Differential Evolution for Multi-Modal Multi-Objective Problems

Monalisa Pal[*]
Machine Intelligence Unit, Indian Statistical Institute,
203 Barrackpore Trunk Road, Kolkata - 700108, India
monalisap90@gmail.com

Sanghamitra Bandyopadhyay
Machine Intelligence Unit, Indian Statistical Institute,
203 Barrackpore Trunk Road, Kolkata - 700108, India
sanghami@isical.ac.in

ABSTRACT

Multi-modal multi-objective problems (MMMOPs) have two or more distinct Pareto-optimal sets (PSs) mapping to the same Pareto-front (PF). Identifying all such PSs assists in informed decision-making. However, existing multi-objective evolutionary algorithms are not equipped to discover multiple PSs. Recently, a few studies have been conducted to design algorithms for such MMMOPs. However, the diversity of the solutions in the PF, obtained by these algorithms, are poor. Moreover, two effective strategies, identified to address MMMOPs, are niching methods and population filtering, based on convergence and diversity of solutions in PF along with diversity of solutions in PS. Motivated by these requirements, this study presents Differential Evolution for MMMOPs (DE-TriM). Its novel contributions include mating pool selection strategy and resource allocation scheme based on reference vector based decomposition of objective space. The effectiveness of DE-TriM is validated by its performance analysis on 11 benchmark MMMOPs in terms of four performance measures as compared to three recent optimization algorithms. The results demonstrate similar performance of DE-TriM in decision space and its superior performance in objective space as compared to the state-of-the-art multi-modal multi-objective evolutionary algorithm.

CCS CONCEPTS

• **Theory of computation → Evolutionary algorithms**; Optimization with randomized search heuristics; • **Computing methodologies →** *Optimization algorithms*;

KEYWORDS

Multi-Modal Multi-Objective Problems, Pareto-dominance, Reference Vector based Decomposition, Decision Space, Objective Space

ACM Reference Format:

Monalisa Pal and Sanghamitra Bandyopadhyay. 2019. Differential Evolution for Multi-Modal Multi-Objective Problems. In *Genetic and Evolutionary Computation Conference Companion (GECCO '19 Companion), July 13–17, 2019, Prague, Czech Republic.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3319619.3326862

[*]Corresponding author.

GECCO '19 Companion, July 13–17, 2019, Prague, Czech Republic

ACM ISBN 978-1-4503-6748-6/19/07…$15.00
https://doi.org/10.1145/3319619.3326862

1 INTRODUCTION

Multi-objective optimization problems have two or more conflicting optimization criteria (objectives) [3]. Formally, a box-constrained multi-objective minimization problem [3, 9] (Eq. (1)) maps a n-dimensional vector ($X = [x_1, \cdots, x_n]$) from the decision space (Ω) to a M-dimensional vector ($F(X)$) in the objective space ($\mathbb{R}^M$).

$$\begin{aligned} &\text{Minimize } F(X) = [f_1(X), f_2(X), \cdots, f_M(X)] \\ &\text{where, } X \in \Omega \left(\subseteq \mathbb{R}^n\right), F(X) : \Omega \mapsto \mathbb{R}^M \\ &\text{and } \Omega : x_j^L \leq x_j \leq x_j^U, \forall j = 1, 2, \cdots, n \end{aligned} \tag{1}$$

Two vectors are compared by Pareto-dominance relation. X Pareto-dominates Y by Eq. (2). If $\nexists X \in \Omega$ that dominates a solution $X^\star \in \Omega$, then $X^\star$ is a Pareto-optimal solution. A set Pareto-optimal solution vectors and their corresponding objective vectors form Pareto-optimal set (PS) and Pareto-front (PF), respectively.

$$\begin{aligned} X \prec Y \iff &\left(f_i(X) \leq f_i(Y) \land f_j(X) < f_j(Y)\right), \\ &\forall i \in \{1, 2, \cdots, M\}, \text{ and } \exists j \in \{1, 2, \cdots, M\} \end{aligned} \tag{2}$$

However, for a Multi-Modal Multi-Objective Problem (MMMOP), more than one decision vectors ($X_1, X_2, \cdots, X_k$) maps to the same objective vector $F(X) = F(X_1) = F(X_2) = \cdots = F(X_k)$. This leads to the following challenges for an Evolutionary Algorithm (EA) to optimize a MMMOP:

(1) Maintenance of convergence and diversity in the objective space while maintaining diversity in the decision space (or specifically, in each PS).
(2) Requirement of a large population to efficiently represent the MMMOP. For example, if each point on the PF can be mapped from k (e.g. 9 for a SYM-PART simple problem) PSs, and p points (e.g. 100) are required to represent the PF, then the population size required to represent the final solution is $k \times p$ (e.g. $9 \times 100 = 900$).

Let a real-world path-planning problem [18] be considered which aims at finding the trade-off routes minimizing travelling time and number of stops between origin and destination. There can be multiple routes requiring the same time (f_1) and having same number of stops (f_2). However, the communal facilities (like availability of a motel, a gas station, a night patrol, etc.) associated with each route can be different which, in turn, will affect the final choice of the decision-maker. This example highlights the practical importance of finding multiple PSs associated with the same PF which allows the end-users to make an informed decision. Such MMMOPs are challenging for the three kinds of existing Multi-Objective Evolutionary Algorithms (MOEAs): Pareto-dominance based algorithms (e.g. NSGA-II [7]), indicator-based algorithms (e.g. HypE [1]) and decomposition-based algorithms (e.g. MOEA/D [19]).

Designing EAs for such MMMOP are still at its infancy. Studies on analyzing the solution distribution in the decision space were performed long ago. Such notable works are as follows. Omni-optimizer [8] introduced the concept crowding distance in the decision space. Another work [2] considered diversity preservation based on neighborhood count. However, the computation of neighborhood radius was computationally expensive. The third work [20] designed a probabilistic model to approximate PSs and PF but this model fails when PS is a linear manifold. Within the last four years, the trend to design such EAs for MMMOP has regained attention. Recent works include DN-NSGA-II [11] which, like Omni-optimizer, performs non-dominated sorting accompanied by crowding distance in decision space. Another recent work is MOEA/D-AD [17] where the concept of (almost) equivalent Pareto-optimal solutions is introduced. Another recent algorithm (TriMOEA_TA&R [12]) is beneficial when convergence related decision variables forms a smaller subspace than the complete decision space. Finally, the state-of-the-art algorithm for MMMOP is the MO_Ring_PSO_SCD [18] which emphasized that along with diversity preservation during population filtering, niching methods (like ring topology) plays a crucial role. For example, MO_PSO with the ring topology produces much more diverse solutions in decision space than MO_PSO with special crowding distance (SCD) as seen from Fig. 7 in [18].

Motivated by the need to research better niching methods for reproduction of viable candidates to maintain diversity in both objective and decision space as well as the need to improve the performance of EAs in the objective space for MMMOPs, an EA is presented in this work. The proposed framework is called Differential Evolution for Multi-Modal Multi-objective problems (DE-TriM). Along with a novel strategy to select the mating candidates, this work also proposes a novel scheme to create a feedback on size of the mating pool. These features make the proposed framework adaptive to problem characteristics. Experiments on the test functions from [18] and comparison with algorithms for MMMOPs establishes the efficacy of the proposed work.

Rest of the paper is organized as follows. Section 2 outlines the proposed framework of DE-TriM. Section 3 discusses the experiments performed to establish its efficacy. Section 4 concludes the article with future scopes to extend this work.

2 PROPOSED APPROACH: DESCRIPTION OF DE-TRIM

The proposed framework is outlined in Algorithm 1. It takes the problem description, population size (NP), maximum number of fitness evaluations ($MaxFES$), and the set of reference vectors (R) as inputs. It generates PSs and PF as outputs. The location of R depends upon decision-maker's preference. If such preference is unavailable, Das and Dennis's approach [4] of reference vector generation is followed. This continues to be the standard strategy for reference vector generation in various works [6, 10, 15, 19].

Algorithm 1 has the following major sections:

- *Initialization*: Parameter initialization as well as population initialization occurs from line 2 to 6.
- *Parameter Update*: At the launch of G^{th} generation, the mean of F (scale factor) and CR (crossover rate) is updated at line 8. These are the parameters of Differential Evolution [5, 16].

Algorithm 1 General Framework of DE-TriM

Input: *prob*: A multi-modal multi-objective problem having n-dimensional decision space (lower-bounded by X^L and upper-bounded by X^U) and M-dimensional objective space; NP: Population size; $MaxFES$: Maximal of fitness evaluations; R: n_{dir} number of M-dimensional reference vectors

Output: PS: Pareto-optimal sets; PF: Pareto-front

1: **procedure** DETRIM($prob$, NP, $MaxFES$, R)
2: $G_{max} = MaxFES/n_{dir}$ (Calculate maximum generations)
3: $PS_{G=1}$: Randomly initialize a $NP \times n$ matrix bounded by X^L and X^U
4: $PF_{G=1} = F(PS_{G=1})$ (Evaluate fitness according to *prob*)
5: Initialize F and CR for all candidates in PS
6: $SP_{G=1}$: Initialize a vector of length n_{dir} having same sub-population sizes ($= NP/n_{dir}$) for each direction
7: **for** $G = 1$ to G_{max} (until termination) **do**
8: Obtain mean values, F_m and CR_m, over F and CR of all candidates in PS_G
9: **for** $dir = 1$ to n_{dir} (for each direction) **do**
10: Create a sub-population with $SP_{dir,G}$ candidates from PS_G which are closest to $W_{dir} \in R$ in terms of $d2$ (Eq. (3))
11: Cur: Assign any random candidate of the sub-population as the current candidate
12: Sample $F \leftarrow N(F_m, 0.1)$ and $CR \leftarrow N(CR_m, 0.1)$ for Cur such that $F \in (0, 1]$ and $CR \in (0, 1]$
13: New: Obtain a new candidate according to DE/rand/1/bin (Eq. (4) and (5)) using Cur as X_i and the mating pool created in step 10
14: PS_G: Append New to PS_G
15: PF_G: Append $F(New)$ to PF_G
16: **end for**
17: PS_{G+1} and PF_{G+1}: Create population of size NP for next generation using non-dominated sorting and SCD (as secondary criteria) on PS_G and PF_G
18: AD: Find indices of the directions in R to which each candidate of PF_{G+1} is closest to in terms of $d2$ (Eq. (3))
19: $SP_{G+1} = Feedback\_Allocation(AD, NP, n_{dir})$ (Algorithm 2)
20: **if** G is divisible by 10 **then**
21: Re-assign all F and CR to initial values
22: **end if**
23: **end for**
24: Return $PS = PS_{G_{max}}$ and $PF = PF_{G_{max}}$
25: **end procedure**

- *Mating pool creation*: For dir^{th} direction, at the G^{th} generation, a mating pool of size $SP_{dir,G}$ is created in line 10. The mating pool consists of candidates from PS_G closest (in terms of Eq. (3)) to the dir^{th} reference vector (W_{dir}) from R.

$$
\begin{aligned}
d2\left(X_i, W_j\right) &= \left\| F\left(X_i\right) - \left(f^{min} + d1\left(X_i, W_j\right) \frac{W_j}{\left\|W_j\right\|} \right) \right\| \\
\text{where, } d1\left(X_i, W_j\right) &= \frac{\left\| \left(F\left(X_i\right) - f^{min}\right)^T W_j \right\|}{\left\|W_j\right\|}
\end{aligned}
\tag{3}
$$

Algorithm 2 Feedback for resource allocation to determine sub-population sizes

Input: *AD*: A vector of length *NP* representing indices (j) of directions ($W_j \in R$) to which $X_i \in PS$ is closest to in terms of $d2$ (Eq. (3)); *NP*: Population size; n_{dir}: Number of reference vectors

Output: *SP*: A vector of length n_{dir} with updated sub-population sizes

procedure FEEDBACK_ALLOCATION(AD, NP, n_{dir})
  for $dir = 1$ to n_{dir} (for each direction) **do**
    NP_{dir}: Calculate the number of indices in AD that are equal to dir
    $share = (NP_{dir}/NP) \times 100$ (Calculate the share of population closest to dir)
    $SP_{dir} = \frac{100-share}{n_{dir}-1} \times \frac{NP}{100}$ (Assign larger sub-population sizes to the directions having smaller shares and vice-versa)
  end for
  Return SP
end procedure

- *Reproduction (DE/rand/1/bin)*: Reproduction and related operations occur between line 11 to 15. At first, a random candidate from the mating pool (created at line 10) is considered as the parent candidate Cur. Next, from Gaussian distribution with F_m, CR_m and a standard deviations of 0.1, F and CR are independently sampled. Using F and three random candidates (X_{r_1}, X_{r_2} and X_{r_3}) sampled from the mating pool (created at line 10) and mutation with 1 difference vector (Eq. (4)) is performed to generate the donor vector, V_i. Using CR, V_i and $X_i = Cur$, binomial crossover (Eq. (5)) is performed to generate trial vector $U_i = New$. During crossover, a random decision variable (I_{rand}) is always borrowed from the donor vector such that $New \neq Cur$. The new candidate is added to the population.

$$V_{i,G+1} = X_{r_1,G} + F \times \left(X_{r_2,G} - X_{r_3,G}\right) \tag{4}$$

$$u_{ij,G+1} = \begin{cases} v_{ij,G+1}, \text{ if } rand(1) \leq CR \text{ or } j = I_{rand} \\ x_{ij,G+1}, \text{ otherwise} \end{cases} \tag{5}$$

- *Environmental selection*: At the end of the G^{th} generation, n_{dir} new candidates have been added to the population. To keep the population size constant (equal to NP), non-dominated sorting with Special Crowding Distance (SCD) [18] is performed in line 17. SCD considers crowding distance in decision space as well as in objective space. For further details of SCD, readers are referred to [18].
- *Feedback in terms of sub-population size*: This occurs between line 18 to 19 which calls Algorithm 2. The idea is to assign larger sub-population size to the directions having lesser number of associated candidates. As seen from Fig. 1, if a region (e.g. C) is empty or much less dense as compared to other regions (e.g. A), the mating pool will borrow candidates from neighboring regions. Increasing the size of this mating pool helps to increase the chances of generation of a candidate in the empty region. Such a scenario is very common for problems with imbalance difficulties or with biased density of solutions [13, 15].

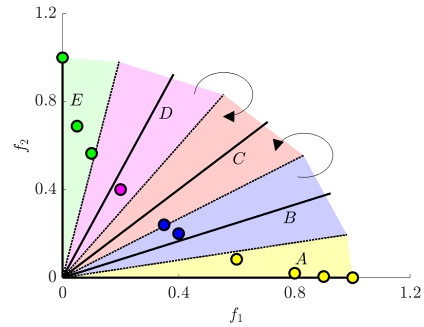
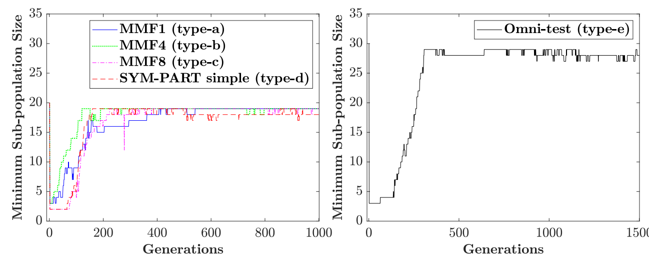


Figure 1: Candidate selection dictating the mating pool formation with respect to the objective space which is divided by the reference lines into 5 regions associated with each line (A, B, C, D, E)



(a) For 2D decision space **(b) For 3D decision space**

Figure 2: Variation of minimum sub-population size with generations for 5 types of PFs appearing over 11 MMMOPs.

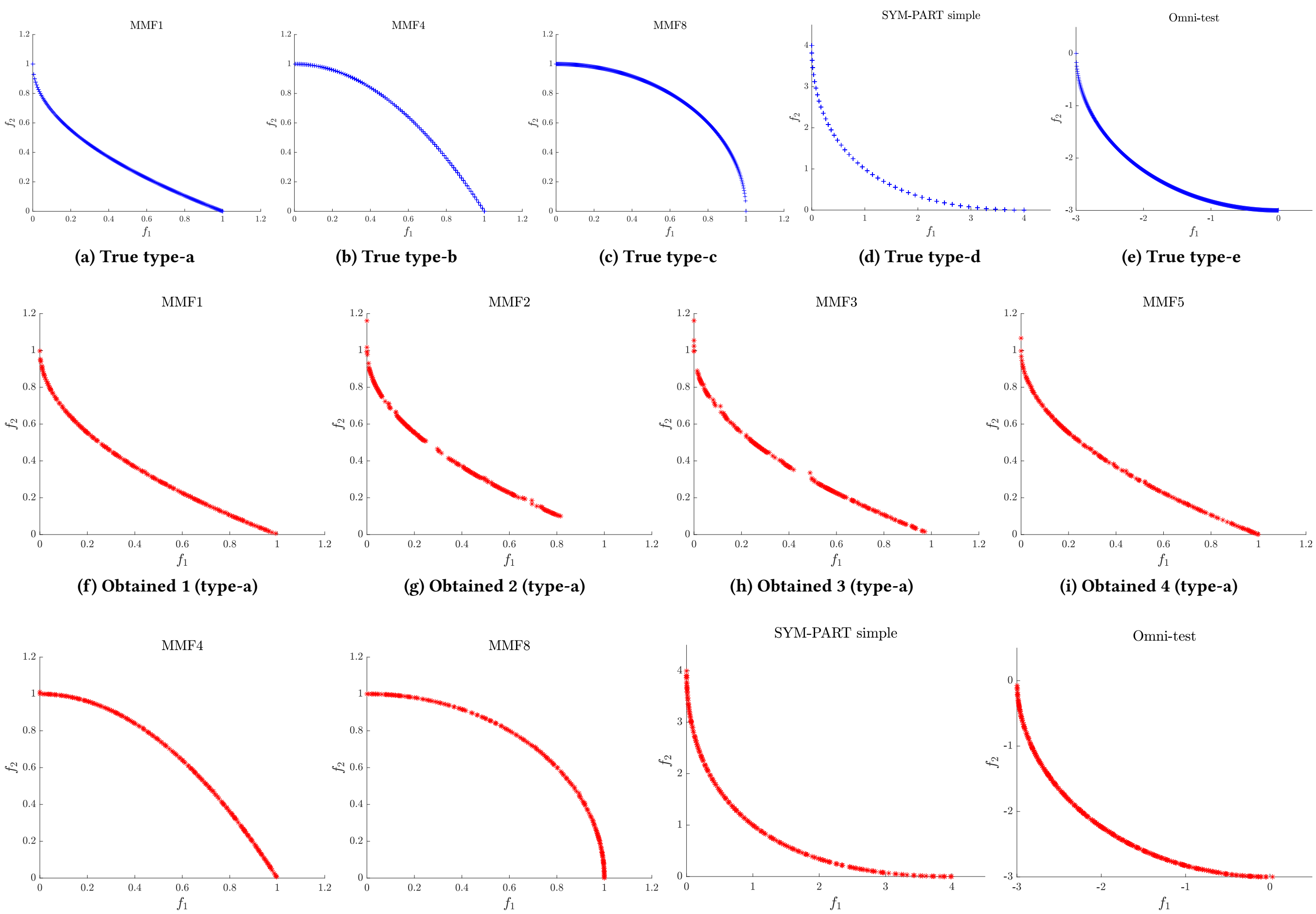
- *Refresh parameters*: At the end of a small number of generations, DE-TriM forgets the learned parameters and F and CR are re-initialized in line 20 to 22 like in [14].

2.1 Algorithmic differences of DE-TriM with other decomposition based MOEAs

Though the reference vector generation as well as association rule of DE-TriM are similar to other decomposition based MOEAs [6, 10, 19], it stands out at regulating the number of candidates (Algorithm 2) for forming the sub-population (mating pool) for a given direction. For the example in Fig. 1, in initial phase, the number of candidates in the mating pool of each direction is two. Thus, the two candidates associated with direction A but farthest from it, will neither participate in the mating pool for direction A nor for its neighboring direction B. Moreover, unlike MOEA/D [19] or MOEA/DD [10] which uses scalarization of objective vectors, DE-TriM uses SCD in its environmental selection step, which helps it to deal with MMMOPs. The third feature of DE-TriM, which sets it apart, is the adaptive parameters for its reproduction operators.

3 PERFORMANCE ANALYSIS

To analyze the performance of the proposed framework, DE-TriM is implemented in MATLAB R2018a using a 64-bit computer with 8GB



(a) True type-a (b) True type-b (c) True type-c (d) True type-d (e) True type-e

(f) Obtained 1 (type-a) (g) Obtained 2 (type-a) (h) Obtained 3 (type-a) (i) Obtained 4 (type-a)

(j) Obtained 5 (type-b) (k) Obtained 6 (type-c) (l) Obtained 7 (type-d) (m) Obtained 8 (type-e)

Figure 3: PFs obtained through DE-TriM as compared to true PFs. There are five types of PFs (type-a: MMF1, MMF2, MMF3, MMF5, MMF6 and MMF7; type-b: MMF4; type-c: MMF8; type-d: SYM-PART simple and SYM-PART rotated; and type-e: Omni-test). Out of the six type-a MMMOPs, converged PFs are obtained for MMF1, MMF6 and MMF7 as shown in Obtained 1. For the other three type-a MMMOPs, there are gaps or outliers in the obtained PFs. For both the problems of type-d category, identical PFs are obtained as shown in Obtained 7. Rest of the category types have one problem each and their corresponding PFs as obtained by DE-TriM are shown as Obtained 5, Obtained 6 and Obtained 8, respectively.

RAM having Intel Core i7 processor @ 2.20GHz. The specifications of the benchmark test functions, performance measures, and the parameter settings of the algorithms are provided in this section. An experiment is also presented to provide the effectiveness of the feedback module of DE-TriM.

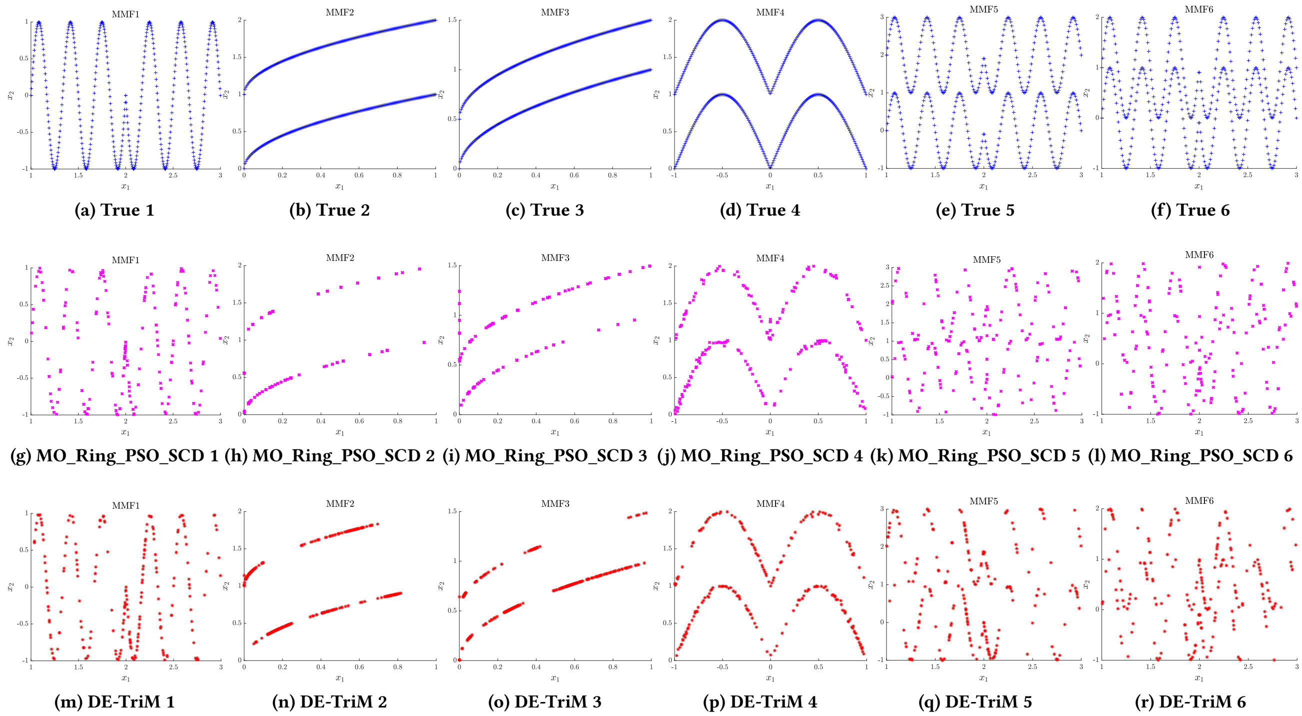
3.1 Test Functions

This work considers 11 test problems [18] viz. MMF1, MMF2, MMF3, MMF4, MMF5, MMF6, MMF7, MMF8, SYM-PART simple, SYM-PART rotated and the Omni-test problem. For each problem, $MaxFES$ and NP are considered as $5000n$ and $100n$, respectively. The characteristics of these MMMOPs are enlisted in Table 1. Unlike the other problems, #PSs in Omni-test problem are scalable. Hence, an Omni-test problem with $n = 3$ has 27 equivalent PSs.

3.2 Performance Measures

Performance of traditional MOEAs are noted in terms of their abilities to reach the true PF (convergence) and to make the population widely spread and uniformly distributed across the PF (diversity) [3]. Most common measures to record these properties are the inverted generational distance (IGD) [3, 13] and the hypervolume indicator (HV) [1, 15]. For HV evaluation, a reference point (F_{ref}) is selected for forming the hyper-rectangle. For IGD evaluation, N_{IGD} number of points are sampled from the true PF. The choice[1] of F_{ref} and N_{IGD} for each problem is stated in Table 1.

[1]The reference data for PSs and PF, values of F_{ref} and the MATLAB implementations of MO_Ring_PSO_SCD and DN-NSGA-II are obtained from http://www5.zzu.edu.cn/ecilab/info/1036/1163.htm.



(a) True 1 (b) True 2 (c) True 3 (d) True 4 (e) True 5 (f) True 6

(g) MO_Ring_PSO_SCD 1 (h) MO_Ring_PSO_SCD 2 (i) MO_Ring_PSO_SCD 3 (j) MO_Ring_PSO_SCD 4 (k) MO_Ring_PSO_SCD 5 (l) MO_Ring_PSO_SCD 6

(m) DE-TriM 1 (n) DE-TriM 2 (o) DE-TriM 3 (p) DE-TriM 4 (q) DE-TriM 5 (r) DE-TriM 6

Figure 4: PSs obtained through DE-TriM and MO_Ring_PSO_SCD as compared to true PSs for MMF1, MMF2, MMF3, MMF4, MMF5 and MMF6 (labelled as 1, 2, 3, 4, 5 and 6, respectively) problems.

Table 1: Problem specifications: dimension of decision space (n), dimension of objective space (M), lower bounds of decision variables (X^L), upper bounds of decision variables (X^U), reference point for hypervolume calculation (F_{ref}), number of points in the reference sets for IGDF and IGDX evaluations (N_{IGD}) and number of Pareto-optimal sets (#PSs).

Problems	n	M	X^L	X^U	F_{ref}	N_{IGD}	#PSs
MMF1	2	2	$[1, -1]$	$[3, 1]$	$[1.1, 1.1]$	400	2
MMF2	2	2	$[0, 0]$	$[1, 1]$	$[1.1, 1.1]$	400	2
MMF3	2	2	$[0, 0]$	$[1, 1.5]$	$[1.1, 1.1]$	400	2
MMF4	2	2	$[-1, 0]$	$[1, 2]$	$[1.1, 1.1]$	400	4
MMF5	2	2	$[1, -1]$	$[3, 3]$	$[1.1, 1.1]$	400	4
MMF6	2	2	$[1, -1]$	$[3, 2]$	$[1.1, 1.1]$	400	4
MMF7	2	2	$[1, -1]$	$[3, 1]$	$[1.1, 1.1]$	400	2
MMF8	2	2	$[-\pi, 0]$	$[\pi, 9]$	$[1.1, 1.1]$	400	4
SYM-PART simple	2	2	$[-20, -20]$	$[20, 20]$	$[4.4, 4.4]$	396	9
SYM-PART rotated	2	2	$[-20, -20]$	$[20, 20]$	$[4.4, 4.4]$	396	9
Omni-test	3	2	$[0, 0, 0]$	$[6, 6, 6]$	$[4.4, 4.4]$	600	27

Although until recently MMMOPs have not received much attention, yet about a decade ago IGD in decision space has been considered as a performance indicator in [20]. Also, Pareto Sets Proximity (PSP) [18] is recently being considered for assessing the performance of an EA in the decision space.

In this work, for distinction, IGD in decision space is mentioned as IGDX and IGD in objective space is mentioned as IGDF. Also, in order to have lower value as the better measure over all the measures, the reciprocal of PSP (mentioned as rPSP) and HV (mentioned as rHV) are considered.

3.3 Analysis of Allocation with Feedback

This experiment aims to show the utility of the feedback module (Algorithm 2). With its help, all sub-regions (created by the reference vectors based decomposition of the objective space), have almost equal number of candidates associated to it. If all the sub-regions have equal number of candidates, the minimum number of candidates in a sub-region will be equal to $NP/n_{dir} = 100n/10 = 10n$ which is 20 when $n = 2$ and 30 when $n = 3$. As can be seen from Figs. 2a and 2b, the minimum sub-population size saturates near 20 and 30, respectively. Thus, this experiment shows that for all types of Pareto-front shapes over the 11 benchmark MMMOPs, the feedback step works efficiently.

3.4 Performance of DE-TriM

For the performance in the decision space (in terms of rPSP and IGDX), DE-TriM is compared with other EAs in Table 2. For the performance in the objective space (in terms of rHV and IGDF), DE-TriM is compared with other EAs in Table 3. For all results, p-values are noted as obtained from two-tailed paired t-test [13] under the null hypothesis (H_0) that DE-TriM is equivalent to other EAs at 95% confidence interval. Superiority of DE-TriM ($p \leq 0.05$, H_0 rejected)

Table 2: Mean and standard deviation of rPSP and IGDX for MMMOPs over 31 independent runs[3]

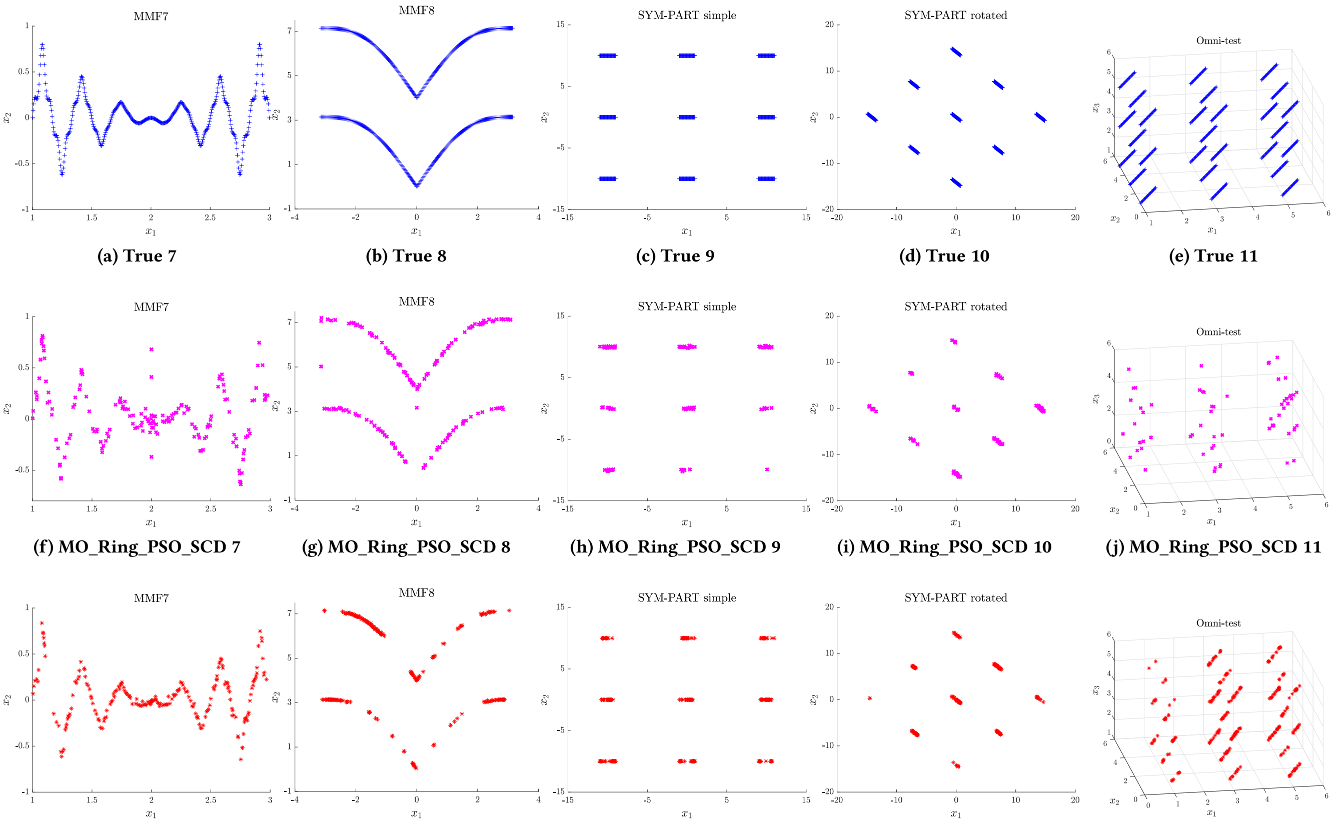
	rPSP				IGDX			
Problems	DE-TriM	MO_Ring_PSO_SCD	DN-NSGA-II	NAEMO	DE-TriM	MO_Ring_PSO_SCD	DN-NSGA-II	NAEMO
MMF1	**0.0468** ± **0.0062**	0.0495 ± 0.0017 (∼)	0.0992 ± 0.0126 (+)	0.3498 ± 0.1101 (+)	**0.0467** ± **0.0062**	0.0488 ± 0.0016 (∼)	0.0903 ± 0.0108 (+)	0.2291 ± 0.0344 (+)
MMF2	0.0586 ± 0.0174	**0.0455** ± **0.0086** (−)	0.1486 ± 0.0470 (+)	0.0632 ± 0.0294 (∼)	0.0517 ± 0.0053	**0.0397** ± **0.0099** (−)	0.1135 ± 0.0530 (+)	0.0633 ± 0.0189 (+)
MMF3	**0.0276** ± **0.0072**	0.0309 ± 0.0065 (∼)	0.1210 ± 0.0929 (+)	0.0546 ± 0.0050 (+)	**0.0239** ± **0.0046**	0.0267 ± 0.0056 (∼)	0.1150 ± 0.0593 (+)	0.0514 ± 0.0336 (+)
MMF4	**0.0238** ± **0.0042**	0.0271 ± 0.0014 (+)	0.0846 ± 0.0287 (+)	0.0889 ± 0.0348 (+)	**0.0211** ± **0.0073**	0.0263 ± 0.0015 (+)	0.0828 ± 0.0222 (+)	0.0886 ± 0.0293 (+)
MMF5	0.0886 ± 0.0218	**0.0861** ± **0.0044** (−)	0.1804 ± 0.0167 (+)	0.3065 ± 0.0511 (+)	0.0895 ± 0.0133	**0.0866** ± **0.0049** (∼)	0.1141 ± 0.0144 (+)	0.2473 ± 0.0367 (+)
MMF6	0.0772 ± 0.0169	**0.0727** ± **0.0037** (∼)	0.1499 ± 0.0161 (+)	0.4906 ± 0.1002 (+)	0.0764 ± 0.0136	**0.0732** ± **0.0048** (∼)	0.1387 ± 0.0183 (+)	0.3257 ± 0.0471 (+)
MMF7	**0.0188** ± **0.0056**	0.0266 ± 0.0015 (+)	0.0531 ± 0.0112 (+)	0.1341 ± 0.0668 (+)	**0.0197** ± **0.0049**	0.0264 ± 0.0019 (+)	0.0558 ± 0.0074 (+)	0.1235 ± 0.0164 (+)
MMF8	0.1049 ± 0.0154	**0.0677** ± **0.0048** (−)	0.3307 ± 0.0965 (+)	5.7662 ± 0.2983 (+)	0.0988 ± 0.0029	**0.0674** ± **0.0055** (−)	0.2957 ± 0.1106 (+)	1.9494 ± 0.5552 (+)
SYM-PART simple	**0.0737** ± **0.0035**	0.1776 ± 0.0402 (+)	4.1401 ± 1.1040 (+)	1.2148 ± 0.1614 (+)	**0.0740** ± **0.0034**	0.1817 ± 0.0313 (+)	4.0551 ± 0.7562 (+)	1.2065 ± 0.7072 (+)
SYM-PART rotated	**0.1639** ± **0.0817**	0.3536 ± 0.0753 (+)	7.6152 ± 3.8126 (+)	6.3824 ± 2.6238 (+)	**0.1890** ± **0.0762**	0.3228 ± 0.0249 (+)	3.8414 ± 0.0450 (+)	4.4007 ± 1.5689 (+)
Omni-test	**0.0762** ± **0.0352**	0.3955 ± 0.0540 (+)	1.3811 ± 0.2010 (+)	0.7222 ± 0.3339 (+)	**0.0735** ± **0.0439**	0.4019 ± 0.0975 (+)	1.3734 ± 0.2426 (+)	0.7181 ± 0.2284 (+)
Sum-up	+/−/∼	5/3/3	11/0/0	10/0/1	+/−/∼	5/2/4	11/0/0	11/0/0

[3] An odd number of runs such that the median run is unique for comparison of results by visualization.

Table 3: Mean and standard deviation of rHV and IGDF for MMMOPs over 31 independent runs[3]

	rHV				IGDF			
Problems	DE-TriM	MO_Ring_PSO_SCD	DN-NSGA-II	NAEMO	DE-TriM	MO_Ring_PSO_SCD	DN-NSGA-II	NAEMO
MMF1	**1.1456** ± **0.0007**	1.1483 ± 0.0005 (+)	1.1493 ± 0.0017 (+)	1.3408 ± 0.2700 (+)	**0.0026** ± **0.0003**	0.0037 ± 0.0002 (+)	0.0044 ± 0.0005 (+)	0.0828 ± 0.0080 (+)
MMF2	**1.1525** ± **0.0468**	1.1819 ± 0.0059 (+)	1.1968 ± 0.0397 (+)	1.2141 ± 0.0313 (+)	**0.0037** ± **0.0002**	0.0215 ± 0.0030 (+)	0.0357 ± 0.0025 (+)	0.0259 ± 0.0094 (+)
MMF3	**1.1560** ± **0.0150**	1.1730 ± 0.0036 (+)	1.1890 ± 0.0525 (+)	1.2487 ± 0.1241 (+)	**0.0043** ± **0.0004**	0.0150 ± 0.0025 (+)	0.0331 ± 0.0087 (+)	0.0220 ± 0.0101 (+)
MMF4	**1.8521** ± **0.0044**	1.8616 ± 0.0021 (+)	1.8580 ± 0.0011 (+)	1.9722 ± 0.0978 (+)	**0.0023** ± **0.0002**	0.0036 ± 0.0004 (+)	0.0032 ± 0.0002 (+)	0.0278 ± 0.0054 (+)
MMF5	**1.1463** ± **0.0007**	1.1484 ± 0.0005 (+)	1.1495 ± 0.0015 (+)	1.2672 ± 0.1695 (+)	**0.0028** ± **0.0003**	0.0037 ± 0.0002 (+)	0.0040 ± 0.0008 (+)	0.0858 ± 0.0092 (+)
MMF6	**1.1456** ± **0.0006**	1.1483 ± 0.0009 (+)	1.1488 ± 0.0018 (+)	1.3450 ± 0.0532 (+)	**0.0025** ± **0.0001**	0.0034 ± 0.0003 (+)	0.0036 ± 0.0003 (+)	0.1825 ± 0.0094 (+)
MMF7	**1.1453** ± **0.0003**	1.1484 ± 0.0006 (+)	1.1499 ± 0.0011 (+)	1.2097 ± 0.0088 (+)	**0.0024** ± **0.0001**	0.0038 ± 0.0003 (+)	0.0041 ± 0.0004 (+)	0.0357 ± 0.0037 (+)
MMF8	**2.3739** ± **0.0069**	2.4111 ± 0.0169 (+)	2.3812 ± 0.0019 (+)	3.9678 ± 1.4813 (+)	**0.0028** ± **0.0002**	0.0048 ± 0.0002 (+)	0.0040 ± 0.0004 (+)	0.1174 ± 0.0440 (+)
SYM-PART simple	**0.0600** ± **0.0000**	0.0605 ± 0.0001 (+)	0.0601 ± 0.0000 (+)	0.0625 ± 0.0030 (+)	**0.0099** ± **0.0011**	0.0435 ± 0.0038 (+)	0.0129 ± 0.0016 (+)	0.1088 ± 0.0297 (+)
SYM-PART rotated	**0.0601** ± **0.0000**	0.0606 ± 0.0001 (+)	**0.0601** ± **0.0000** (∼)	0.0615 ± 0.0010 (+)	**0.0120** ± **0.0023**	0.0456 ± 0.0068 (+)	0.0155 ± 0.0023 (+)	0.1049 ± 0.0176 (+)
Omni-test	**0.0189** ± **0.0000**	0.0190 ± 0.0000 (+)	**0.0189** ± **0.0000** (∼)	0.0190 ± 0.0001 (+)	**0.0061** ± **0.0036**	0.0415 ± 0.0038 (+)	0.0079 ± 0.0006 (+)	0.0412 ± 0.0157 (+)
Sum-up	+/−/∼	11/0/0	9/0/2	11/0/0	+/−/∼	9/0/2	8/0/3	11/0/0

[3] An odd number of runs such that the median run is unique for comparison of results by visualization.



(a) True 7 (b) True 8 (c) True 9 (d) True 10 (e) True 11

(f) MO_Ring_PSO_SCD 7 (g) MO_Ring_PSO_SCD 8 (h) MO_Ring_PSO_SCD 9 (i) MO_Ring_PSO_SCD 10 (j) MO_Ring_PSO_SCD 11

(k) DE-TriM 7 (l) DE-TriM 8 (m) DE-TriM 9 (n) DE-TriM 10 (o) DE-TriM 11

Figure 5: PSs obtained through DE-TriM and MO_Ring_PSO_SCD as compared to true PSs for MMF7, MMF8, SYM-PART simple, SYM-PART rotated and Omni-test (labelled as 7, 8, 9, 10 and 11, respectively) problems.

is marked by +, equivalence ($p > 0.05$, H_0 not rejected) by ∼ and inferiority ($p \leq 0.05$, H_0 rejected) by −. The obtained PFs (Fig. 3) and PSs (Figs. 4 and 5) of the median run are also visualized for further analysis of the results. With *MaxFES* and *NP* as mentioned before, the other parameter specifications of the competitor EAs are as follows:

(1) MO_Ring_PSO_SCD: This work[1] is setup as per the specifications in [18]. It is important as it introduced the concept of SCD.
(2) DN-NSGA-II: This work[1] is setup as per the specifications in [11]. It is important as it established the importance of niching for MMMOPs.
(3) NAEMO: This work is setup as per the specifications in [15]. However, in [15], $L_{hard} = n_{dir}$ was considered. As L_{hard} of NAEMO dictates its final population size, for this experiment, $n_{dir} = 10$ and $L_{hard} = NP$ are considered. NAEMO is important as it laid the theoretical foundation for the concept of neighboring zones in the objective space is mapped from neighboring zones in the decision space.
(4) DE-TriM: This work[2] initializes $n_{dir} = 10$, $F = 0.5$, $CR = 0.2$ and learning period for F and CR as 10 generations. However, F and CR in DE-TriM are made adaptive to local problem characteristics.

[2]MATLAB source code for DE-TriM is attached in the supplementary material.

3.4.1 Performance in the decision space. From Table 2, DE-TriM is found to have best rPSP and IGDX values for 7 out of 11 problems. For the remaining 4 problems, MO_Ring_PSO_SCD outperforms DE-TriM. Moreover, neither DN-NSGA-II nor NAEMO outperforms either of DE-TriM or MO_Ring_PSO_SCD for any of the problems. This is because DN-NSGA-II ignores diversity in objective space whereas NAEMO ignores the diversity in the decision space. For further comparison of DE-TriM and MO_Ring_PSO_SCD, Figs. 4 and 5 are considered and the following observations are made:

(1) Comparing Figs. 4h with 4n, Figs. 4k with 4q, Figs. 4l with 4r, and Figs. 5g with 5l, indeed MO_PSO_Ring_SCD have generated more uniform distribution of solutions in decision space than DE-TriM for MMF2, MMF5, MMF6 and MMF8 problems.
(2) For both the SYM-PART problems, MO_Ring_PSO_SCD (Figs. 5h and 5i) and DE-TriM (Figs. 5m and 5n) have identified all the 9 Pareto-optimal sets. However, unlike DE-TriM, for MO_Ring_PSO_SCD, more candidates have converged to some PSs than others. This hampers the rPSP and IGDX values in Table 2.

(3) For MMF4 problem, the distribution of solutions in decision space is equally good for MO_Ring_PSO_SCD (Fig. 4j) and DE-TriM (Fig. 4p).
(4) For MMF1 and MMF3 problems, the distribution of solutions in decision space is equally poor for MO_Ring_PSO_SCD (Figs. 4g and 4i) and DE-TriM (Figs. 4m and 4o). For both the problems, one PS has higher distribution of solution than the other.
(5) For MMF7 and Omni-test problems, DE-TriM (Figs. 5k and 5o) yield better approximations of PSs than MO_Ring_PSO_SCD (Figs. 5f and 5j). There are several outliers in the MMF7 solution obtained by MO_Ring_PSO_SCD (Fig. 5f). For Omni-test problem, some of the solutions have not been discovered by MO_Ring_PSO_SCD (Fig. 5j).

3.4.2 Performance in the objective space. From Table 3, DE-TriM is found to be a clear winner. This is an important result as for most of the algorithms which address MMMOPs, the performance in objective space gets deteriorated [12, 18]. It can be seen that although rHV and IGDF values of DE-TriM are lowest among the competitor algorithms for MMF2, MMF3 and MMF5, their PFs (Figs. 3g, 3h and 3i, respectively) have holes or outliers. A 2-objective problem, being the simplest MMMOP, still shows scope of improvement in the performance in objective space.

Thus, overall, DE-TriM has similar performance like the state-of-the-art algorithm (MO_Ring_PSO_SCD) for addressing MMMOPs in decision space, but much better performance in objective space which establishes the superiority of the proposed approach (DE-TriM).

4 CONCLUSION

This study demonstrates a Differential Evolution approach to address Multi-Modal Multi-objective problems (DE-TriM). It proposes a novel mating pool selection strategy and a novel resource allocation strategy based on reference vector based decomposition of the objective space. These two strategies, accompanied by non-dominated sorting with special crowding distance [18], helps in efficiently addressing multi-modal multi-objective problems.

In future, more studies are to be undertaken in order to further improve the diversity in objective and decision space. As with increase in dimension of objective space, dominance resistance is observed [13], novel methods will be required to overcome this issue for multi-modal many-objective problems. An interesting observation is that the shape of all the Pareto-optimal sets for a given problem are identical, hence, some correlation based reproduction operator could be designed to generate more identical solutions over all the Pareto-optimal sets. Such works are open for research.

ACKNOWLEDGMENTS

This work is partially supported by Indian Statistical Institute and by J. C. Bose Fellowship (SB/SJ/JCB-033/2016) of Department of Science and Technology, Government of India.

REFERENCES

[1] Johannes Bader and Eckart Zitzler. 2011. HypE: An algorithm for fast hypervolume-based many-objective optimization. *Evolutionary computation* 19, 1 (2011), 45–76.
[2] K. P. Chan and T. Ray. 2005. An evolutionary algorithm to maintain diversity in the parametric and the objective space. In *International Conference on Computational Robotics and Autonomous Systems (CIRAS), Centre for Intelligent Control, National University of Singapore.*
[3] Carlos A. Coello Coello. 2017. Recent Results and Open Problems in Evolutionary Multiobjective Optimization. In *Theory and Practice of Natural Computing*, Carlos Martín-Vide, Roman Neruda, and Miguel A. Vega-Rodríguez (Eds.). Springer International Publishing, Cham, 3–21.
[4] Indraneel Das and John E Dennis. 1998. Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. *SIAM Journal on Optimization* 8, 3 (1998), 631–657.
[5] Sambarta Dasgupta, Swagatam Das, Arijit Biswas, and Ajith Abraham. 2009. On stability and convergence of the population-dynamics in differential evolution. *AI Communications* 22, 1 (2009), 1–20.
[6] Kalyanmoy Deb and Himanshu Jain. 2014. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints. *IEEE Transactions on Evolutionary Computation* 18, 4 (2014), 577–601.
[7] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation* 6, 2 (2002), 182–197.
[8] Kalyanmoy Deb and Santosh Tiwari. 2005. Omni-optimizer: A Procedure for Single and Multi-objective Optimization. In *Evolutionary Multi-Criterion Optimization*, Carlos A. Coello Coello, Arturo Hernández Aguirre, and Eckart Zitzler (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 47–61.
[9] Hisao Ishibuchi, Noritaka Tsukamoto, and Yusuke Nojima. 2008. Evolutionary many-objective optimization: A short review. In *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*. IEEE, 2419–2426.
[10] Ke Li, Kalyanmoy Deb, Qingfu Zhang, and Sam Kwong. 2015. An evolutionary many-objective optimization algorithm based on dominance and decomposition. *IEEE Transactions on Evolutionary Computation* 19, 5 (2015), 694–716.
[11] J. J. Liang, C. T. Yue, and B. Y. Qu. 2016. Multimodal multi-objective optimization: A preliminary study. In *2016 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2454–2461.
[12] Yiping Liu, Gary G. Yen, and Dunwei Gong. 2018. A multi-modal multi-objective evolutionary algorithm using two-archive and recombination strategies. *IEEE Transactions on Evolutionary Computation* (2018).
[13] Monalisa Pal and Sanghamitra Bandyopadhyay. 2019. ESOEA: Ensemble of single objective evolutionary algorithms for many-objective optimization. *Swarm and Evolutionary Computation* (2019). https://doi.org/10.1016/j.swevo.2019.03.006
[14] A Kai Qin and Ponnuthurai N Suganthan. 2005. Self-adaptive differential evolution algorithm for numerical optimization. In *2005 IEEE congress on evolutionary computation*, Vol. 2. IEEE, 1785–1791.
[15] Raunak Sengupta, Monalisa Pal, Sriparna Saha, and Sanghamitra Bandyopadhyay. 2019. NAEMO: Neighborhood-sensitive archived evolutionary many-objective optimization algorithm. *Swarm and Evolutionary Computation* 46 (2019), 201 – 218. https://doi.org/10.1016/j.swevo.2018.12.002
[16] Rainer Storn and Kenneth Price. 1997. Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *Journal of Global Optimization* 11, 4 (01 Dec 1997), 341–359. https://doi.org/10.1023/A:1008202821328
[17] Ryoji Tanabe and Hisao Ishibuchi. 2018. A Decomposition-Based Evolutionary Algorithm for Multi-modal Multi-objective Optimization. In *Parallel Problem Solving from Nature – PPSN XV*, Anne Auger, Carlos M. Fonseca, Nuno Lourenço, Penousal Machado, Luís Paquete, and Darrell Whitley (Eds.). Springer International Publishing, Cham, 249–261.
[18] Caitong Yue, Boyang Qu, and Jing Liang. 2018. A multiobjective particle swarm optimizer using ring topology for solving multimodal multiobjective problems. *IEEE Transactions on Evolutionary Computation* 22, 5 (2018), 805–817.
[19] Qingfu Zhang and Hui Li. 2007. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation* 11, 6 (2007), 712–731.
[20] Aimin Zhou, Qingfu Zhang, and Yaochu Jin. 2009. Approximating the set of Pareto-optimal solutions in both the decision and objective spaces by an estimation of distribution algorithm. *IEEE transactions on evolutionary computation* 13, 5 (2009), 1167–1189.