysis, in: Evolutionary Computation, 2003. CEC '03. The 2003 Congress on, 3, 2003, pp. 2066–2073.
[37] R.C. Purshouse, P.J. Fleming, On the evolutionary optimization of many conflicting objectives, Evol. Comput., IEEE Trans. 11 (6) (2007) 770–784.
[38] Z. Qingfu, L. Hui, MOEA/D: a multiobjective evolutionary algorithm based on decomposition, Evol. Comput., IEEE Trans. 11 (6) (2007) 712–731.
[39] A. Sülflow, N. Drechsler, R. Drechsler, Robust multi-objective optimization in high dimensional spaces, in: S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, T. Murata (Eds.), Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science, 4403, Springer Berlin Heidelberg, 2007, pp. 715–726.
[40] Y.-y. Tan, Y.-c. Jiao, H. Li, X.-k. Wang, MOEA/D + uniform design: a new version of MOEA/D for optimization problems with many objectives, Comput Oper. Res. 40 (6) (2013) 1648–1660.
[41] J.Y. Tey, R. Ramli, Comparison of computational efficiency of MOEA\D and NSGA-II for passive vehicle suspension optimization, in: Proceedings - 24th European Conference on Modelling and Simulation, ECMS 2010, 2010, doi:10.7148/2010-0219-0225.
[42] C. von Lücken, B. Barán, C. Brizuela, A survey on multi-objective evolutionary algorithms for many-objective problems, Comput.Optimization Appl. 58 (3) (2014) 707–756.
[43] L. While, L. Bradstreet, L. Barone, A fast way of calculating exact hypervolumes, Evol. Comput., IEEE Trans. 16 (1) (2012) 86–95.
[44] U.K. Wickramasinghe, R. Carrese, L. Xiaodong, Designing airfoils using a reference point based evolutionary many-objective particle swarm optimization algorithm, in: Evolutionary Computation (CEC), 2010 IEEE Congress on, 2010, pp. 1–8.
[45] Y. Yuan, H. Xu, B. Wang, X. Yao, A new dominance relation-based evolutionary algorithm for many-objective optimization, IEEE Trans. Evol. Comput. 20 (1) (2016) 16–37.
[46] H. Zhenan, G.G. Yen, Diversity improvement in decomposition-based multi-objective evolutionary algorithm for many-objective optimization problems, in: *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference* on, 2014, pp. 2409–2414.

[47] A. Zhou, Q. Zhang, G. Zhang, Approximation model guided selection for evolutionary multiobjective optimization, in: R. Purshouse, P. Fleming, C. Fonseca, S. Greco, J. Shaw (Eds.), Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science, 7811, Springer Berlin Heidelberg, 2013, pp. 398–412.
[48] E. Zitzler, S. Künzli, et al.X. Yao, et al. (Eds.), Indicator-based selection in multiobjective search, Parallel Problem Solving from Nature - PPSN VIII, 3242 (2004) 832–842.
[49] E. Zitzler, M. Laumanns, L. Thiele, SPEA2: improving the strength pareto evolutionary algorithm for multiobjective optimization, Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems, 2001.
[50] E. Zitzler, L. Thiele, M. Laumanns, C.M. Fonseca, V.G. da Fonseca, Performance assessment of multiobjective optimizers: an analysis and review, Evol. Comput., IEEE Trans. 7 (2) (2003) 117–132.