DPb-MOPSO: A Dynamic Pareto bi-level Multi-Objective Particle Swarm Optimization Algorithm

Ahlem Aboud [*, a, b], Nizar Rokbani [b, c], Raja Fdhila [b], Abdulrahman M. Qahtani [d], Omar Almutiry [e], Habib Dhahri [e], Amir Hussain [f], and Adel M. Alimi [b, g]

[a] University of Sousse, ISITCom, 4011, Sousse, Tunisia.
[b] REsearch Groups in Intelligent Machines (REGIM Lab), University of Sfax, National Engineering School of Sfax (ENIS), BP 1173, Sfax, 3038, Tunisia.
[c] High Institute of Applied Science and Technology of Sousse, University of Sousse, Tunisia.
[d] Department of Computer Science, College of Computers and Information Technology, Taif University, P.O.Box. 11099, Taif 21944, Saudi Arabia.
[e] College of Applied Computer Science, King Saud University, Riyadh, Saudi Arabia.
[f] Edinburgh Napier University, School of Computing, Edinburgh EH10 5DT, Scotland, U.K.
[g] Department of Electrical and Electronic Engineering Science, Faculty of Engineering and the Built Environment, University of Johannesburg, South Africa.

Abstract
Particle Swarm Optimization (PSO) system based on the distributed architecture over multiple sub-swarms is very efficient for static multi-objective optimization but has not been considered for solving dynamic multi-objective problems (DMOPs). Tracking the most effective solutions over time and ensuring good exploitation and exploration are the main challenges of solving DMOP. This study proposes a Dynamic Pareto bi-level Multi-Objective Particle Swarm Optimization (DPb-MOPSO) algorithm including two parallel optimization levels. At the first level, all solutions are managed in a single search space. When a dynamic change is successfully detected in the objective values, the Pareto ranking operator is used to enable multiple sub-swarm' subdivisions and processing which drives the second level of enhanced exploitation. A dynamic handling strategy based on random detectors is used to track the changes in the objective function due to time-varying parameters. A response strategy consisting in reevaluating all unimproved solutions and replacing them with newly generated ones is also implemented. The DPb-MOPSO system is tested on DMOPs with different types of time-varying Pareto Optimal Set (POS) and Pareto Optimal Front (POF). Inverted generational distance (IGD), mean inverted generational distance (MIGD), hypervolume difference (HVD), Robust IGD (RIGD) and Robust General Distance (RGD) metrics are used to assess the DPb-MOPSO performance. Quantitative results are analyzed using Friedman's analysis of variance, and the Wilcoxon sum ranks test, while the stability is analyzed using Lyapunov's theorem. The DPb-MOPSO is more robust than several dynamic multi-objective evolutionary algorithms in solving 21 complex problems over a range of changes in both the POS and POF. On IGD and HVD, DPb-MOPSO can solve 8/13 and 8/13 of the 13 UDF and ZJZ functions with moderate changes. DPb-MOPSO can resolve 7/8 FDA and DMOP benchmarks with severe changes to the MIGD, and 6/8 with moderate changes. DPb-MOPSO assumes 7/8, 6/8, and 5/8 for solving FDA, and dMOP functions on IGD and 6/8, 5/8, and 5/8 on HVD metrics considering severe, moderate, and slight environmental changes respectively. Also, it is the winner for solving 8 DMOPs based on RIGD, and RGD metrics.

Keywords: Dynamic Multi-Objective Optimization Problem, Dynamic Particle Swarm Optimization, Detectors, Dynamic Response, Friedman Analysis of Variance, Wilcoxon Test, Lyapunov Theorem.

----

[*] Corresponding author. E-mail address: ahlem.aboud@regim.usf.tn (A. Aboud).
Contributing authors: E-mail addresses: nizar.rokbani@ieee.org (N. Rokbani), raja.fdhila@ieee.org (R. Fdhila), amqahtani@tu.edu.sa (A. M. Qahtani), oalmutiry@ksu.edu.sa (O. Almutiry), hdhahri@ksu.edu.sa (H. Dhahri), a.hussain@napier.ac.uk (A. Hussain), adel.alimi@ieee.org (A. M. Alimi).

1. Introduction

Dynamic Multi-Objective Problems (DMOPs) have two or three conflicting functions characterized by time-varying objectives, parameters, and/or constraints. Farina *et al.*[1] have categorized DMOPs into four types. The main difference has been investigated in the time-varying Pareto Optimal Set (POS) and Pareto Optimal Front (POF). For type I, the POS changes, and the POF remains unchanged. Then, for type II both POS and POF change over time. However, the POF changes and the POS remains constant for DMOPs in type III. Despite all changes in the system, both POS and POF are unchanged for DMOPs in type IV. Multi-Objective Evolutionary Algorithms (MOEAs) and bio-inspired intelligent techniques like the Genetic Algorithm (GA) [2], and the Particle Swarm Optimization (PSO) method [3]– [5] have been investigated to manage the evolutionary stagnation issues for static single/multi-objective optimization. Also, a set of MOEAs has been established as suitable for dynamic multi-objective optimization [6]– [8], and able to find the best compromise of solutions on POS and POF without managing environmental changes.

Different Dynamic Multi-Objective Evolutionary Algorithms (DMOEAs) have been developed to overcome the loss of convergence and diversity issues when solving time-varying problems. Some mechanisms are considered for diversity detection by monitoring a few random individuals to re-evaluate the objectives or constraints violations values [9]. The change detection is usually done frequently by monitoring the evolution of fitness functions using one fixed solution [10], the best-obtained position [11], or the average of all evaluated solutions [12]. The convergence detection methods using one or small set of detectors have the limit to wrongly detecting the converged/ or the non-converged individuals. Also, the efficiency of the detection mechanisms decreases when the size of the affected environmental change decreases, while the need for an important number of detectors. Furthermore, change detection includes additional techniques to react effectively to the change, and the most known are the reinitialization of a certain percentage of the population and the hypermutation which are explicitly used to ensure good convergence and diversity in dynamic search space [13], [14]. Nonetheless, the Multi-Objective Particle Swarm Optimization (MOPSO) approach with distributed architecture is considered for static optimization [15] and has not yet been processed to manage the time-varying POS and POF when solving DMOPs. Besides, many transfer learning-based methods are proposed for DMOPs, which are time-consuming, and increase the diminishing of solutions diversity [16].

In general, the sequential search procedures proposed in the standard MOPSO approach [17] have followed a centralized architecture, where the particles are iteratively updated according to a random global best position (leader) and their historical personal best position. As a consequence, if the personal best and the global best are confused or close to each other's, the MOPSO algorithm may stagnate in local optima and lose diversity over time [18]. In the literature, the most existing DMOEAs designed for DMOPs are developed based on the Evolutionary Algorithm (EA) [19], [20], [21], [22] or Genetic Algorithm (GA) [9]. Furthermore, the use of Multi-Objective Particle Swarm Optimization (MOPSO) over a distributed architecture, consisting of multiple sub-swarms [15], has not yet been investigated to solve DMOPs. Also, it should be noted, that the standard PSO method [3] is not very efficient for solving DMOPs with time-varying POS and POF [18] because of its limited capacities in addressing the dual issue of the outdated memory and the loss of diversity over time.

The state of the art involves comparable DMOEAs to the novel DPb-MOPSO system based essentially on five metrics [14]: Inverted Generational Distance (IGD), Mean Inverted Generational Distance (MIGD), Hypervolume Difference (HVD), Robust IGD (RIGD), and Robust General Distance (RGD) [24], which are chosen referred to the compared state-of-the-art methods in papers [16], [19], and [24]. Different DMOEAs, are compared with the DPb-MOPSO algorithm including: Multi-objective Evolutionary Algorithm Based on Decomposition (MOEA/D) [25], Dynamic Competitive-Cooperative Evolutionary Algorithm (dCOEA) [26], Population Prediction Strategy (PPS) [27], Dynamic Non-dominated Sorting Genetic Algorithm II (DNSGA-II) [9] and Steady-State Generational Evolutionary Algorithm (SGEA) [19]. Also, PSO-based methods are considered namely: Dual multi-objective particle swarm optimization (dMOPSO) [28], distributed Multi-Objective Particles Swarms Optimization, based on the dynamic subdivision of the population using Pareto fronts (pbMOPSO) [15] and Multi-Objective Particles Swarms Optimization (MOPSO) [17], and Dynamic-MOPSO [18]. The transfer learning-based methods aim to modify the main system of PPS [27], and MOEA/D-KF [20] with the baseline MOEA/D algorithm [25]. However, a set of TL methods are detailed in [16] and compared with the DPb-MOPSO namely; MMTL-MOEA/D, RI-MOEA/D, PPS-MOEA/D, KF-MOEA/D, SVR-MOEA/D [21] and Tr-MOEA/D [22]. Also, the proposed DPb-MOPSO algorithm is compared with five prediction-based methods [24] that adapt average, weighted, and adaptively ensemble prediction techniques among five standard multi-objective evolutionary algorithms: SPEA2 [29], MOPSO [17], NSGA-III [30], MOEA/D [25], and MOEA/D-DE [31].

In this paper, we propose a new Dynamic Pareto bi-level Multi-Objective Particle Swarm Optimization (DPb-MOPSO) algorithm. The contribution of this study is detailed as follows:

- ✓ The DPb-MOPSO algorithm is proposed to handle Dynamic Multi-Objective Optimization Problems (DMOPs) with different types of time-varying POS and POF.
- ✓ The novel DPb-MOPSO algorithm integrates a distributed architecture over multiple sub-swarms including a dynamic handling strategy to manage different types of change. The distributed architecture allows higher diversity during the search process.
- ✓ The main difference between the new proposed DPb-MOPSO algorithm and the standard MOPSO algorithm has been investigated in the advantages of using a distributed MOPSO architecture for solving dynamic multi-objective problems and monitoring dynamic change.

The remaining of this paper is organized as follows. Section 2 introduces the overview of dynamic multi-objective optimization methods. Section 3 details the proposed DPb-MOPSO algorithm. Section 4 introduces the preliminary of the experimental study. The analysis and discussion of the results are shown in Section 5. The Lyapunov theorem for stability analysis is detailed in Section 6. Finally, Section 7 summarizes the paper and suggests some prospects for future works.

2. Dynamic Multi-Objective Optimization Methods

Dynamic multi-objective optimization problem f_t is a MOP with M time-varying objective function, variables, and/or constraints [32]. An algorithm G is needed to solve DMOPs in a given period $[t_{began}, t_{end}]$. The mathematical definition of DMOP is expressed in Equation (1).

$$\text{Minimize } F(X,t) = (f_1(X,t), f_2(X,t), \dots, f_M(X,t)) \quad \textbf{(1)}$$

$$\text{Subject to: } g_i(X,t) \leq 0 \text{ and } h_j(X,\text{t}) = 0$$

$$\forall\, i = 1, \dots, n_g(t) \text{ and } j = 1, \dots, n_h(t)$$

$$X \in [X_{min}, X_{max}], t \in [t_{begin}, t_{end}]$$

$$X \in \Omega_X \text{ , } t \in \Omega_t$$

where M is the number of conflicting objective functions, $n_g(t)$ and $n_h(t)$ are the number of inequality and quality constraints at time t respectively. X is a set of bounded decision variables with n-dimensional search space generated between minimum boundary (X_{min}) ,and maximum boundary (X_{max}). $F(X,t)$ is the objective vector that optimize the solution X at time t. $\Omega_X \subseteq \mathbb{R}^n$ is the decision space, and $\Omega_t \in \mathbb{R}$ is the time space bounded between a starting time t_{begin} and end time t_{end} . The objective vector is denoted by $F(X,t)\colon \Omega_X \times \Omega_t \to \mathbb{R}^{M_t}$ presenting the resulting values for each solution X at time t.

Table 1 presents five categories of DMOEAs have been proposed for solving DMOPs. The first category is the standard MOEAs designed for static multi-objective optimization and considered for DMOPs without any additional change detection mechanism. Among them are the Strength Pareto Evolutionary Algorithm 2 (SPEA2) [29], Non-dominated Sorting Genetic Algorithm II (NSGA-II) [2], MOEA/D [25], and dCOEA [26].

Table 1. Dynamic Multi-Objective Optimization Methods

Existing Approaches for DMOPs		Tested Benchmarks
Transfer Learning-based Methods	MMTL-MOEA/D, Jiang *et al.*, (2020) [16]	FDA1-5 [1], dMOP1-3 [26]
	RI-MOEA/D, Jiang *et al.*, (2020) [16]	
	PPS-MOEA/D, Jiang *et al.*, (2020) [16]	
	KF-MOEA/D, Jiang *et al.*, (2020) [16]	
	SVR-MOEA/D, Cao *et al.*, (2020) [21]	FDA1-5 [1], dMOP1-3 [26] and the DF test suite [33]
	Tr-MOEA/D, Jiang *et al.*, (2018) [22]	dMOP1-3 [26], FDA1-5, DIMP2, DMOP2, HE2,7,9 [1]
Diversity-based Approaches	Dynamic-MOPSO, Aboud *et al.*, (2017) [18]	FDA1 [1] and DIMP2, dMOP3 [26]
	DC-NSGA-II, Azzouz *et al.*, (2015) [34]	DCTPs test problems [34]
	DNSGA-II-A and DNSGA-II-B, Deb *et al.*, (2007) [9]	Modified FDA2 and hydro-thermal scheduling problem[9]
Memory-based Approaches	A-Dy-NSGA-II, Azzouz *et al.*, (2017) [35]	FDA1, FDA2 [1], DMZDT test functions and WYL
	SGEA, Jiang *et al.*, (2016) [19]	5 FDA [1], 6 dMOP [26], 6 ZJZ [27], and 7 UDF [36].
	dCOEA, Goh*et al.*, (2009) [26]	FDA1 [1] and 3 dMOP [26]
Prediction-based Approaches	Prediction-based methods (average, weighted, and adaptively methods) among MOEAs, Guo *et al.*, (2019) [24]	FDA1-5 [1], dMOP1-2 [26], and UDF [36] functions
	MOEA/D-KF, Muruganantham *et al.*, (2016) [20]	FDA1-5 [1], dMOP1-2 [26], and F5-10 [27]
	PPS, Zhou *et al.*, (2014) [27]	FDA1, FDA4 [1] and dMOP1, dMOP2 [26], F5-F8 [27]
Parallel Approaches	DMOEA, Liu *et al.*, (2007) [8]	FDA1, modified FDA2, FDA3, FDA4 and FDA5 [1]
	PSFGA, Camara *et al.*, (2007) [37]	FDA1 and FDA2 [1]
Static MOEAs adapted for DMOPs	SPEA2, Kim *et al.* (2004) [29]	DSW1, DSW2, DTF, FDA DMOPs [1]
	MSOPS, Hughes (2003) [38]	
	NSGA-II, Deb *et al.*, (2002) [2]	

Furthermore, a set of Transfer Learning-based (TL) methods were proposed to construct a predictive model that can learn from previous experience to predict future changes. Generally, TL methods have been proposed based on the baseline MOEA/D system [25]. The modified MOEA/D algorithm with random initialization (RI-MOEA/D) [16] generates 10% of the new population after the environmental change. The Tr-MOEA/D algorithm [22] integrates transfer learning and population-based evolutionary algorithms (EAs) to solve the DMOPs. Tr-MOEA/D algorithm determines the best potential space to select the optimal values of numerous hyper-parameters after each change. Memory-driven Manifold Transfer Learning algorithm (MMTL-MOEA/D) [16] aims to predict the initial population based on the elite of the previous solutions. The Kalman Filter (KF) prediction model based on MOEA/D (KF-MOEA/D) [20] uses a prior estimation to predict the posterior experience. An alternative Support Vector Regression MOEA/D (SVR-MOEA/D) [21] has also been proposed to predict the new population over time.

The PPS system [27] is an auto-regressive model that aims to predict the center point and the manifold based on previous experience. However, a new variant is denoted PPS-MOEA/D [17], where the original algorithm is modified using the MOEA/D system [25]. Different diversity-based approaches maintain population diversity through a re-initialization or a hyper-mutation behavior after each transition or throughout the execution time. DNSGA-II-A and DNSGA-II-B [9] select a random solution, and assess its re-evaluation to monitor any environmental change. The Dynamic Constrained NSGA-II (DC-NSGA-II) [39] proposed an adaptive penalty function to deal with time-varying constraints. To maintain diversity over time, the Dynamic-MOPSO system [18] controls the evolution of the objective function and reinitializes non-suitable solutions. A memory-based method is proposed in the dynamic Competitive-Cooperative Co-Evolutionary Algorithm (dCOEA) [26]. The Adaptive Dynamic NSGA-II (A-Dy-NSGA-II) [28] is proposed with an additional memory mechanism to implicitly or explicitly store outdated individuals for potential use. Steady-State Generational Evolutionary Algorithm (SGEA) [19] aims to reuse outdated solutions with good distribution to relocate solutions closer to the new POF. Different parallel methods are proposed based on evolutionary [8] and genetic [37] algorithms including multiple sub-populations for solving DMOPs. However, a set of performance metrics [14] have been considered for accuracy, diversity, and robustness measurements. Ultimately, convergence and diversity are measured using IGD, MIGD, and HVD quality indicators. In addition, Guo *et al.* [24] have proposed three prediction-based methods using average, weighted, and adaptively ensemble prediction mechanisms among five classical multi-objective evolutionary algorithms: SPEA2 [29], MOPSO [17], NSGA-III [30], MOEA/D [25], and MOEA/D-DE [31], and compared using RIGD and RGD metrics for solving a set of DMOPs.

Many DMOEAs got limited results to assume good convergence and diversity in the dynamic search space. To sum up, the diversity-based approaches [9], [18], [34] have shown their ability for solving dynamic problems with continuous and small time-varying parameters and show their limits in problems with severe environmental changes. Furthermore, many DMOPs have presented some periodical or recurrent changes making storing historical experience of solutions useful to preserve diversity. Memory-based approaches [19], [35], [26] use a redundant representation of an evolutionary algorithm using extra-memory components to help detect future changes. The memory-based approaches are very effective to solve DMOPs with periodically time-varying properties. However, such mechanisms slow down the convergence and strengthen diversity in DMOEAs. The main disadvantage of memory-based algorithms is the ineffectiveness of redundant solutions stored in the archive. On the other hand, prediction-based methods [20], [24], [27] tend to predict changes based on limited patterns. Such systems can detect the global best solution quickly but they fail when the changes are stochastic which increases their relative training error rates. The parallel approaches [8], [37] present an optimization process over multiple sub-swarms that may handle the problem on separate search space and are recommended for multi-modal problems while being computationally expensive. A key challenge for these methods is finding the appropriate number of sub-swarms and their sizes. Finally, the transfer learning-based methods [16], [21], [22] have the advantage to re-use previous computational experience to improve the efficiency of the new generated populations after each change detection by adding transfer learning mechanisms which is a time-consuming process. Furthermore, the No Free Lunch (NFL) theorem [40] proposed by Wolpert and Macready in 1997 logically proves that no optimization algorithm can solve all problems. This means that the algorithm is useful for a set of problems, but not efficient for other types. More details about the proposed DPb-MOPSO are presented in section 3.

To assess the quality of metaheuristics for DMOPs proposals are compared over DMOPs testbeds which are in general presented in Table 2 and organized as follows: [1], [26], [27], [36].

- five continuous FDA functions [1] with dynamic shift on POF and POS,
- three dMOP problems [26],
- five ZJZ (F5 to F10) [27] with the non-linear correlation between decision variables,
- nine Unconstrained Dynamic Functions (UDFs) [36].

Table 2. Characteristics and Parameter Settings of DMOPs

DMOPs		Number objectives	Variable's dimension	Type	POS Change Nature	POF Change Nature	
						Shift	Curvature
FDA problems, Farina *et al.* [1]	FDA1	2	20	I	sinusoidal change + vertical shift	no change	no change + convex
	FDA2	2	15	II		vertical + density of solutions changes	cyclic change: convex to concave
	FDA3	2	30	II		vertical + spread of solutions change	cyclic change: convex
	FDA4	3	12	I		no change + spread of solutions change	no change + nonconvex
	FDA5	3	12	II		radius variations + dynamic spread	change + nonconvex
dMOP problems, Goh *et al.* [26]	dMOP1	2	10	III	no change	vertical	change: convex to concave
	dMOP2	2	10	II	sinusoidal change + vertical shift	vertical	change: convex to concave
	dMOP3	2	10	I	sinusoidal change + vertical shift	no change	no change + convex
ZJZ problems, Zhou *et al.* [27]	F5	2	20	II	trigonometric + vertical shift	vertical	change: convex to concave
	F6	2	20	II			
	F7	2	20	II			
	F8	3	20	II			
	F9	2	20	II			
	F10	2	20	II			
UDF problems, Biswas *et al.* [36]	UDF1	2	10	I	trigonometric+ vertical shift	diagonal	no change + linear + continuous
	UDF2	2	10	I	polynomial+ vertical shift	vertical	no change + linear + continuous
	UDF3	2	10	III	trigonometric + no variation	diagonal	linear + discontinuous
	UDF4	2	10	II	trigonometric + horizontal shift	angular shift	convex to concave + continuous
	UDF5	2	10	II	polynomial + vertical shift	angular shift	change: convex to concave + continuous
	UDF6	2	10	III	trigonometric + no variation	diagonal + angular	linear + discontinuous
	UDF7	3	10	III	trigonometric + no variation	shifting of the center	radius concave + 3D

3. The Proposed Dynamic Pareto bi-level Multi-Objective Particle Swarm Optimization Algorithm

Considering the sate-of-the-art and the level of obtained results three key problems rises which motivated the dynamic bi-level MOPSO proposal. The main motivation for this contribution is:

- ✓ The lack of dynamism handling strategy in some existing approaches [15], [17], [28] leads to premature convergence and the loss of diversity in dynamic search space. Even, the existing mechanisms [16], [19],[24] are not able to detect and respond efficiently to all types of the dynamic change.
- ✓ The outdated memory is the main issue for different multi-objective evolutionary algorithms [15], [17], [28] for solving DMOPs when the set of non-dominated solutions cannot be conserved due to the dynamic change of both decisions variables and objective values.
- ✓ The most existing dynamic MOEAs are developed based on the Genetic Algorithm [9], [30] or the Evolutionary Algorithm [21], [22], [25], [26], [27], [29], [31]. Few methods have been developed based

on the MOPSO approach [18] to overcome the dual issue of the outdated memory and the loss of diversity over time.

To overcome these issues the proposed contribution main features are the following:

- ✓ The new DPb-MOPSO algorithm includes a search space control allowing to manage multiple sub-swarms in a specific different potential solutions space. The distributed sub-swarms conduct independent searches with separate pools of best solutions.
- ✓ The proposed DPb-MOPSO includes a dynamic handling strategy which allows managing the evolution of the search toward optimality.
- ✓ The two dynamic optimization levels allow to recover from any engagement within a local optimum.
- ✓ The new DPb-MOPSO algorithm adopts a distributed architecture with two parallel optimization levels. For the upper level (L_1), the swarm acts in a single search space, pushing all particles to the best-compromise solutions. Nevertheless, if a dynamic change is successfully detected by monitoring the objective values at the first level. The second level provides an exploitation enhancement over a dynamic multiple sub-swarm's subdivision based on the Pareto ranking operator.
- ✓ The DPb-MOPSO proposes a dynamic handling strategy to detect and effectively react to the change of POS and POF. For the detection strategy, a set of random solutions $POS(t)$ has been used as detectors and compared with the previous solutions set $POS(t-1)$. The response strategy consists in reevaluating all solutions with negative improvement and replaced them with new random solutions. Solutions with positive progress are transferred to build a new population to speed up the optimization process after each change. Then, all non-dominated solutions in the leader's archive are re-evaluated to avoid outdated memory issues

The general flowchart of the proposed Dynamic Pareto bi-level Multi-Objective Particle Swarm Optimization (DPb-MOPSO) algorithm is shown in Figure 1. DPb-MOPSO has two optimization levels are detailed in Algorithms 2 and 3 respectively, and denoted by the upper level (L_1), and the lower level (L_2). The two levels have included a dynamism handling strategy presented in subsection 3.3 to detect and react to changes. The DPb-MOPSO algorithm consists of the following iterative steps:

3.1. DPb-MOPSO: Upper Level ($\mathbf{L_1}$)

The details of the optimization process at the first level L_1 are shown in Figure 2.

- **Step 1 in $\mathbf{L_1}$:** The upper level (L_1) starts with random initialization of the position (X) of N particles p_i in swarm S (see Algorithm 1). Each particle is a candidate solution X_i^* in d-dimensional search space.
- **Step 2 in $\mathbf{L_1}$:** For each iteration, all particles are evaluated to calculate their fitness function F.
- **Step 3 in $\mathbf{L_1}$:** Then, all non-dominated solutions of L_1 are stored in the leader's archive A based on the crowding distance mechanism [41]. All non-dominated solutions are sorted in ascending order of objective values, and the crowning distance is calculated at each iteration using the largest cuboid to estimate the density of solutions surrounding a particular solution X_i^* for each objective value. The boundary solutions with the highest and lowest objective values are selected. The particle with the lowest crowding distance is selected as a global best solution ($gbest$). The personal best solution ($pbest$) is determined by historical personal best experience.
- **Step 4 in $\mathbf{L_1}$:** At discrete time (t), the position (X_t) and velocity (V_t) for each particle are updated using Equations (2) and (3).

$$X_{t+1} = X_t + V_{t+1} \quad \textbf{(2)}$$

$$V_{t+1} = wV_t + c_1r_1(pbest(t) - X_t) + c_2r_2(gbest(t) - X_t) \quad \textbf{(3)}$$

where w is the inertia weight, c_1 and c_2 are random acceleration coefficients. r_1 and r_2 are two random parameters designed to affect the cognitive and social components of each particle. $pbest(t)$ and $gbest(t)$ are the best personnel position and the global leader, respectively. The new velocity V_{t+1}, is updated based on the best self-experience $pbest(t)$ as a local component and the global experience of all neighbors $gbest(t)$ as a global component.

Algorithm 1: Swarm Initialization

Input: ***P*** (swarm of N particles), ***N*** (size of the population)

Output: Set of non-dominated solutions in the leader's archive ***A***

1. Initialize t=0;
2. **for** i = 1 **to** N **do**
3. Randomly initialize the vector of position $\boldsymbol{X_i}$ and zero velocity $\boldsymbol{V_i}$;
4. Evaluate fitness function for the particle $\boldsymbol{p}_\mathbf{i}$;
5. Initialize the personal best solution pbest ← $\boldsymbol{p}_\mathbf{i}$;
6. **end for**
7. Store the non-dominated solutions in the leader's archive ***A***

- **Step 5 in $\mathbf{L_1}$:** If the fitness function is changed, the lower level (L_2) is started in a parallel execution with L_1 (see Algorithm 3) and more details are presented in the subsection 3.2. The dynamic change is detected by controlling the evolution and deterioration of the fitness functions of particles. The DMOP has a cyclic time-varying parameter T presented in Equation (4), and changes independently at each periodic frequency τ_t. After some periodic time, equal to the frequency τ_t the dynamic change is checked by comparing the current population $POS(t)$ with the previous one $POS(t-1)$, more details are presented in the subsection 3.3.

$$T = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_t} \right\rfloor \quad \textbf{(4)}$$

where; n_t, and τ are the severity of the change and the iteration counter respectively, $\forall\, t \in T_{max}$. Most existing studies [16], [19] fix the frequency τ_t at 5, 10 and 20 respectively indicating severe, moderate, and slight changes in the environment.

- **Step 6 in $\mathbf{L_1}$:** In each optimization step, all non-dominated solutions X_i^* in L_1 are stored in the leader's archive A, presenting the best-approximated solutions to the true POF. Furthermore, the leader's archive A is a shared component between two levels, and used to select the best global leader (gbest). So, the leader's archive A is a finite component, if it is full all non-dominated solutions are sorted in descending crowding distance value, and one random particle from the bottom portion is replaced by another non-dominated solution.
- **Step 7 in $\mathbf{L_1}$:** The optimization process in L_1 is performed until a maximum number of iterations T_{max} is reached.

Algorithm 2: DPb-MOPSO upper level (L_1)

Input: $\boldsymbol{P}$ (swarm of N particles), $\boldsymbol{N}$ (size of the population), $\boldsymbol{T_{max}}$ (max-iterations)
Output: Set of non-dominated solutions in the leader's archive $\boldsymbol{A}$
Swarm initialization (see Algorithm 1)
while $\mathbf{t \leq T_{max}}$ **do**
  if the fitness function has changed **then**
    start the level $\boldsymbol{L_2}$ as in the **Algorithm 3**;
  end if
  for $\mathbf{i = 1}$ **to N do**
    Select the leader **gbest** using the epsilon dominance comparator among two random particles in A;
    Evaluate the fitness function **F** of each $\boldsymbol{p_i}$;
    Current population ← record the previous solutions set $\mathbf{POS(t-1)}$;
    Determine **pbest** and gbest;
    Update the positions $\mathbf{X_i}$ and the velocity $\mathbf{V_i}$ using equations 2 and 3;
    // Apply the dynamism handling strategy.
    for each interval of time = $\boldsymbol{\tau_t}$ **do**
      //Select detectors
      Current population ← 10% of $\mathbf{POS(t)}$ as random detectors;
      Previous population ← 10% of their projection in previous $\boldsymbol{POS(t-1)}$;
      for $\mathbf{i = 0}$ **to** $\mathbf{|POS(t)|}$ **do**
        //Change detection strategy.
        Compare objective values of $\mathbf{POS(t)}$ vs. $\mathbf{POS(t-1)}$ using strong epsilon-dominance operator;
        if $\mathbf{(1+\varepsilon) \times F_i(t) \geq F_i(t-1)}$ **then**
          Negative change ← true;
          Select $\mathbf{POS(t)}$ with negative changes;
          Select $\mathbf{POS(t)}$ with positive changes;
          //Response strategy.
          Re-initialize all solutions with negative changes in $\mathbf{POS(t)}$;
          Construct population $POS(t+1)$ using positive $POS(t)$;
          Update the leaders archive $\boldsymbol{A}$;
        end if
      end for;
  end for
  Update the leader's archive $\boldsymbol{A}$;
  t ← t+1;
end while;
Return the best non-dominated solutions from the archive $\boldsymbol{A}$;

3.2. DPb-MOPSO: Lower Level (L_2)

- **Step 1 in L_2:** When fitness function values changed during the optimization process of L_1 (see Algorithm 2), the second level was performed on a distributed architecture using multiple sub-swarms based on the Pareto ranking operator. The parallel execution of the second lower level (L_2) begins with the first level (L_1) on the basis of the first population P'.

- **Step 2 in $\mathbf{L_2}$:** The Pareto ranking operator [2] is applied to dynamically split the extracted population P' into K fronts denoted by $F_0, \dots, F_{k-1}$. The Pareto front analysis strategy is used to fix K while it was detailed in Algorithm 4. All non-dominated solutions are considered as the first front (F_0), and then the next front is obtained by discarding all non-dominated solutions of the first front, and determining the second front (F_1) of non-dominated solutions using the rest of the population. This process is repeated until each solution belongs to a front. In the DPb-MOPSO algorithm, the number of sub-swarms $K = \frac{P'}{|F_0|}$ is fixed through the analysis of the Pareto non-dominated solutions in F_0 . K=2 is the minimal sub-warming behavior in which the algorithm uses two independent sub-populations to process.
- **Step 3 in $\mathbf{L_2}$:** The Pareto ranking operator based on non-dominance aims to generate K independent sub-swarms with different sizes defined toward a set of non-dominated solutions by subdividing the size of the original populations over K. To assume an equal size for all sub-swarms, the process of parameters configuration for each sub-swarm is a primordial step to ensure an equal size between all sub-populations. To set the parameters of K sub-swarms (size and injected solutions), a set of additional individuals are randomly selected from the solution set $POS(t)$ performed on L_1 and injected into each front. The proposed DPb-MOPSO algorithm considers at least 2 fronts ($K > 1$) to be generated.
- **Step 4 in $\mathbf{L_2}$:** Moreover, all K sub-swarms are optimized in parallel until the maximum number of iterations (T_{max}) as presented in Algorithm 3, performing the same process in Figure 2 including the dynamism handling strategy detailed in subsection 3.3.
- **Step 5 in $\mathbf{L_2}$:** All non-dominated solutions of L_2 from the parallel K instance of the DPb-MOPSO algorithm are stored in the archive A. At the end of the two optimization levels the best non-dominated solutions in archive A are generated as the output of the proposal.

Algorithm 3: DPb-MOPSO lower level ($\mathbf{L_2}$)

Require: $\boldsymbol{F_i} = \{\boldsymbol{F_{i=0}}, \dots, \boldsymbol{F_{i=k-1}}\}$ (set of k fonts),$\mathbf{T_{max}}$ (maximum iterations);
Output: **A** (leader's archive);

1. Obtain k sub swarms using the **Algorithm 4**
2. //Execute K parallel instances of the proposed DPb-MOPSO algorithm
3. **while** $\mathbf{t \leq T_{max}}$ **do**
4.   **for** $\mathbf{i = 0}$ **to** $\mathbf{K - 1}$ **do**
5.   Create $\boldsymbol{F_i}$ instance of the DPb-MOPSO algorithm (details are in Figure 2);
6.   **end for**
7.   **do in parallel**
8.   Run parallel $\boldsymbol{K}$ instances of **DPb-MOPSO** $(\frac{\boldsymbol{population\ P'}}{|F_0|})$;
9.   Store the non-dominated solutions on the archive $\boldsymbol{A}$;
10.   **end**
11. **end while**
12. Return the best non-dominated solutions in the archive **A**;

Algorithm 4: Obtain K Sub-swarms

Require:$\mathbf{P'}$ (population of $\boldsymbol{L_1}$)

Output: $\boldsymbol{F_i} = \{\boldsymbol{F_{i=0}}, \ldots, \boldsymbol{F_{i=k-1}}\}$ (set of k fronts)

Select the intermediate population $\mathbf{P'}$of $\mathbf{L_1}$;
Initialise a set of non-dominated solutions $\boldsymbol{S_p} \leftarrow \emptyset$;
Initialise a set of dominated solutions of the next front $\boldsymbol{Q} \leftarrow \emptyset$;
Initialise dominated solutions counter $\boldsymbol{n_p} = \mathbf{0}$;
Initialise the front counter $\boldsymbol{K} = \mathbf{1}$;
for $\boldsymbol{p} = \mathbf{1}$ **to** $|\mathbf{P'}|$ **do** //construct k sub-swarms using the non-dominance rank
  for $\boldsymbol{q} = \boldsymbol{p} + \mathbf{1}$ **to** $|\mathbf{P'}|$ **do**
    if $\boldsymbol{solution_p}$ **dominate** $\boldsymbol{solution_q}$ **then**
      $\boldsymbol{S_p} \leftarrow \boldsymbol{S_p} \cup \boldsymbol{solution_q}$; //store non dominated solutions q into $\boldsymbol{S_p}$
      $\boldsymbol{n_p} = \mathbf{0}$;
    else if $\boldsymbol{solution_q}$ **dominate** $\boldsymbol{solution_p}$ **then**
      $\boldsymbol{n_p} = \boldsymbol{n_p} + \mathbf{1}$;
    end if
  end for
  if $\boldsymbol{n_p} = \mathbf{0}$ **then**
    $\boldsymbol{F_0} = \boldsymbol{F_0} \cup \{\boldsymbol{solution_p}\}$; //Determine the first front $\boldsymbol{F_0}$
  end if
end for
//Apply the pareto ranking operator to subdivide $\mathbf{P'}$ into $\boldsymbol{K} = \frac{|\mathbf{P'}|}{|\boldsymbol{F_0}|}$ fronts ranked from $\mathbf{1}$ to $\boldsymbol{k}$
Determine the number of generated k sub-swarms $\boldsymbol{K} = \frac{|\mathbf{P'}|}{|\boldsymbol{F_0}|}$ (k is an integer)
Determine the size of $\boldsymbol{K}$ sub-swarm $|\mathbf{F}_{\boldsymbol{K}}| = \frac{|\mathbf{P'}|}{\boldsymbol{K}}$;
while $\boldsymbol{F_K} \neq \emptyset$ **do**
$\boldsymbol{n_q} = |\boldsymbol{S_p}|$;
  for $\boldsymbol{p} = \mathbf{1}$ **to** $|\boldsymbol{F_K}|$ **do**
    for $\boldsymbol{q} = \mathbf{1}$ **to** $|\boldsymbol{S_p}|$ **do**
//Determine the non-dominated solutions $\boldsymbol{q}$ from rest of the population $\boldsymbol{S_p}$
      $\boldsymbol{n_q} = \boldsymbol{n_q} - \mathbf{1}$;
      if $\boldsymbol{n_q} = \mathbf{0}$ **then**
        $\boldsymbol{K} = \boldsymbol{K} + \mathbf{1}$;
        $\boldsymbol{Q} = \boldsymbol{Q} \cup \{\boldsymbol{solution_q}\}$;
      end if
    end for
  end for
  $\boldsymbol{K} = \boldsymbol{K} + \mathbf{1}$;
// Set the non-dominated solutions in the next front k,
// Adjust the size of sub-swarm equal to the size of the population $\mathbf{P'}$ divided by the size of the first front $\boldsymbol{F_0}$
  for $\boldsymbol{i} = \mathbf{1}$ **to** $\boldsymbol{i} < \boldsymbol{K}$ **do**
    $\boldsymbol{F_i}$= Q
    if $|\mathbf{F}_{\boldsymbol{i}}| < \frac{|\mathbf{P'}|}{\boldsymbol{K}}$ **then**
      //Inject random solutions from the intermediate population $\mathbf{P'}$
      Set the number of the injected solutions $\boldsymbol{I_p} = \frac{|\boldsymbol{P'}|}{\boldsymbol{K}} - |\boldsymbol{F_i}|$
      $\boldsymbol{I_p} \leftarrow \boldsymbol{I_p} \cup \{\boldsymbol{solution_p}\}$;
      $\boldsymbol{F_i} = \boldsymbol{Q} \cup \boldsymbol{I_p}$;
    end if
  end for
end while
Return the set of $\boldsymbol{K}$ fronts $\boldsymbol{F_i} = \{\boldsymbol{F_{i=0}}, \ldots, \boldsymbol{F_{i=K-1}}\}$ (sub-swarms);

3.3. DPb-MOPSO: Dynamic Changes Handling Strategy

Dynamic changes detection aims essentially to help a better search process and to prospect any interesting solutions. The two steps of dynamism handling of the proposed algorithm are detailed in lines 12 to 25 in the pseudo-code of Algorithm 2, and detailed as follows:

- **Step 1: Environment change detection strategy**

The environmental changes are checked periodically at each frequency τ_t equal to 5, 10, and 20 showing severe, moderate, and slight changes respectively. All historical values of the previous population $POS(t-1)$ and the current $POS(t)$ are recorded at each iteration. By selecting 10% of the previous $POS(t-1)$ as random detectors, the comparison process starts after evaluating of the fitness function to obtain the new optimized population. Then, select 10% of the current mapping solutions in $POS(t)$ to control their evolution over time. The strong dominance operator of the Epsilon-Dominance method [42] is used to compare the time-varying behavior of each particle between t and $t-1$. A solution S_1 is ϵ-dominate a second solution S_2 for some $\epsilon > 0$ and denoted by $S_1 >_{\epsilon} S_2$, if and only if for all F_i objectives $\forall, i = 1, \dots, M$, $F(S_1^*) = \{F_1, \dots, F_M\}$, and $F(S_2) = \{F_1, \dots, F_M\}$ such that $(1+\epsilon)\, F_i(S_1) \geq F_i(S_2)$. For all M objectives, if S_t is strictly better than S_{t-1}, $(S_t \succ S_{t-1})$, then the solution S_t strongly dominates the solution S_{t-1}.

- **Step 2: Response change strategy**

If the change is successfully identified, all solutions $POS(t)$ with negative changes are reinitialized. To assume that only solutions with positive effects have good diversity and convergence, all detectors with positive changes $POS(t)$ are transferred to build a new population at $t+1$. In addition, the leader's archive (A) is updated to overcome the deterioration of research capabilities.

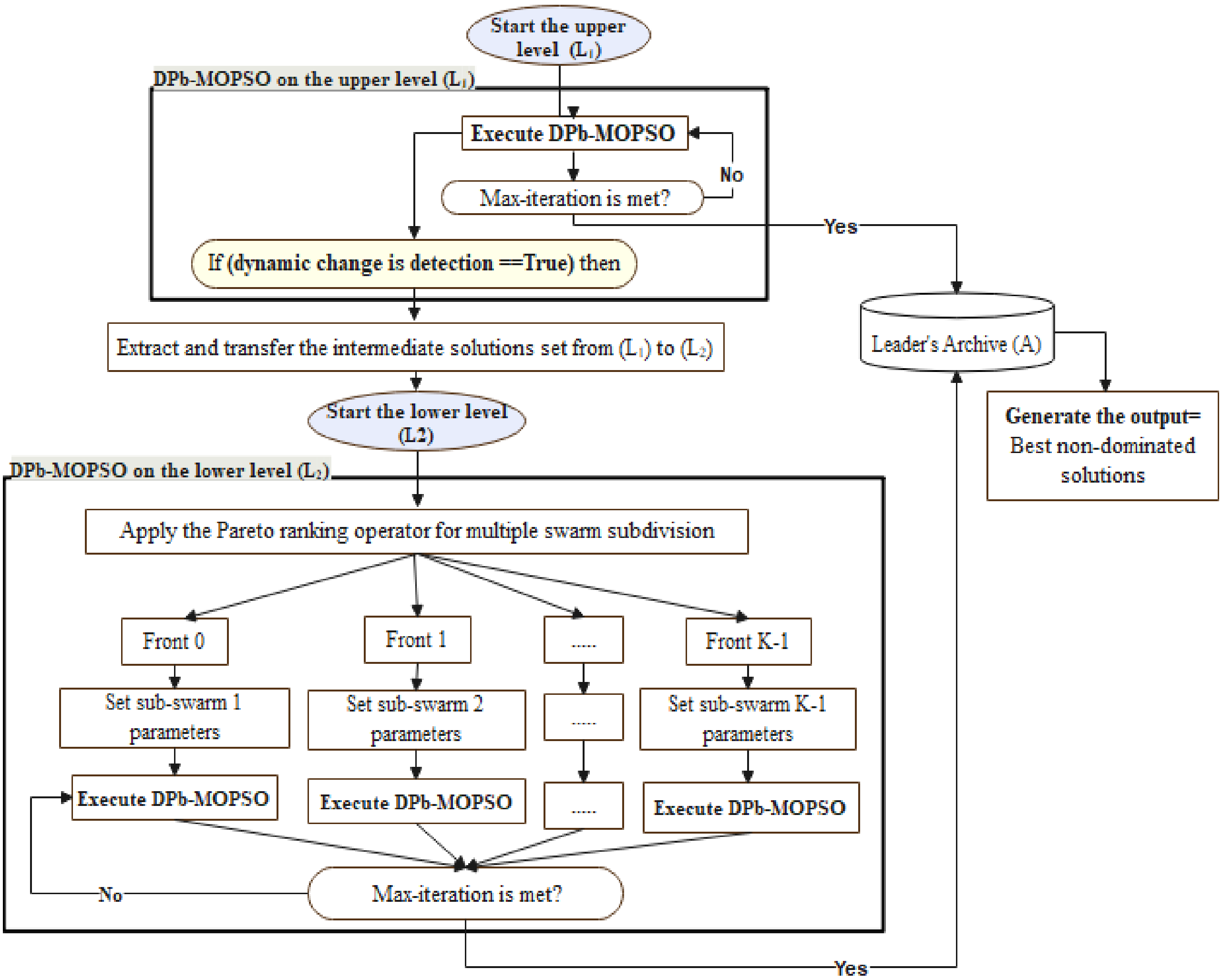


Fig. 1. The Distributed Flowchart of the Proposed Dynamic Pareto bi-level Multi-Objective Particle Swarm Optimization (DPb-MOPSO) Algorithm.

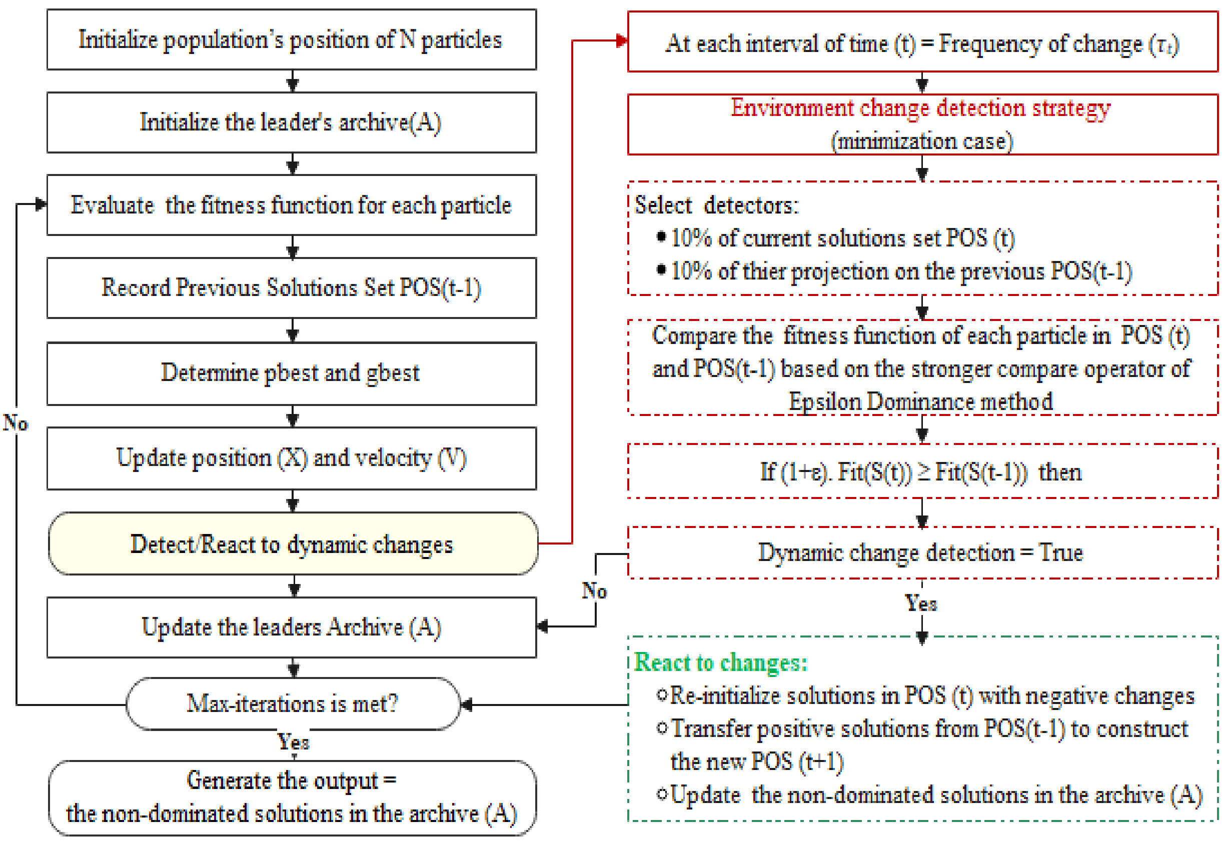


Fig. 2. The Detailed Steps of DPb-MOPSO in the Upper and Lower Levels.

3.4. Computational Complexity of DPb-MOPSO Algorithm

The computational complexity of the proposed DPb-MOPSO algorithm was measured based on the asymptotic growth of the algorithm behaviors in terms of the time complexity. As detailed in Algorithm 1, the DPb-MOPSO algorithm starts with random initialization of both properties: positions X_i with d-dimensional search space, and velocities (V_i) for a swarm of N particles. The proposed algorithm aims to optimize a swarm of N particles, each particle is a candidate solution, and the objective function F is performed until a maximum number of iterations (T_{max}) is reached.

- The initialization procedure in Algorithm 1 requires $O(N * d)$ times, where N is the size of the population, and d is the dimension of search space. At the first iteration, all non-dominated solutions are stored in the leader's archive (A). The process to compare each solution and determining the set of non-dominated solutions takes $O(M * N)$, where M is the number of objectives.
- Moreover, the main loop for the first upper level (L_1) in the Algorithm 2 is executed until the maximum number of iterations (T_{max}) is reached, and each iteration takes $O\ (N * d)$. At each iteration, all steps in Algorithm 2 (lines 2 to 28) are executed. The running time of the DPb-MOPSO algorithm for each level consists of N iterative loops performing logarithmic statements during T_{max} iterations and takes a complexity time ct_1:
- $ct_1 = O(N * d\ log(T))$ times.

  where;

  - N is the size of the population,
  - d is the dimension of the search space,

- T is the running time.

- However, if the dynamic change is usefully detected in the objective function, the main loop for the second lower level (L_2) in the Algorithm 3 is started in parallel execution to the first upper level until the maximum iteration (T_{max}) based on K fronts. In the worst case, the process to find N dominated solutions for K front takes a complexity time ct_2:
- $ct_2 = O(M * N^2)$

  where;

  - M is the number of objectives
- The update of particle positions is being preceded by determining the global best solution ($gbest$), and the personal best position ($pbest$) from the leader's archive (A) with a complexity of $O(N)$.
- Furthermore, the fitness function is re-evaluated within a complexity of $O(N)$.
- At each time t, the leaders archive (A) is updated using the dominance operator, then all dominated solutions are removed and replaced with a set of non-dominated solutions and takes $O(N)$.
- Assuming that all particles are pre-sorted, the determination of each loop takes $O(N * d)$ times.

To sum up, the overall complexity of the proposed DPb-MOPSO algorithm is equal to: $ct_1 + ct_2 = O(N * d\ log(T)) + O(M * N^2)$.

4. Experimental Study

This section describes the Orthogonal Experimental Design (OED) of the proposed DPb-MOPSO Algorithm. It also describes the parameters settings for all comparable DMOEAs, the testbeds and the quality indicators used for the experimental study.

4.1. Orthogonal Experimental Design for DPb-MOPSO Algorithm

In this subsection, the parameters design of the proposed DPb-MOPSO algorithm is systematically analysed using the Taguchi's method [43] as implemented in [44]. The Taguchi's method was proposed by Genichi Taguchi in 2004 as a robust engineering method to study the parameter design optimization and tolerance design. In this experiment, a reasonable combinations of parameters design are considered using the orthogonal arrays (OAs) of the Taguchi's method aiming to minimize the number of runs (or combinations) needed for the experiment. The OAs are denoted by $L_a(b^c)$, where a is the number of experimental runs, b is the number of levels of each factor, c is the number of columns in the array, and L denotes Latin square design. In this research, the proposed DPb-MOPSO algorithm consists of 11 key parameters (also known as control factors). In contrast, an array of three level factors is used, where all the parameters are varied with three levels and fixed in accordance with some published papers [16] and [19], and set as follows:

- The swarm size (N), $N \in \{100, 150, 200\}$.
- The maximum number of iterations (T_{max}), $T_{max} \in \{200, 350, 650\}$.
- The archive size (A), $A \in \{100, 150, 200\}$.
- The sub-swarm size (K), $K \in \{10, 20, 30\}$.
- The number of fonts (f), $F \in \{5, 10, 15\}$.
- Two cognitive parameters $r1$ and $r2$, r1 and r2 $\in \{0.1, 0.5, 0.9\}$.
- Two acceleration coefficients $c1$ and $c2$, c1 and c2 $\in \{1.5, 1.8, 2.0\}$.

- The inertia weight (w), $w \in \{0.1, 0.5, 0.9\}$.
- The rate of detectors $\in \{10, 50, 100\}$.

A full-factorial analysis needs $3^{11} = 177\,147$ experimental runs. The application of the Taguchi orthogonal arrays has identified an array of $L_{27}(3^{11})$ with only 27 best runs for different combination of parameters design. Tables 1 - 21 in Supplementary Material show the experimental results based on the orthogonal array of the Taguchi method and factor assignment aiming to analyse the sensitivities of user-defined parameters. The mean values for the three quality indicators HVD, IGD and MIGD are displayed over 30 runs of the proposed DPb-MOPSO algorithm for 21 benchmarks (5 FDA, 3 dMOP, 7 UDF and 6 F (ZJZ) testbeds). As results, Tables 3 and 4 reported the best design of experiment over acceptable user-defined parameter settings for the proposed DPb-MOPSO algorithm based on 11 control factors with three levels.

Table 3. Design of Experimental using Taguchi Method for **HVD** Metric with Severe, Moderate and Slight Dynamic Changes for 21 DMOPs.

Type of changes	**DMOPs**	**Swarm size**	**Max-iterations**	**Archive size**	**Sub-swarm size**	**Number of fonts**	**r1**	**r2**	**C1**	**C2**	**w**	**Rate of Detectors**
Severe changes	FDA1	200	200	200	20	15	0,5	0,1	2	1,8	0,1	100
	FDA2	200	200	200	20	10	0,1	0,9	1,8	1,5	0,9	50
	FDA3	200	200	200	20	15	0,5	0,1	2	1,8	0,1	100
	FDA4	200	200	200	20	5	0,9	0,5	1,5	2	0,5	10
	FDA5	100	200	100	10	15	0,9	0,9	2	2	0,9	100
	dMOP1	150	200	150	30	15	0,1	0,5	2	1,5	0,5	100
	dMOP2	200	200	200	20	5	0,9	0,5	1,5	2	0,5	10
	dMOP3	200	200	200	20	15	0,5	0,1	2	1,8	0,1	100
Moderate change	FDA1	200	350	100	30	15	0,5	0,1	1,5	2	0,5	50
	FDA2	100	350	150	20	10	0,5	0,5	2	2	0,9	10
	FDA3	200	350	100	30	15	0,5	0,1	1,5	2	0,5	50
	FDA4	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	FDA5	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	dMOP1	100	350	150	20	15	0,9	0,9	1,5	1,5	0,1	50
	dMOP2	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	dMOP3	100	350	150	20	5	0,1	0,1	1,8	1,8	0,5	100
	UDF1	150	350	200	10	5	0,5	0,9	1,8	2	0,1	100
	UDF2	100	350	150	20	15	0,9	0,9	1,5	1,5	0,1	50
	UDF3	200	350	100	30	15	0,5	0,1	1,5	2	0,5	50
	UDF4	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	UDF5	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	UDF6	150	350	200	10	10	0,9	0,1	2	1,5	0,5	10
	UDF7	150	350	200	10	10	0,9	0,1	2	1,5	0,5	10
	F5	200	350	100	30	15	0,5	0,1	1,5	2	0,5	50
	F6	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	F7	200	350	100	30	15	0,5	0,1	1,5	2	0,5	50
	F8	150	350	200	10	5	0,5	0,9	1,8	2	0,1	100
	F9	200	350	100	30	15	0,5	0,1	1,5	2	0,5	50
	F10	200	350	100	30	15	0,5	0,1	1,5	2	0,5	50
Slight changes	FDA1	150	650	100	20	15	0,1	0,5	1,8	2	0,1	10
	FDA2	100	650	200	30	15	0,9	0,9	1,8	1,8	0,5	10
	FDA3	100	650	200	30	10	0,5	0,5	1,5	1,5	0,1	100
	FDA4	150	650	100	20	5	0,5	0,9	2	1,5	0,5	50
	FDA5	150	650	100	20	5	0,5	0,9	2	1,5	0,5	50
	dMOP1	200	650	150	10	15	0,5	0,1	1,8	1,5	0,9	10
	dMOP2	150	650	100	20	5	0,5	0,9	2	1,5	0,5	50
	dMOP3	200	650	150	10	5	0,9	0,5	2	1,8	0,1	50

Table 4. Design of Experimental using Taguchi Method for **IGD/MIGD** Metric with Severe, Moderate and Slight Dynamic Changes for 21 DMOPs.

Type of changes	**DMOPs**	**Swarm size**	**Max-iterations**	**Archive size**	**Sub-swarm size**	**Number of fonts**	**r1**	**r2**	**C1**	**C2**	**w**	**Rate of Detectors**
Severe changes	FDA1	150	200	150	30	15	0,1	0,5	2	1,5	0,5	100
	FDA2	150	200	150	30	15	0,1	0,5	2	1,5	0,5	100
	FDA3	200	200	200	20	15	0,5	0,1	2	1,8	0,1	100
	FDA4	200	200	200	20	5	0,9	0,5	1,5	2	0,5	10
	FDA5	200	200	200	20	15	0,5	0,1	2	1,8	0,1	100
	dMOP1	150	200	150	30	15	0,1	0,5	2	1,5	0,5	100
	dMOP2	200	200	200	20	5	0,9	0,5	1,5	2	0,5	10
	dMOP3	200	200	200	20	15	0,5	0,1	2	1,8	0,1	100
Moderate change	FDA1	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	FDA2	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	FDA3	200	350	100	30	15	0,5	0,1	1,5	2	0,5	50
	FDA4	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	FDA5	150	350	200	10	10	0,9	0,1	2	1,5	0,5	10
	dMOP1	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	dMOP2	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	dMOP3	200	350	100	30	15	0,5	0,1	1,5	2	0,5	50
	UDF1	150	350	200	10	5	0,5	0,9	1,8	2	0,1	100
	UDF2	100	350	150	20	15	0,9	0,9	1,5	1,5	0,1	50
	UDF3	200	350	100	30	15	0,5	0,1	1,5	2	0,5	50
	UDF4	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	UDF5	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	UDF6	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	UDF7	200	350	100	30	5	0,9	0,5	1,8	1,5	0,9	100
	F5	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	F6	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	F7	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	F8	150	350	200	10	10	0,9	0,1	2	1,5	0,5	10
	F9	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
	F10	200	350	100	30	10	0,1	0,9	2	1,8	0,1	10
Slight changes	FDA1	150	650	100	20	15	0,1	0,5	1,8	2	0,1	10
	FDA2	150	650	100	20	15	0,1	0,5	1,8	2	0,1	10
	FDA3	100	650	200	30	10	0,5	0,5	1,5	1,5	0,1	100
	FDA4	200	650	150	10	5	0,9	0,5	2	1,8	0,1	50
	FDA5	200	650	150	10	5	0,9	0,5	2	1,8	0,1	50
	dMOP1	200	650	150	10	5	0,9	0,5	2	1,8	0,1	50
	dMOP2	150	650	100	20	15	0,1	0,5	1,8	2	0,1	10
	dMOP3	100	650	200	30	10	0,5	0,5	1,5	1,5	0,1	100

4.2. Experimental Settings for All Comparable DMOEAs

To begin with, this section presents the empirical study referred to the contributions [16] and [19]. The proposed DPb-MOPSO approach is compared with different DMOEAs, including transfer learning-based methods in [16], DMOEAs in [19], MOPSO-based approaches [18], [15], [17], [28]. All algorithms are executed during 30 independent runs. Each run is executed during the maximum number of iterations $T_{max} = 3 \times n_t \times \tau_t + 50$, where; τ_t is the frequency of change fixed to 5, 10 and 20 respectively, and n_t is the severity of the change fixed to 10. Then, a set of MOEAs have been compared in [45], and enhanced as a prediction-based algorithms [24] for solving different DMOPs are compared to the proposed DPb-MOPSO algorithm. The parameters settings of all comparable DMOEAs have been fixed as suggested by the authors in the original contributions in [16], [19], and [24]. All parameters' settings are summarized in Table 5. The Java implementation of the proposed DPb-MOPSO algorithm is done using the jMETAL framework [46] on a personal computer with 8 Go of Ram, 1 To, and i7 intel processor.

Table 5. Parameters Settings of the Comparable DMOEAs.	
DMOEAs	**Parameters Settings**
MOPSO, dMOPSO, pbMOPSO, Dynamic-MOPSO, DPb-MOPSO	- Perturbation index= 1.0 - Mutation probability= 1.0/number of variables - r1 and r2= rand (0.0, 1.0) - c1 and c2= rand (1.5, 2.5) - w= rand (0.1, 0.4) - Swarm size=100 - Archive size=100 - Max-iterations= $3 \times n_t \times \tau_t + 50$ - Independent runs =30
dCOEA, PPS, MOEAD, SGEA, DNSGA-II, TL-methods	- Crossover probability=1.0 - Mutation distribution= 20 - Crossover distribution= 20 - Swarm size=100 - Archive size=100 - Max-iterations= $3 \times n_t \times \tau_t + 50$ - Independent runs =30
MOEA/D, MOEA/D-DE, SPEA2, MOPSO, NSGAIII	- Crossover probability =1 - Mutation probability= 1.0/number of variables - Neighborhood size = 20 - Archive size = 100 - Independent runs =20

4.3. Dynamic Multi Objective Problems

This study concerns 21 benchmarks including: five FDA [1], three dMOP [26], six F(ZJZ) [27], and seven UDF functions [36]. Table 2 lists all DMOPs parameter settings and the dynamic nature of POS and POF.

4.4. Quality Indicators

Firstly, IGD, MIGD and HVD are the three quality indicators that have been considered for convergence and diversity measurements, and the smallest value is the best. The three metrics are selected and referred to state-of-the-art comparisons. Jiang *et al.* [19] has considered IGD and HVD quality indicators to compare four DMOEAs (DNSGA-II, dCOEA, PPS, MOEA/D) with their proposed SGEA system. However, six transfer-learning methods [16] were compared using the MIGD indicator. The mathematical definition of all metrics is detailed as follows:

- The **Inverted General Distance** (IGD) is calculated using Equation (5) to measure the minimum Euclidean distance d between i^th points in the generate Pareto Optimal Front (POF) and the true POF^*.

$$IGD(POF^*, POF) = \frac{\sum_{i \in POF^*} d(i, POF)}{|POF^*|} \quad \textbf{(5)}$$

- The **Mean Inverted General Distance** (MIGD) is the average of IGD values computed using Equation (6).

$$MIGD(POF_t^*, POF_t) = \frac{1}{T} \sum_{t \in T} IGD(POF_t^*, POF_t) \quad \textbf{(6)}$$

- The **Hypervolume Difference** (HVD) indicator is designed to measure the difference between the hypervolume (HV) of the obtained POF^* to the true POF^* and presented in Equation (7).

$$HVD = HV(POF_t^*) - HV(POF_t) \quad \textbf{(7)}$$

Second, Guo *et al.* [24] considered the Robust General Distance (RGD), and the Robust Inverted General Distance (RIGD) indicators to determine the performance of prediction-based approaches for solving a set of DMOPs.

- The **Robust General Distance** (RGD) metric aims to determine the convergence of the obtained Pareto front toward the true POF based on the maximum survival time L_i and the robust i-th Pareto optimal solution. The mathematical presentation is presented in Equation (8).

$$RGD = \frac{1}{L}\sum_{i=1,\ q=k_i,\dots,k_i+L_{k_i}}^{L} \max GD(q) \tag{8}$$

- The **Robust Inverted General Distance** (RIGD) indicator is the mean value of the presented in Equation (9), and aims to calculate both convergence and distribution of the robust POS.

$$RIGD = \frac{1}{L}\sum_{i=1,\ q=k_i,\dots,k_i+L_{k_i}}^{L} \max IGD(q) \tag{9}$$

where;

- L_i is the survive time of the robust i-th Pareto optimal set (POS),
- q is the robust Pareto optimal set (POS).

5. Experimental Study: Results and Statistical Analysis

This section aims to analyse the quantitative results over IGD, MIGD, HVD, RIGD, and RGD quality indicators. The mean and standard deviation values are reported in the appendices section (see Tables 1-7). The relevant values are highlighted in bold and grey. The inferential statistical concept of Friedman's two-way Analysis of Variance (ANOVA) method, and the Wilcoxon sum ranks test are used to determine the level of significance between all tested DMOEAs.

5.1. Results Analysis through Mean and Standard Deviation

The impact of changing frequency on DMOPs was demonstrated on DMOEAs by considering different configurations based on the constant value of the severity of change (n_t) and the frequency (τ_t) variation. These configurations are implemented for severe, moderate, and slight environmental changes in accordance with the comparable systems in [16], [19], and [24].

5.1.1. Results on UDF and F(ZJZ) Problems

Experimental results in Table 1, 2 (see appendices) have shown that DPb-MOPSO outperforms DNSGA-II, dCOEA, PPS, MOEA/D, SGEA, MOPSO, dMOPSO, and pbMOPSO methods for solving the majority of UDF and F(ZJZ) functions using both IGD and HVD metrics. The 13 functions are categorized into three categories: type I (UDF1, UDF2), type II (F5-F10, UDF4, UDF5), and type III (UDF3, UDF6, UDF7). In order to show statistically reliable conclusions, the boxplots of the one-way ANOVA test have been considered in Figure 3. As compared to Dynamic-MOPSO, DPb-MOPSO is a competitive system with the best convergence on IGD and the best distribution on HVD. The best results were 8/13 (UDF1, UDF2, UDF4, UDF5, UDF6, F5, F6, F8) on IGD, and 8/13 (UDF1, UDF2, UDF3, UDF4, UDF5, UDF6, F6, F8) on HVD. In contrast, Dynamic-MOPSO has 3/13 (F7, F9, F10) on IGD and 5/13 (F5, F7, F9, F10, UDF7) on HVD. It is notable that when solving UDF7 on IGD, DPb-MOPSO differs slightly from pbMOPSO algorithm. However, it only has poor convergence compared to dMOPSO on the IGD measurement for solving UDF3 functions.

5.1.2. Results on FDA and dMOP Problems

In the case of 8 FDA and dMOP functions, all quantitative results of MIGD, IGD and HVD indicators are reported in Tables 3, 4, and 5 respectively (see the appendix section) with severe, moderate and slight changes. In order to facilitate the analysis of the results, the boxplots of the one-way ANOVA test are considered in Figures

4, 5, and 6. In this part of the study, 8 FDA and dMOP benchmarks were tested. Most FDA and dMOP functions belong to type I (FDA1, FDA4, dMOP3) or types II (FDA2, FDA3, FDA5, dMOP2), and only dMOP1 belongs to type III. The one-way ANOVA results in Figure 4 determine the importance of DPb-MOPSO compared with MMTL-MOEAD, KF-MOEAD, PPS-MOEAD, SVR-MOEAD, Tr-MOEAD, RI-MOEAD, MOPSO, dMOPSO, pbMOPSO and Dynamic-MOPSO to solve 8 DMOPs through MIGD metric. Figures 5 and 6 show the boxplots of the one-way ANOVA test of the IGD and the HVD quality indicators. Compared with MMTL-MOEA/D , KF-MOEA/D, PPS-MOEA/D, SVR-MOEA/D, Tr-MOEA/D, RI-MOEA/D, MOPSO, dMOPSO, and pbMOPSO, the DPb-MOPSO algorithm is the most effective to solve 7/8 (FDA1, FDA3, FDA4, FDA5, dMOP1, dMOP2, dMOP3), 6/8 (FDA3, FDA4, FDA5, dMOP1, dMOP2, dMOP3), and 8/8 (5 FDA, and 3 dMOP functions) using the MIGD indicator for severe, moderate and slight environmental changes respectively. Dynamic-MOPSO approach has only good results when solving 2/8 (FDA1, FDA2), 1/8 (FDA2) on the MIGD metric with moderate, and severe environmental changes respectively.

Using IGD metric, DPb-MOPSO aims to solve 7/8 (FDA2, FDA3, FDA4, FDA5, dMOP1, dMOP2, dMOP3), and MOPSO has 1/8 (FDA1) for severe changes. For moderate changes, 6/8 (FDA1, FDA3, FDA4, FDA5, dMOP1, dMOP2) functions are resolved using DPb-MOPSO, 1/8 (FDA1) uses DNSGA-II, 1/8 (dMOP3) uses dCOEA. For slight changes, 5/8 (FDA3, FDA4, FDA5, dMOP1, dMOP2) benchmarks are resolved using DPb-MOPSO, 2/8 (FDA1, FDA2) use MOPSO, and 1/8 (dMOP3) use dCOEA. Compared with MOPSO which aims to solve 1/8 (FDA1), SGEA 1/8 (FDA2), and dCOEA 1/8 (dMOP3) with moderate and slight change, the DPb-MOPSO system can solve 6/8 (FDA3, FDA4, FDA5, dMOP1, dMOP2, dMOP3), 5/8 (FDA3, FDA4, FDA5, dMOP1, dMOP2), and 5/8 (FDA3, FDA4, FDA5, dMOP1, dMOP2) functions for the severe, moderate and slight environmental changes using HVD metric. Based on the reported quantitative results in both Table 6, 7 (see appendices), it is remarkable that the proposed DPb-MOPSO algorithm performs all prediction-based approaches, namely; MOEA/D, MOEA/D-DE, SPEA2, MOPSO, and NSGAIII with three different prediction techniques (Ave, Adopt, and Weight) over RIGD and RGD for solving 5 FDA, and 3 dMOP benchmarks. Also, the boxplot in both Figures 7, and 8 details the superiority of the novel DPb-MOPSO algorithm.

5.2. Discussion through the non-parametric statistical methods

The use of the mean and standard deviation values of IGD, MIGD, HVD, RIGD, and RGD indicators are insufficient to determine the performance of swarm intelligence algorithms and highlight their significance level. For this reason, Friedman's two-way ANOVA ranking test and its post-hoc procedure [47] are used to analyze the results of different environmental changes on IGD, MIGD, HVD, RIGD, and RGD. Friedman's two-way ANOVA test is a non-parametric alternative to repeated-measures ANOVA. The test does not assume a normal distribution to compare multiple treatments and determine the level of significance between algorithms. First, define the null hypothesis $H_0: \mu_1 = \mu_i$ (there is no difference between the means), and the alternative hypothesis $H_1: \mu_1 \neq \mu_i$ (there is a difference). If the P-value $\leq \alpha$ indicates the statistical significance level, then reject the null hypothesis, otherwise H_1 is accepted, where Alpha (α) is equal to 0.05. Friedman's two-way ANOVA ranking is calculated by assigning a ranking value (r_i) to each algorithm i while; 1 is the highest result for k, and k is the worst result. For a tie case, the average of the rankings is assigned. The decision rule is defined as the critical value (CV) determined according to the chi-square (χ^2) table[1]. The CV value is fixed according to the degree of freedom (df)

[1] http://uregina.ca/~gingrich/appchi.pdf.

equal to $k-1$, where k is the number of comparison algorithms and $\alpha = 0.05$. If the calculated χ^2 is greater than CV, the null hypothesis will be rejected. In addition, if the computed P-value is less than 0.05, it assumes that there is a significant difference between the test algorithms.

5.2.1 Friedman rankings test

According to the following Tables 6 and 7, it can be concluded that DPb-MOPSO is the winner using IGD for solving FDA, and dMOP functions with severe, moderate, and slight environmental changes. However, when using the IGD, HVD, and MIGD quality indicators, it ranks higher in addressing the moderate environmental changes of 21 DMOPs (FDA, dMOP, UDF, and ZJZ (F)). Whereas, assuming that the overall statistical difference between all tested algorithms is significant, all calculated χ^2 are greater than CV.

At the 0.05 level of significance, all means are not equal. Almost, most p-values less than 0.05 represent the assumption of rejecting the null hypothesis and assuming that some means are not exactly equal and there are significant differences. Friedman's ranking test detected a significant difference between multiple comparisons. However, it is difficult to determine which group pairs are significantly different. Therefore, Friedman 1 × N ANOVA multiple comparisons were considered for the results discussion, aiming to compare DPb-MOPSO as a control method with other DMOEAs.

In addition, Table 8 presents the Friedman's ANOVA mean ranks of the proposed DPb-MOPSO algorithm compared with different prediction-based approaches. It is remarkable that the novel DPb-MOPSO obtain the best mean ranks for solving 5 FDA and 3 dMOP functions over both RIGD, and RGD quality indicators with a p-values less than 0.05 significance level.

Table 6. Friedman's ANOVA Mean Ranks on IGD and HVD for FDA, dMOP, F (ZJZ) and UDF.

Mean Ranks on IGD and HVD						
Dynamic changes	Severe (τ_t=5, n_t=10)		Moderate (τ_t=10, n_t=10)		Slight (τ_t=20,n_t=10)	
DMOPs	FDA, dMOP		FDA, dMOP, F(ZJZ), UDF		FDA, dMOP	
QI	**IGD**	**HVD**	**IGD**	**HVD**	**IGD**	**HVD**
DNSGA-II	8.94	7.25	7.33	6.45	8.38	6.25
dCOEA	6.56	5.63	8.02	6.57	7.00	6.50
PPS	8.00	7.25	7.24	6.17	8.13	6.25
MOEA/D	8.38	7.75	7.79	6.83	7.38	7.00
SGEA	6.13	4.50	6.14	4.55	5.25	4.50
MOPSO	3.75	6.00	5.45	7.43	3.31	6.06
dMOPSO	5.50	6.50	6.36	7.55	5.19	6.81
pbMOPSO	2.88	4.38	3.24	4.57	5.25	5.25
Dynamic-MOPSO	3.25	3.88	1.76	2.88	3.25	4.25
DPb-MOPSO	**1.63**	**1.88**	**1.67**	**2.00**	**1.88**	**2.13**
Chi-Square χ^2	50.55	26.62	121.59	75.98	38.55	17.84
P-value ($\alpha = 0.05$)	**8.47E-8**	**2E-3**	**6.29E-22**	**1.01E-12**	**14E-5**	**37E-2**
Statistically Significant	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**yes**
Critical value	**16.92**					
df	**9**					

Table 7. Friedman's ANOVA Mean Ranks on MIGD for FDA, dMOP.

Mean Ranks on MIGD			
DMOPs	FDA, dMOP		
Dynamic changes	Severe (τ_t=5, n_t=10)	Moderate (τ_t=10, n_t=10)	Slight (τ_t=20, n_t=10)
MMTL-MOEA/D	5.88	5.75	5.75
KF-MOEA/D	9.25	8.63	8.13
PPS-MOEA/D	8.13	7.50	8.38
SVR-MOEA/D	8.63	8.13	8.38
Tr-MOEA/D	9.13	8.88	9.13
RI-MOEA/D	9.13	9.25	8.63
MOPSO	3.94	4.50	4.44
dMOPSO	4.94	6.38	5.31
pbMOPSO	4.00	3.88	4.88
Dynamic-MOPSO	1.75	1.63	1.88
DPb-MOPSO	**1.25**	**1.50**	**1.13**
Chi-Square χ^2	66.60	57.34	56.38
P-value $(\alpha = 0.05)$	**2.0E-10**	**1.15E-8**	**1.74E-8**
Statistically Significant	**Yes**	**Yes**	**Yes**
Critical value	18.31		
df	10		

Table 8. Friedman's ANOVA Mean Ranks on RIGD, and RGD for FDA, dMOP Functions

Mean Ranks RIGD and RGD						
DMOPs	**FDA, dMOP Functions**					
Prediction methods	**Ave**		**Adapt**		**Weight**	
QIs	**RIGD**	**RGD**	**RIGD**	**RGD**	**RIGD**	**RGD**
MOEA/D	3.13	3.50	2.38	2.63	2.38	2.63
MOEA/D-DE	4.88	5.00	3.88	4.88	4.75	4.63
SPEA2	3.19	3.00	3.94	3.75	3.44	4.00
MOPSO	3.56	3.38	4.19	3.75	4.19	3.75
NSGA-III	5.23	5.13	5.63	5.00	5.25	5.00
DPb-MOPSO	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
Chi-Square χ^2	26.25	26.25	29.44	25.97	28.94	24.96
P-value $(\alpha = 0.05)$	**80E-5**	**79E-5**	**19E-5**	**90E-5**	**24E-6**	**142E-5**
Statistically Significant	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**
Critical value	11.070					
df	5					

5.2.2 Friedman post-hoc Procedure 1 × N ANOVA Test

The Freidman post-hoc procedure was performed to compare group means, where DPb-MOPSO was the control method. The Freidman post-hoc procedure process results in a p-value (PV) at the 0.05 significance level that determines the degree of rejection of the null hypothesis. The obtained p-values on MIGD, IGD and HVD are reported in Table 9. Note that the sign (+) means that DPb-MOPSO is the best algorithm, and (≅) the same significant level is determined. Let's consider the MIGD indicator, compared with KF-MOEA/D, Tr-MOEA/D, RI-MOEA/D, SVR-MOEA/D, PPS-MOEA/D and MMTL-MOEA/D, the importance of the DPb-MOPSO system can be observed for all environmental changes with P-values are less than 0.05. Also, DPb-MOPSO has the same significance as dMOPSO for solving different DMOPs with moderate and slight change, and MOPSO, pbMOPSO with slight dynamic change. However, it has the same level of significance compared with Dynamic-MOPSO for all types of environmental change, and MOPSO, pbMOPSO, and Dynamic-MOPSO for both severe and moderate changes. In addition, the null hypothesis is only retained for dMOPSO system on moderate changes, and compared with DPb-MOPSO, it has the same significance only for severe changes. The PV values over IGD present the significant difference of DPb-MOPSO compared to DNSGA-II, MOEA/D, PPS, dCOEA, SGEA and dMOPSO for all the types of changes. In addition, it can be inferred that there is no difference in p-value greater than 0.05, for severe and slight changes between DPb-MOPSO compared with MOPSO, pbMOPSO and Dynamic-MOPSO. Furthermore, the computed P-values on HVD metric are greater than 0.05, resulting to retain the null hypothesis of no difference between mean values for the Dynamic-MOPSO for solving DMOPs with all types of dynamic change. However, the DPb-MOPSO system performs six DMOEAs (DNSGA-II, MOEA/D, PPS, dCOEA, dMOPSO, MOPSO) in solving DMOPs with different dynamic change except for the SGEA with severe and slight change, and pbMOPSO for the severe change which achieve the same importance as the proposal.

Table 9. P-values obtained by the post hoc method over the results of Friedman procedure with α = 0.05 on MIGD, IGD and HVD for severe, moderate and slight dynamic change.

QI	DPb-MOPSO vs.	(τ_t, n_t)		
		(5, 10)	**(10, 10)**	**(20, 10)**
MIGD	KF-MOEA/D	**≤0.001+**	**≤0.001+**	**0.001+**
	Tr-MOEA/D	**≤0.001+**	**≤0.001+**	**≤0.001+**
	RI-MOEA/D	**≤0.001+**	**≤0.001+**	**≤0.001+**
	SVR-MOEA/D	**≤0.001+**	**≤0.001+**	**0.001 +**
	PPS-MOEA/D	**0.001 +**	**≤0.001+**	**0.001 +**
	MMTL-MOEA/D	**0.005 +**	**0.010 +**	**0.005 +**
	dMOPSO	0.026 ≅	**0.003 +**	**0.012 +**
	MOPSO	0.105 ≅	0.070 ≅	**0.046 +**
	pbMOPSO	0.097 ≅	0.152 ≅	**0.024 +**
	Dynamic-MOPSO	0.763≅	0.940 ≅	0.651 ≅
IGD	DNSGA-II	**≤0.001 +**	**≤0.001 +**	**≤0.001 +**
	MOEA/D	**≤0.001 +**	**≤0.001 +**	**≤0.001 +**
	PPS	**≤0.001 +**	**≤0.001 +**	**≤0.001 +**
	dCOEA	**0.001 +**	**≤0.001 +**	**0.001 +**
	SGEA	**0.003 +**	**≤0.001 +**	**0.026 +**
	dMOPSO	**0.010 +**	**≤0.001 +**	**0.029 +**
	MOPSO	0.160 ≅	**≤0.001 +**	0.342 ≅
	pbMOPSO	0.409 ≅	0.093 ≅	**0.026 +**
	Dynamic-MOPSO	0.283 ≅	0.919 ≅	0.364 ≅
HVD	DNSGA-II	**≤0.001 +**	**≤0.001 +**	**0.006+**
	MOEA/D	**≤0.001 +**	**≤0.001 +**	**0.001+**
	PPS	**≤0.001 +**	**≤0.001 +**	**0.006+**
	dCOEA	**0.013 +**	**≤0.001 +**	**0.004+**
	SGEA	0.083 ≅	**0.006 +**	0.117≅
	dMOPSO	**0.002 +**	**≤0.001 +**	**0.002+**
	MOPSO	**0.006 +**	**≤0.001 +**	**0.009+**
	pbMOPSO	0.099 ≅	**0.006 +**	**0.039+**
	Dynamic-MOPSO	0.186 ≅	0.346 ≅	0.160≅

+: DPb-MOPSO is the best algorithm, ≅: DPb-MOPSO has the same significant level compared to other DMOEAs (significance level is 0.05).

In conclusion, and in regards to the state-of-the-art methods, the proposed DPb-MOPSO has shown its importance compared with other MOPSO-based methods. In addition, compared with the new DPb-MOPSO, the MOPSO-based system is the most competitive. But they failed to resolve multiple DMOPs (UDF3, UFD7, dMOP3, FDA1, FDA2), thus demonstrating the utility and importance of using dynamic processing strategies for two-stage optimization to detect and effectively react to changes. However, the DPb-MOPSO algorithm is failed in some test problems with time-varying spread, dynamic density of the solution, shifting of the center point and a discontinuous POF. The DPb-MOPSO may require an additional mechanism to detect the dynamic density of the POS and effectively respond to it and assume good distribution using the HVD measurement.

5.2.3 The Wilcoxon Sum Rank Test

In this sub-section, comparative result analysis is conducted for the experimental study of DMOPs using the non-parametric statistical test. The Wilcoxon sign rank test [48], [49] is used to determine the best method among all compared approaches. The performance of the proposed DPb-MOPSO algorithm is compared with different DMOEAs when used for solving different types of DMOPs using HVD, IGD, MIGD, RIGD, and RGD metrics, and all results are reported in Tables 10-13 respectively. This test is conducted based on an experimental statistical analysis over 21 DMOPs (5 FDA, 3 dMOP, 7 UDF, and 6 F(ZJZ)), with an average value for 30 independent runs for IGD, HVD and MIGD quality indicators, and 20 independent runs for RIGD and RGD metrics for each approach. The Wilcoxon ranks, is used to calculate the results variants of the proposed DPb-MOPSO compared with 18 DMOEAs. The results of Tables 10-13 showed that the novel DPb-MOPSO is the best with a p-value greater than 0.05 significant level for the majority of the tested DMOPs. However, the difference between means values of the proposed algorithm are not statistically significant compared with DNSGA-II, MOEA/D, PPS, dCOEA, and SGEA for severe, and slight dynamic change for solving FDA, and dMOP functions over the HVD metric, and the Dynamic-MOPSO with slight change over HVD, IGD metrics.

Table 10. Results of the Wilcoxon Sum Rank Test over HVD metric for Severe, Moderate and Slight Dynamic Change.

Type of dynamic change	DMOPs	DPb-MOPSO vs.	R-	R+	p-value	Best method
Severe change $(\tau_t = 5, n_t$=10)	8 benchmarks (FDA, dMOP)	DNSGA-II	29	7	0.123485	≅
		MOEA/D	30	6	0.092892	≅
		PPS	29	7	0.123485	≅
		dCOEA	29	7	0.123485	≅
		SGEA	29	7	0.123485	≅
		dMOPSO	34	2	0.025062	**DPb-MOPSO**
		MOPSO	35	1	0.017290	**DPb-MOPSO**
		pbMOPSO	36	0	0.011719	**DPb-MOPSO**
		Dynamic-MOPSO	36	0	0.011719	**DPb-MOPSO**
Moderate change $(\tau_t = 10, n_t$=10)	21 benchmarks (FDA, dMOP, UDF, F)	DNSGA-II	198	33	0.004137	**DPb-MOPSO**
		MOEA/D	198	33	0.004137	**DPb-MOPSO**
		PPS	195	36	0.005723	**DPb-MOPSO**
		dCOEA	202	29	0.002642	**DPb-MOPSO**
		SGEA	192	39	0.007838	**DPb-MOPSO**
		dMOPSO	229	2	0.000080	**DPb-MOPSO**
		MOPSO	229	2	0.000080	**DPb-MOPSO**
		pbMOPSO	222	9	0.000214	**DPb-MOPSO**
		Dynamic-MOPSO	126	105	0.715145	≅
Slight change $(\tau_t = 20, n_t$=10)	8 benchmarks (FDA, dMOP)	DNSGA-II	28	8	0.161429	≅
		MOEA/D	28	8	0.161429	≅
		PPS	28	8	0.161429	≅
		dCOEA	25	11	0.326989	≅
		SGEA	28	8	0.161429	≅
		dMOPSO	35	1	0.017290	**DPb-MOPSO**
		MOPSO	35	1	0.017290	**DPb-MOPSO**
		pbMOPSO	36	0	0.011719	**DPb-MOPSO**
		Dynamic-MOPSO	29	7	0.123485	≅

≅ indicate that the DPb-MOPSO algorithm has the same significant level compared to other DMOEAs (significance level is 0.05).

Table 11. Results of the Wilcoxon Sum Rank Test for DPb-MOPSO versus DMOEAs over IGD metric for Severe, Moderate and Slight Dynamic Change

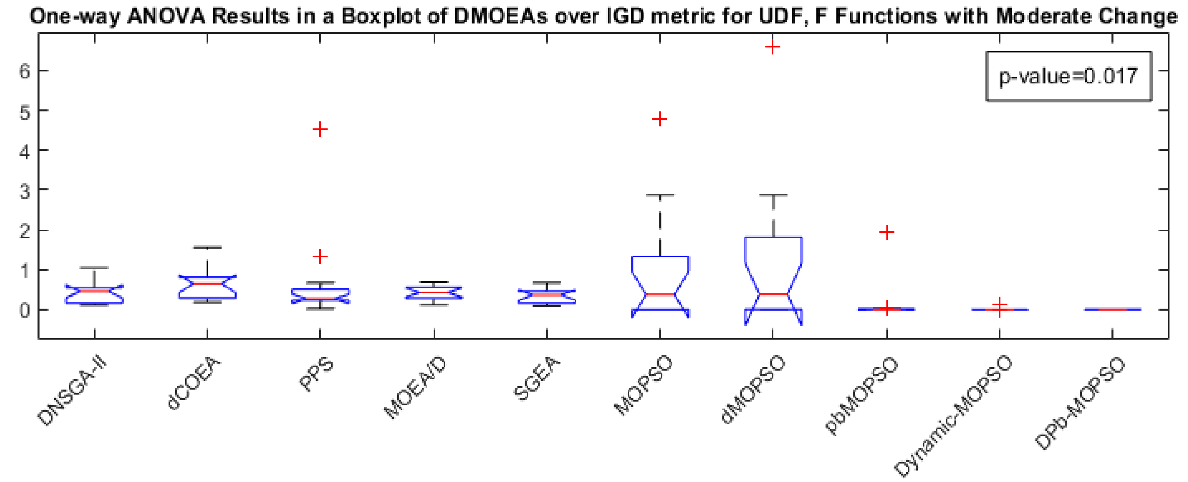
Type of dynamic change	DMOPs	DPb-MOPSO vs.	R-	R+	p-value	Best method
Severe change ($\tau_t = 5, n_t$=10)	8 benchmarks (FDA, dMOP)	DNSGA-II	36	0	0.011719	**DPb-MOPSO**
		MOEA/D	36	0	0.011719	**DPb-MOPSO**
		PPS	36	0	0.011719	**DPb-MOPSO**
		dCOEA	36	0	0.011719	**DPb-MOPSO**
		SGEA	36	0	0.011719	**DPb-MOPSO**
		dMOPSO	35	1	0.017290	**DPb-MOPSO**
		MOPSO	32	4	0.049950	**DPb-MOPSO**
		pbMOPSO	32	4	0.049950	**DPb-MOPSO**
		Dynamic-MOPSO	32	4	0.049950	**DPb-MOPSO**
Moderate change ($\tau_t = 10, n_t$=10)	21 benchmarks (FDA, dMOP, UDF, F)	DNSGA-II	231	0	0.000060	**DPb-MOPSO**
		MOEA/D	231	0	0.000060	**DPb-MOPSO**
		PPS	231	0	0.000060	**DPb-MOPSO**
		dCOEA	231	0	0.000060	**DPb-MOPSO**
		SGEA	231	0	0.000060	**DPb-MOPSO**
		dMOPSO	228	3	0.000092	**DPb-MOPSO**
		DNSGA-II	228	3	0.000092	**DPb-MOPSO**
		MOEA/D	220	11	0.000281	**DPb-MOPSO**
		Dynamic-MOPSO	116	115	0.986134	≅
Slight change ($\tau_t = 20, n_t$=10)	8 benchmarks (FDA, dMOP)	DNSGA-II	36	0	0.011719	**DPb-MOPSO**
		MOEA/D	35	1	0.017290	**DPb-MOPSO**
		PPS	36	0	0.011719	**DPb-MOPSO**
		dCOEA	31	5	0.068704	≅
		SGEA	36	0	0.011719	**DPb-MOPSO**
		dMOPSO	35	1	0.017290	**DPb-MOPSO**
		MOPSO	32	4	0.049950	**DPb-MOPSO**
		pbMOPSO	36	0	0.011719	**DPb-MOPSO**
		Dynamic-MOPSO	29	7	0.123485	≅

Table 12. Results of the Wilcoxon Sum Rank Test for DPb-MOPSO versus DMOEAs over MIGD metric for Severe, Moderate and Slight Dynamic Change

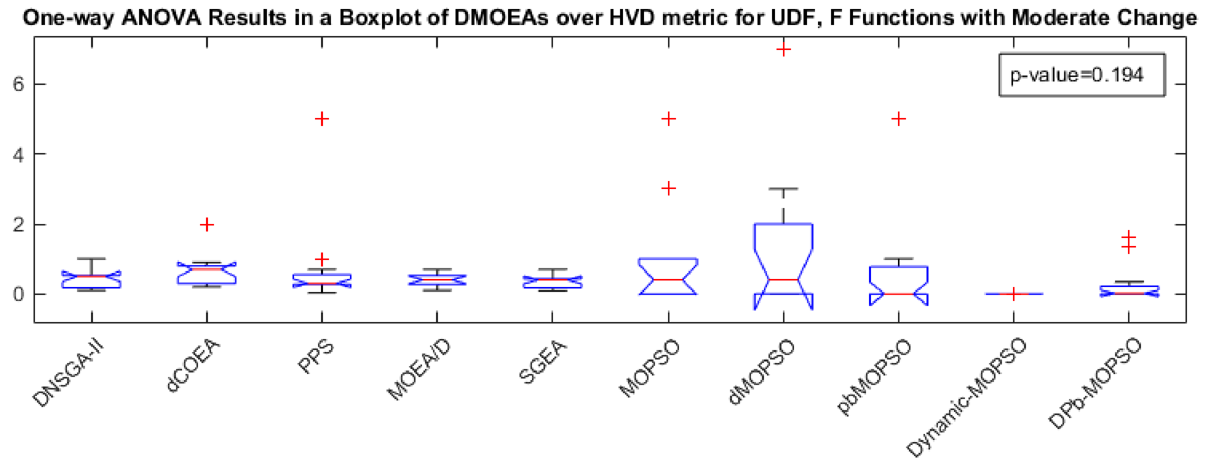
Type of dynamic change	DMOPs	DPb-MOPSO vs.	R-	R+	p-value	Best method
Severe change ($\tau_t = 5, n_t$=10)	8 benchmarks (FDA, dMOP)	MMTL-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		KF-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		PPS-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		SVR-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		Tr-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		RI-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		MOPSO	36	0	0.011719	**DPb-MOPSO**
		dMOPSO	36	0	0.011719	**DPb-MOPSO**
		pbMOPSO	36	0	0.011719	**DPb-MOPSO**
		Dynamic-MOPSO	31	5	0.068704	≅
Moderate change ($\tau_t = 10, n_t$=10)	8 benchmarks (FDA, dMOP))	MMTL-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		KF-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		PPS-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		SVR-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		Tr-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		RI-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		MOPSO	34	2	0.025062	**DPb-MOPSO**
		dMOPSO	36	0	0.011719	**DPb-MOPSO**
		pbMOPSO	36	0	0.011719	**DPb-MOPSO**
		Dynamic-MOPSO	19	17	0.888638	≅
Slight change ($\tau_t = 20, n_t$=10)	8 benchmarks (FDA, dMOP)	MMTL-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		KF-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		PPS-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		SVR-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		Tr-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		RI-MOEA/D	36	0	0.011719	**DPb-MOPSO**
		MOPSO	36	0	0.011719	**DPb-MOPSO**
		dMOPSO	36	0	0.011719	**DPb-MOPSO**
		pbMOPSO	36	0	0.011719	**DPb-MOPSO**
		Dynamic-MOPSO	34	2	0.025062	**DPb-MOPSO**

Table 13. Results of the Wilcoxon Sum Rank Test for DPb-MOPSO versus DMOEAs over RIGD and RGD metrics for Moderate Change

DMOPs	QIs	DPb-MOPSO vs.	R-	R+	p-value	Best method
8 benchmarks (FDA, dMOP))	**RIGD**	MOEA/D	300	0	0.000018	**DPb-MOPSO**
		MOEA/D-DE	300	0	0.000018	**DPb-MOPSO**
		SPEA2	300	0	0.000018	**DPb-MOPSO**
		MOPSO	300	0	0.000018	**DPb-MOPSO**
		NSGAIII	300	0	0.000018	**DPb-MOPSO**
	RGD	MOEA/D	300	0	0.000018	**DPb-MOPSO**
		MOEA/D-DE	300	0	0.000018	**DPb-MOPSO**
		SPEA2	300	0	0.000018	**DPb-MOPSO**
		MOPSO	300	0	0.000018	**DPb-MOPSO**
		NSGAIII	300	0	0.000018	**DPb-MOPSO**

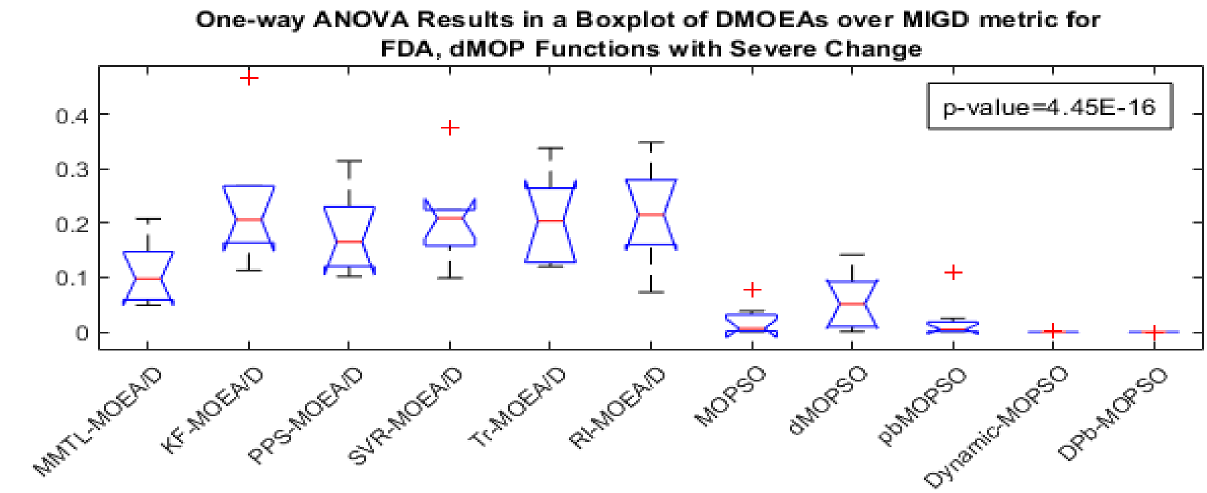


(a)

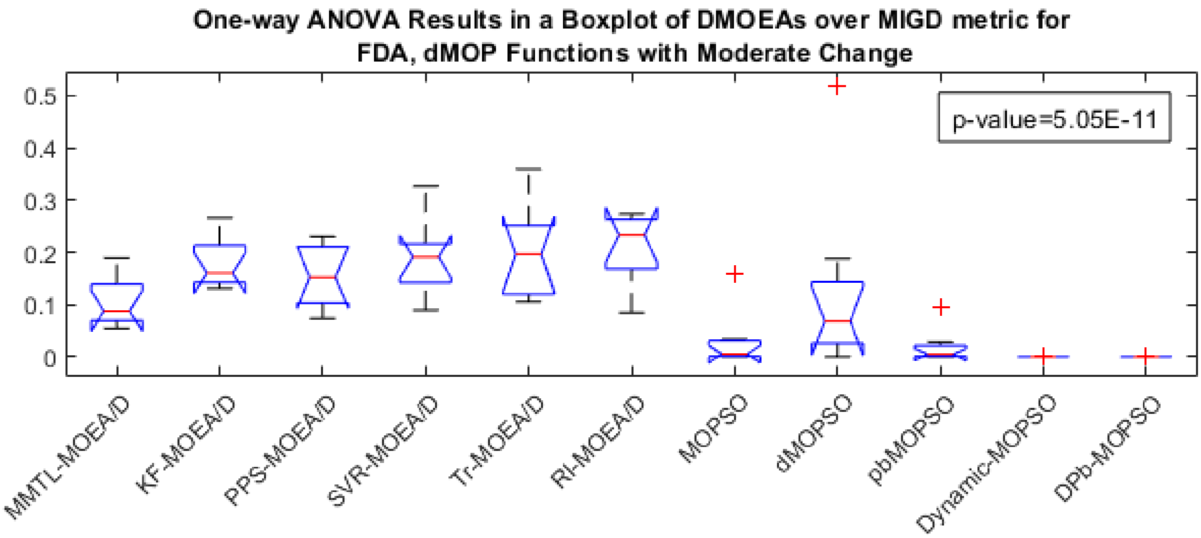


(b)

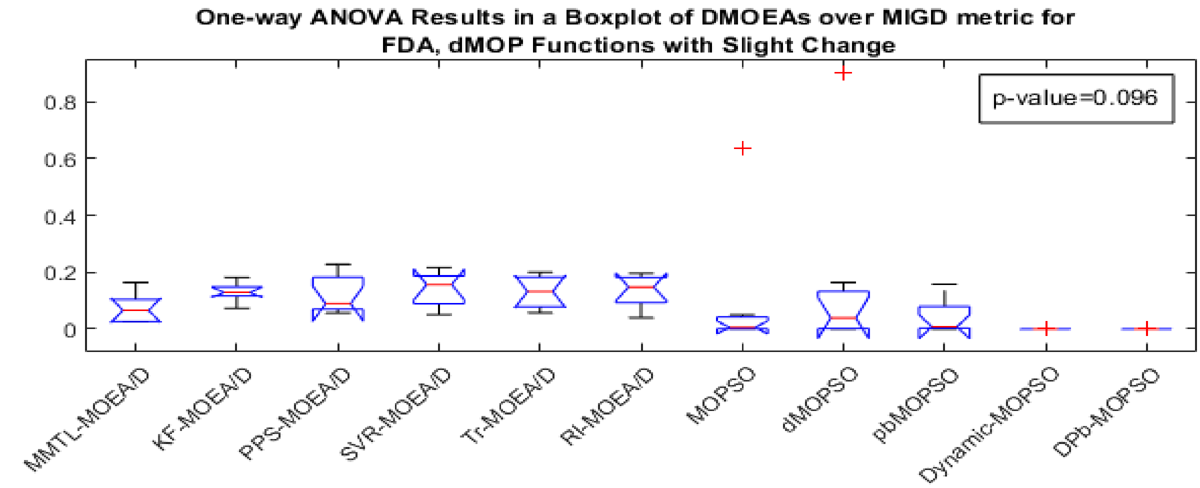
Fig. 3. ANOVA Boxplot on (a) IGD and (b) HVD with Moderate Environmental Changes for UDF and F(ZJZ) functions.



(a)



(b)



(c)

Fig. 4. ANOVA Multiple Comparison on **MIGD** with (a) Severe, (b) Moderate, and (c) Slight Environmental Changes for FDA and dMOP functions.

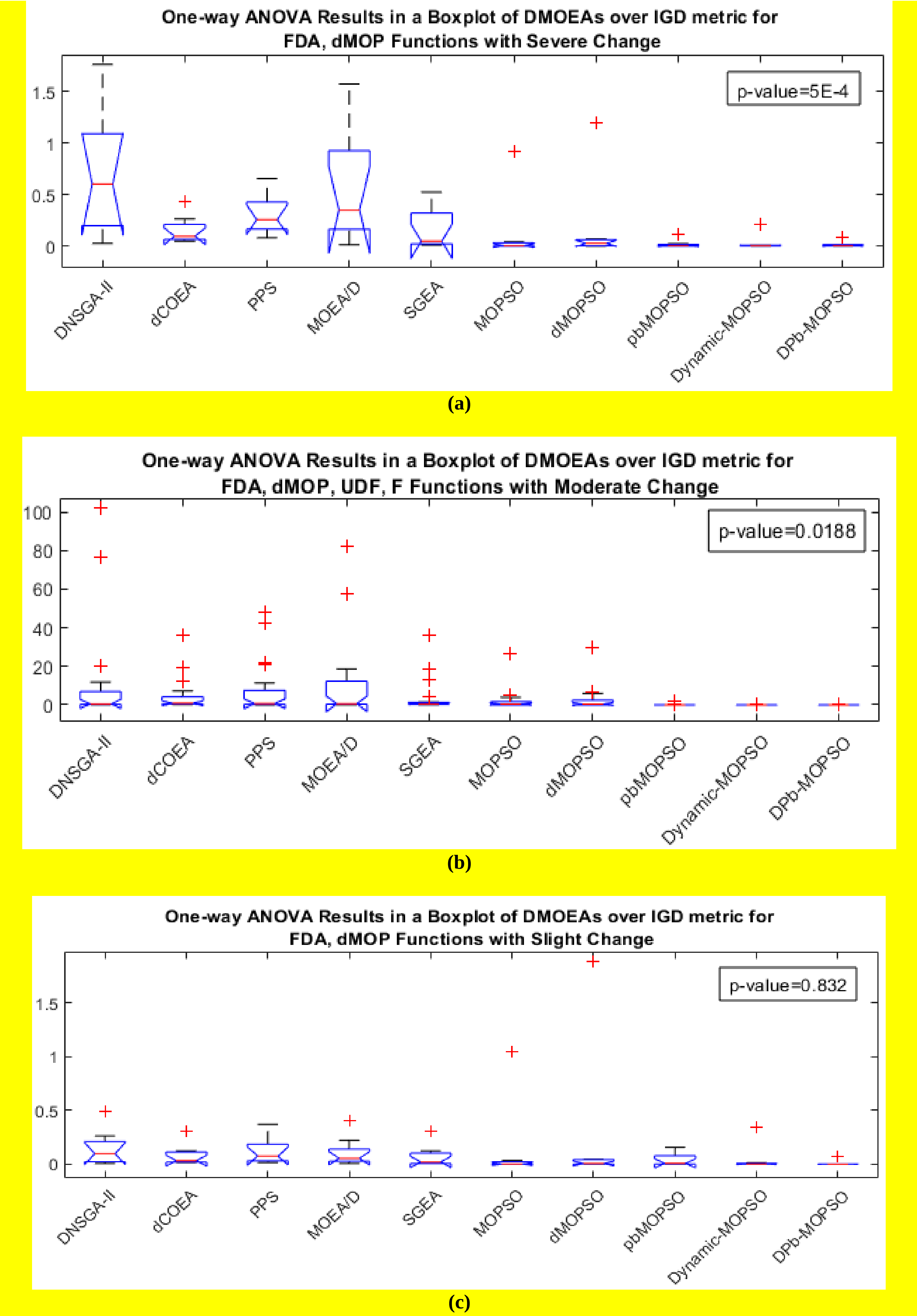
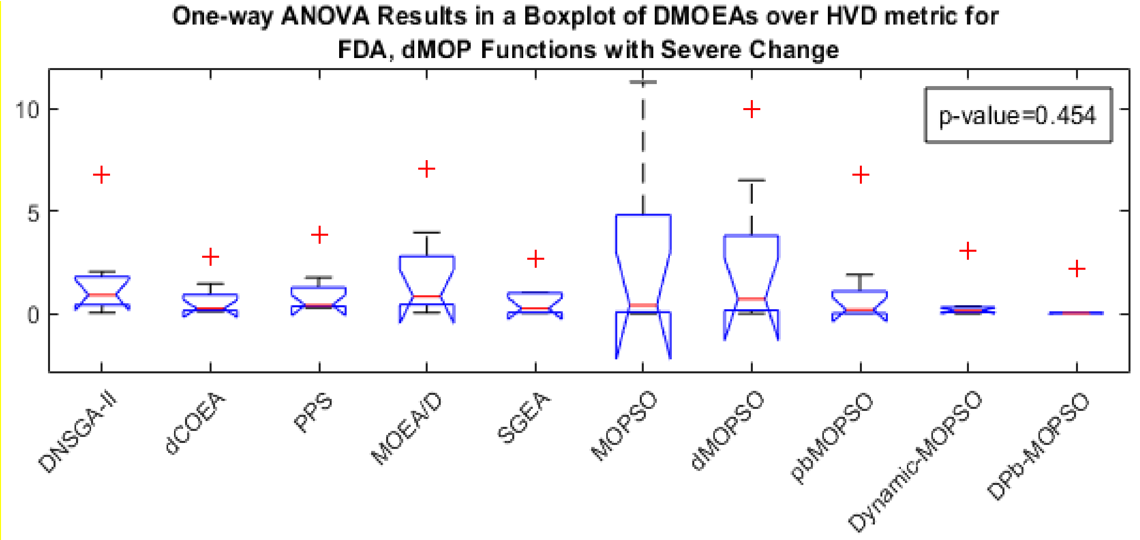
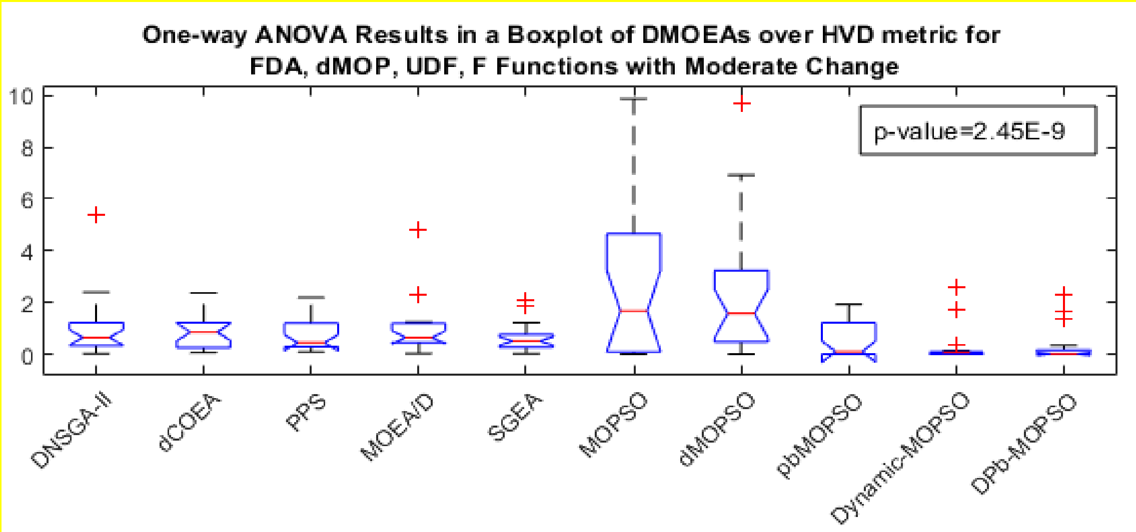


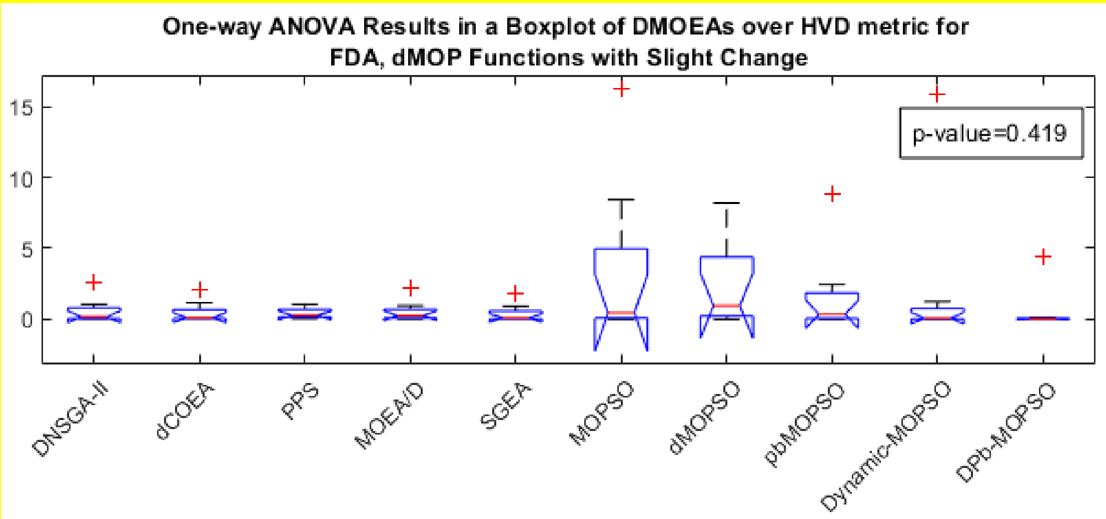
Fig. 5. ANOVA Multiple Comparison on IGD with (a) Severe, (b) Moderate, and (c) Slight Environmental Changes for FDA, dMOP, UDF, and F functions



(a)

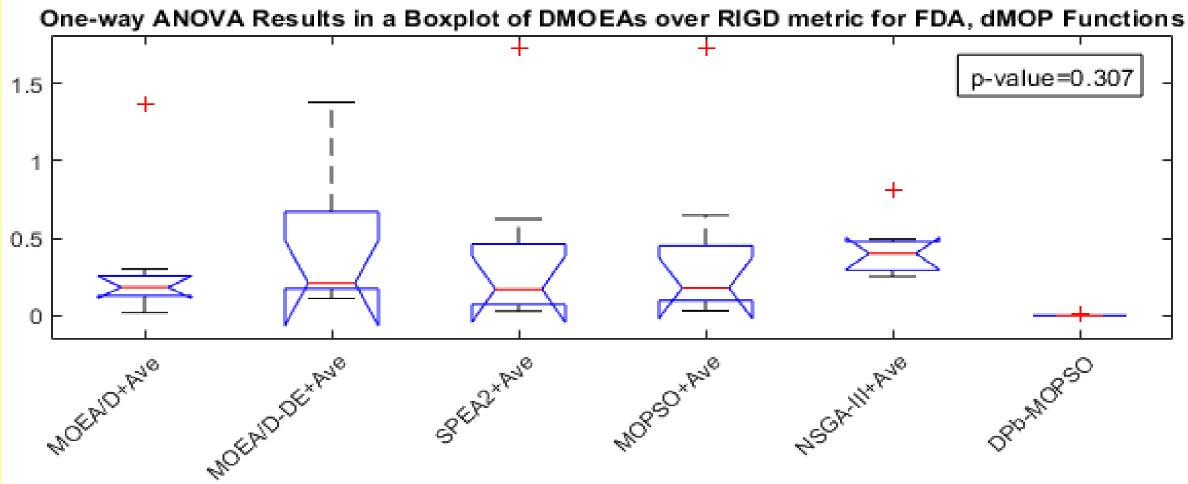


(b)

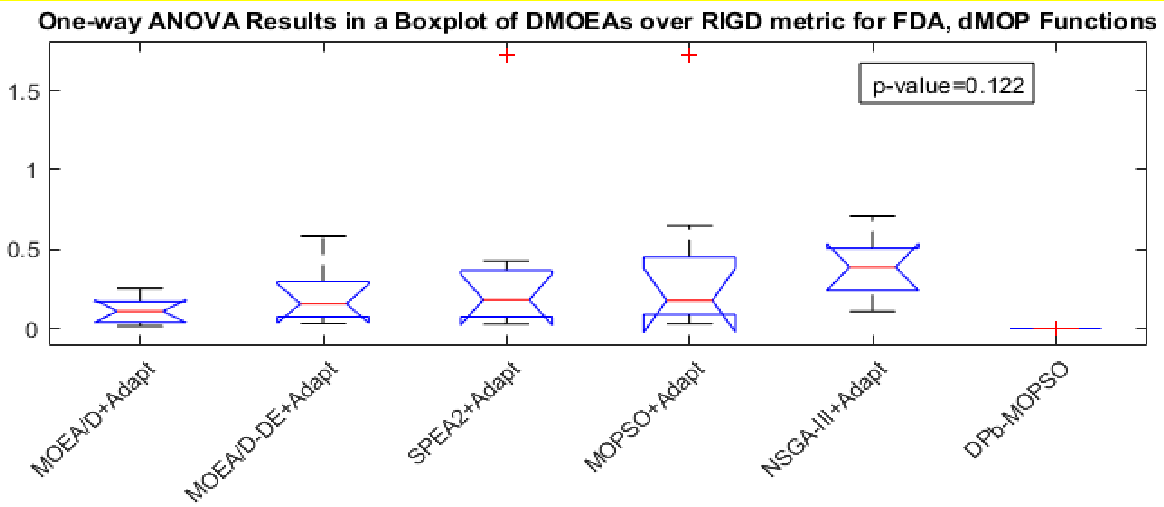


(c)

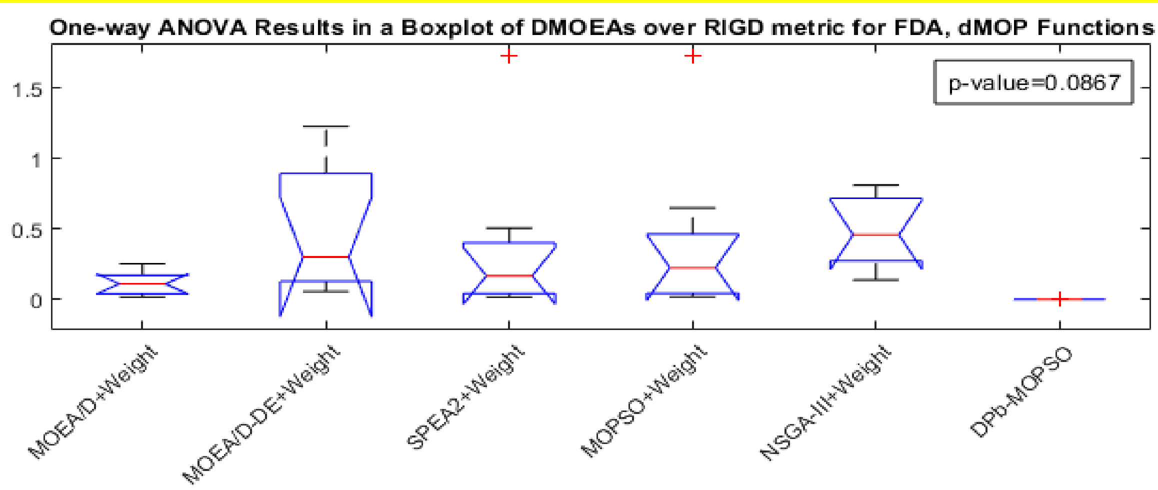
Fig. 6. ANOVA Multiple Comparison on HVD with (a) Severe, (b) Moderate, and (c) Slight Environmental Changes for FDA, dMOP, UDF, F functions.



(a)

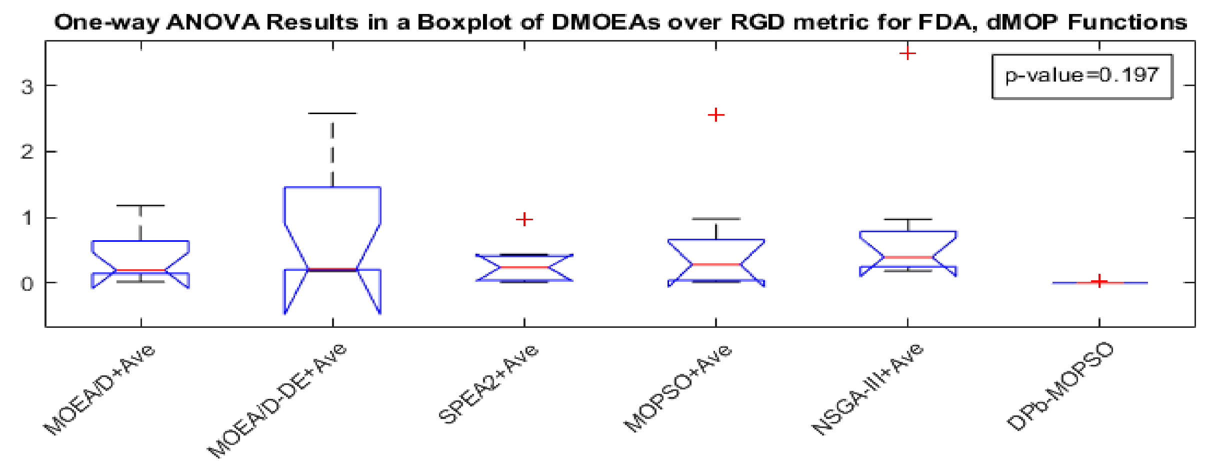


(b)

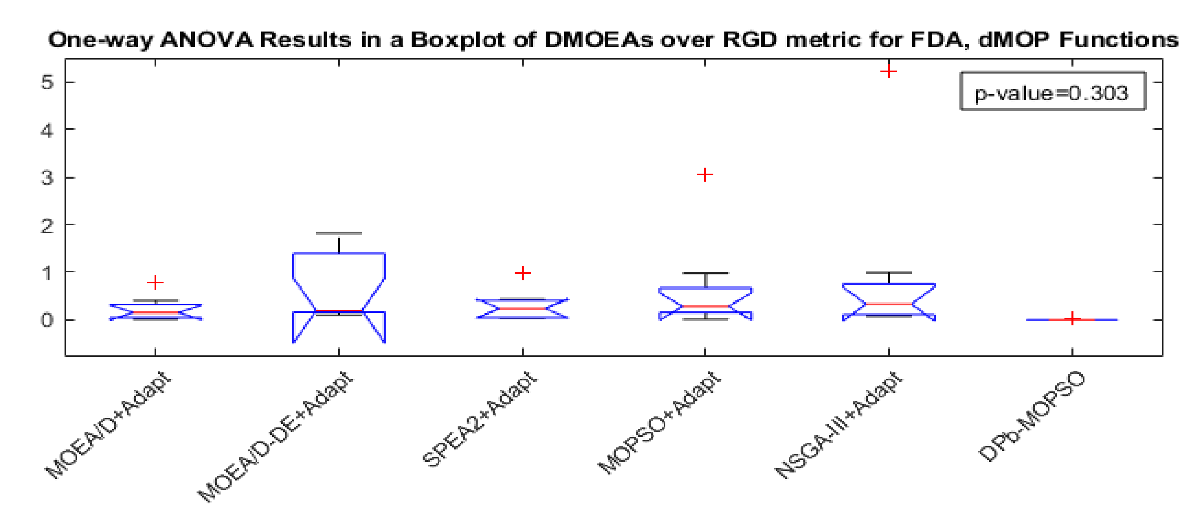


(c)

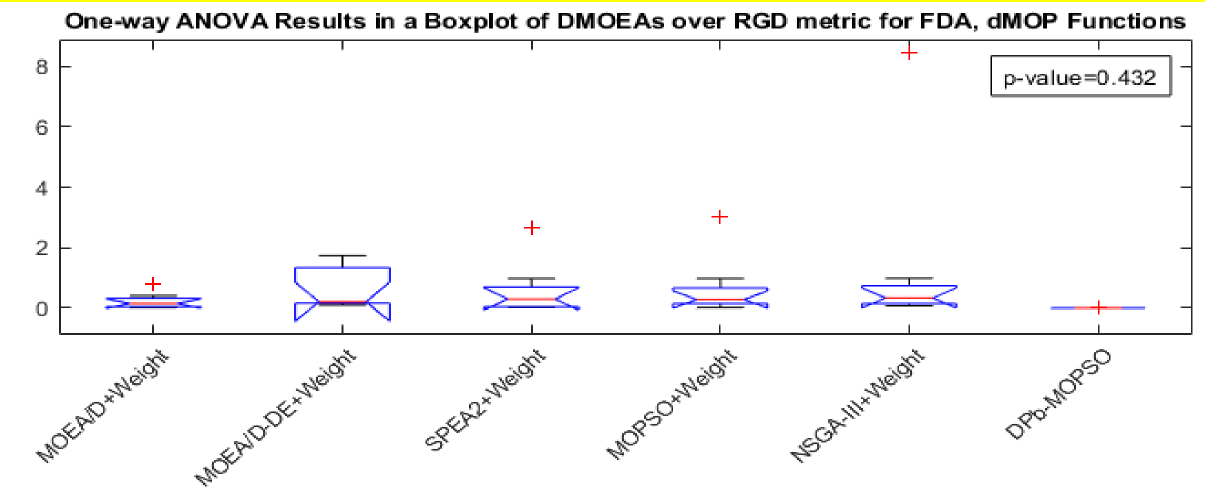
Fig. 7. ANOVA Multiple Comparison of DPb-MOPSO versus DMOEAs with (a) Ave, (b) Adapt, (c) Weight predictions methods on RIGD for FDA, dMOP functions.



(a)



(b)



(c)

Fig. 8. ANOVA Multiple Comparison of DPb-MOPSO versus DMOEAs with (a) Ave, (b) Adapt, (c) Weight predictions methods on RGD for FDA, dMOP functions.

6. Stability Analysis using Lyapunov Theorem

The stability analysis is an important aspect of analyzing the robustness of dynamic systems. The Lyapunov theorem proposed by Alexandr Mikhailovich Lyapunov [50] is a well-known stability analysis mechanism. The theorem aims to measure the growth of the initial values over time, and the small differences from one instance to another is called the Lyapunov Exponent (LE). In general, an ordinary differential system is written in the form of the differential Equation (10).

$$\frac{dx}{dt} = F(x,t) \tag{10}$$

where; x is the state of the system, and the time derivative of x is presented in Equation (11).

$$\frac{dx}{dt} = \lim_{\Delta t \to 0} \frac{x(t+\Delta t) - x(t)}{\Delta t} \tag{11}$$

A solution with initial value x_0 at the initial time t_0 is zero stable, if $x(0,t) = 0$. A solution is asymptotically stable, if $x(t,t0,x0) \ \to \ 0$ when $t \ \to \ +\infty$ and unstable otherwise.

Different modeling and analysis of complex systems has been published by Hiroki Sayama in [51], when the Lyapunov Exponent has been introduced as an analytical metric to measure chaos of the system. The LE can grow to very large differences to indicate the speed at which two initially close dynamics diverge or converge, and presented in Equation (12).

$$|F^t(x_0 + \varepsilon) - F^t(x_0)| \approx \varepsilon e^{\lambda t} \tag{12}$$

$$e^{\lambda t} \approx \frac{|F^t(x_0+\varepsilon) - F^t(x_0)|}{\varepsilon} \tag{13}$$

$$\lambda = \lim_{t\to\infty, \varepsilon \to 0} \frac{1}{t} \log \frac{|F^t(x_0+\varepsilon) - F^t(x_0)|}{\varepsilon} = \lim_{t\to\infty} \frac{1}{t} \sum_{i=0}^{t-1} log \left| \frac{dF}{dx} |x = x_i \right| \tag{14}$$

where;

- $\varepsilon e^{\lambda t}$ is the assumption that the distance $|F^t(x_0 + \varepsilon) - F^t(x_0)|$ between two points grow exponentially over time $t \ \forall \ t \to \infty$, and presented in Equation (13),
- λ is the Lyapunov exponent, and $\varepsilon > 0$, and presented in Equation (14),
- if $\lambda \ > \ 0$, small distances grow indefinitely over time,
- if $\lambda \ < \ 0$, small distances don't grow indefinitely.
- A positive value of LE indicates divergence, while a negative value indicates convergence in phase space.

Referring to papers [52] [53], the LE value has been estimated over the calculation of the quantity $\langle ln[d(k)] \rangle$, to measure the exponential growth of $\langle d(k) \rangle$ with increasing k, where $d(k)$ is the difference between the two sequences after k steps is calculated using the following formula: $d(k) = |t_{i+k} - t_{j+k}|$, and the average $\langle \cdots \rangle$ over pairs points t_i and t_j with $d(0) = |t_i - t_j| < n$, n is a small number as shown in the following Equations (15) and (16).

$$d(0) = |t_i - t_j| + |t_{i+1} - t_{j+1}| < n \tag{15}$$

$$d(k) = |t_{i+k} - t_{j+k}| + |t_{i+k+1} - t_{j+k+1}| \tag{16}$$

If the system is chaotic, $d(k)$ will rise exponentially with k, and the distance between two points t_i and t_j is denoted by $t_i, t_{i+1}, \dots, t_{i+k}$ and $t_j, t_{j+1}, \dots, t_{j+k}$ grows over time. The convergence and divergence of dynamic system is measured using the LE value. Therefore, $ln\, d(k)$ is computed and plotted to estimate the Lyapunov Exponent (LE). When the value of LE is very small and close over time, this state indicates that the system is stable and reaches the same solution over time. In this study, a set of IGD values of 30 independent runs are considered to measure the stability of the DPb-MOPSO algorithm for solving a set of minimization DMOPs with moderate change, where n_t and τ_t are equal to 10. The independent IGD values are calculated using the smallest Euclidean distance d_i between ith points in the best obtained POF and the true POF^* over time t. The Lyapunov Theorem is considered to control the behavior of the DPb-MOPSO algorithm in terms of convergence and diversity. The stability analysis through Lyapunov Exponent is computed using $\ln(IGD())$. Figure 9 shows the spectrum of the Lyapunov Exponents of F, UDF, FDA, and dMOP functions. Moreover, it can be concluded that the Lyapunov exponents for all tested problems are under zero with negative values assuming that the system converges over time. The spectrum of the Lyapunov exponent indicates the stability of DPb-MOPSO with negative values of LE. All location points determine the fast convergence of DPb-MOPSO to all tested problems.

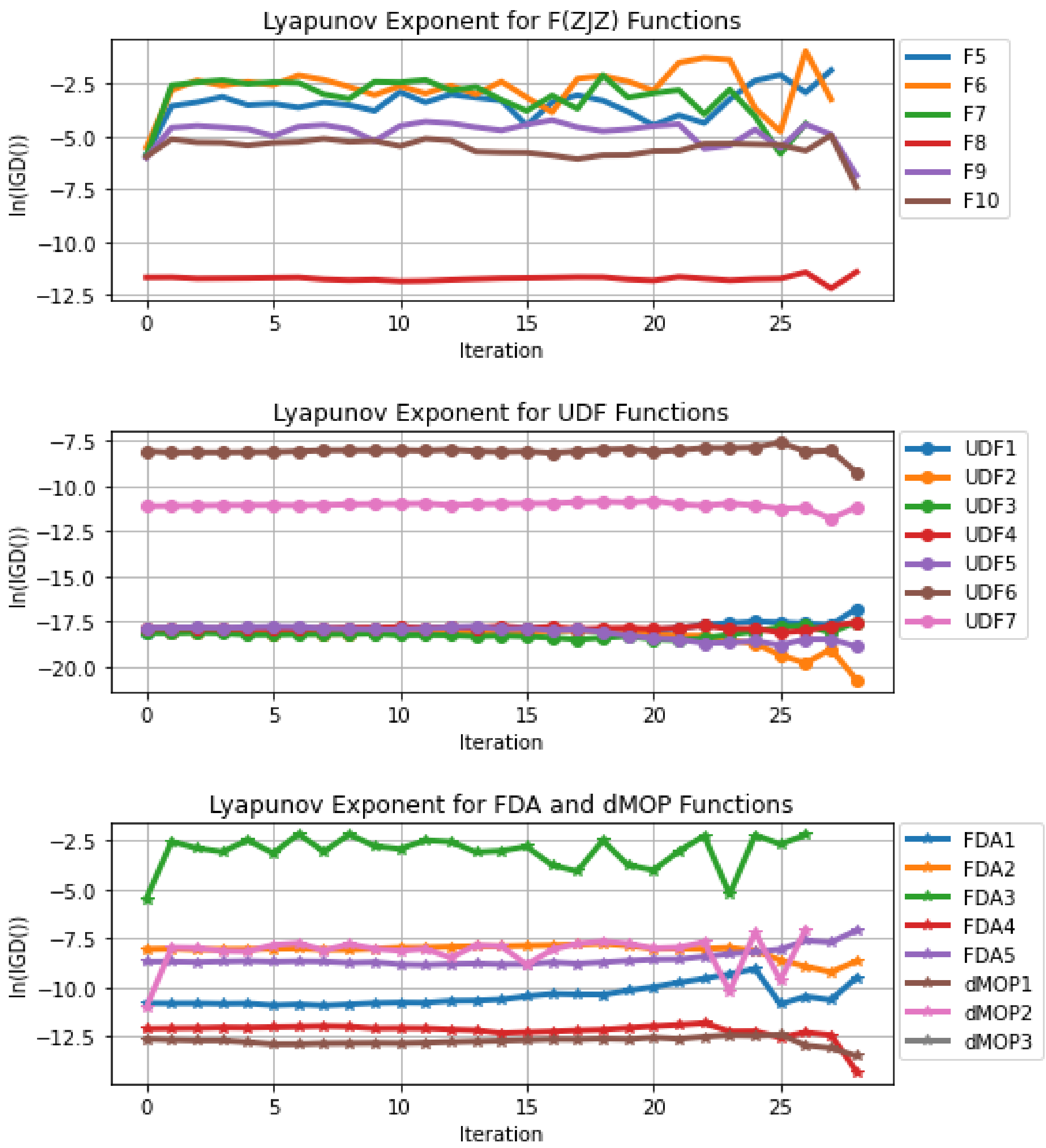


Fig. 9. Spectrum Lyapunov Exponent of DPb-MOPSO for F(ZJZ), UDF, FDA and dMOP Problems.

7. Conclusion

This study presents a novel Dynamic Pareto bi-level Multi-Objective Particle Swarm Optimization (DPb-MOPSO) algorithm for solving different types of DMOPs (I, II and III). The proposed DPb-MOPSO has a distributed architecture with two optimization levels, including a dynamic handling strategy to detect and react to changes. Compared with several DMOEAs, the Friedman's two-way ANOVA test, and the Wilcoxon sum rank test are used to analyze the performance the proposed DPb-MOPSO algorithm. Based on statistical analysis (at 0.05 significance level), DPb-MOPSO proved to be a more robust algorithm because it can ensure a good trade-off between convergence and diversity in a time-dependent environment. Further, the spectrum of Lyapunov exponents has shown to be an efficient tool for analyzing periodic motions and stability analysis in dynamical systems. The superiority of the proposed method is observed when handling DMOPs with two or three objectives including different types of changes in POS and POF. Based on IGD and HVD metrics, the DPb-MOPSO algorithm can solve 8/13 and 8/13 of the 13 UDF and ZJZ functions with moderate changes. Also, the DPb-MOPSO can resolve 7/8 FDA and DMOP benchmarks with severe changes to the MIGD, and 6/8 with moderate changes. DPb-MOPSO assumes 7/8, 6/8, and 5/8 for solving FDA, and dMOP functions on IGD and 6/8, 5/8, and 5/8 on HVD metrics considering severe, moderate, and slight environmental changes respectively. Also, it is the winner for solving 8 DMOPs based on RIGD, and RGD metrics. In conclusion, this study proves the novelty and robustness of DPb-MOPSO in solving different classical dynamic multi-objective problems. For future work, a new variant of the proposed DPb-MOPSO approach will be done for solving DMOPs with dynamic spread and time-varying density of the solutions as well as for pertinent feature selection in real-world problems.

Acknowledgment

We deeply acknowledge Taif University for Supporting this study through Taif University Researchers Supporting Project number (TURSP-2020/327), Taif University, Taif, Saudi Arabia. The research leading to these results has received funding from the Ministry of Higher Education and Scientific Research of Tunisia under the grant agreement number LR11ES48.

Appendices

Table 1. Mean and Standard Deviation Values of DMOEAs over **IGD** for UDF, F Functions with Moderate Environmental Changes ($\tau_t = n_t = 10$).

Prob.	Val.	DMOEAs compared by Jiang *et al*. [19]					MOPSO-based Methods without Dynamism handling strategy			Dynamic MOPSO-based Methods		
		DNSGA-II	**dCOEA**	**PPS**	**MOEA/D**	**SGEA**	**MOPSO**	**dMOPSO**	**pbMOPSO**	**Dynamic-MOPSO**	**DPb-MOPSO** (Random parameters)	**DPb-MOPSO** (Best orthogonal design)
UDF1	M.	1.07E-1	2.91E-1	2.67E-1	1.70E-1	1.24E-1	5.52E-3	5.52E-3	9.26E-5	4.97E-5	**4.64E-5**	6.30e-05
	Std.	2.4E-2-	2.3E-2	2.2E-2	5.1E-2	3.3E-2	8.2E-11	1.0E-10	5.3E-6	4.1E-7	**3.1E-6**	3.9e-06
UDF2	M.	1.12E-1	1.83E-1	2.54E-2	1.16E-1	8.95E-2	5.53E-3	5.53E-3	9.36E-5	4.94E-5	**4.67E-5**	6.50e-05
	Std.	1.0E-2	2.0E-2	5.0E-3	9.5E-3	1.3E-2	1.6E-10	0.0E+0	5.9E-6	4.6E-7	**3.0E-6**	4.4e-06
UDF3	M.	6.06E-1	6.51E-1	4.55E+0	6.06E-1	6.06E-1	1.52E-5	**1.48E-5**	6.17E-5	4.55E-5	4.73E-5	6.16e-05
	Std.	3.3E-6	7.7E-2	1.1E+0	6.3E-5	7.4E-6	8.1E-8	**5.6E-10**	3.8E-6	3.7E-7	2.9E-6	3.7e-06
UDF4	M.	1.70E-1	2.87E-1	1.85E-1	3.19E-1	1.68E-1	8.18E-5	8.18E-5	6.36E-5	5.21E-5	**4.81E-5**	6.29e-05
	Std.	4.7E-2	2.8E-2	8.2E-3	1.3E-1	4.4E-2	2.1E-8	1.8E-10	3.2E-6	1.0E-6	**1.8E-6**	2.6e-06
UDF5	M.	1.18E-1	2.05E-1	2.89E-2	1.61E-1	1.00E-1	8.21E-5	8.21E-5	6.38E-5	5.14E-5	**4.68E-5**	6.14e-05
	Std.	1.2E-2	3.5E-2	1.3E-2	1.4E-2	1.1E-2	1.6E-8	2.0E-10	3.1E-6	1.3E-6	**2.2E-6**	2.8e-06
UDF6	M.	4.57E-1	8.04E-1	1.34E+0	5.31E-1	6.68E-1	2.88E+0	2.88E+0	8.63E-4	1.93E-3	**4.05E-4**	2.86e-03
	Std.	8.7E-2	1.0E-1	7.1E-2	1.6E-1	2.0E-1	2.3E-6	1.8E-6	4.5E-4	2.5E-4	**1.2E-4**	3.5e-04
UDF7	M.	5.24E-1	8.40E-1	6.68E-1	5.08E-1	5.08E-1	3.73E-1	3.81E-1	**1.27E-4**	1.93E-3	7.58E-4	6.80e-04
	Std.	2.2E-2	6.4E-2	4.4E-2	1.4E-1	4.2E-2	2.3E-6	7.7E-3	**1.6E-5**	3.0E-4	4.8E-5	3.7e-05
F5	M.	7.82E-1	8.01E-1	2.69E-1	6.88E-1	4.41E-1	4.80E+0	6.62E+0	9.71E-3	2.75E-3	**2.72E-3**	4.99e-03
	Std.	3.9E-2	2.2E-1	4.3E-2	4.1E-2	4.5E-2	8.2E-2	0.0E+0	5.8E-3	8.9E-4	**6.5E-4**	1.6e-03
F6	M.	3.02E-1	6.57E-1	2.60E-1	3.44E-1	2.90E-1	6.73E-1	6.95E-1	5.76E-2	1.53E-3	1.49E-3	**1.36e-03**
	Std.	2.1E-2	1.3E-1	6.5E-2	5.6E-2	1.3E-2	3.6E-3	1.8E-3	1.9E-2	1.9E-3	4.1E-4	**4.4e-03**
F7	M.	4.19E-1	1.56E+0	2.63E-1	4.18E-1	4.47E-1	4.00E-1	4.26E-1	2.96E-2	**6.41E-3**	9.97E-3	7.76e-03
	Std.	6.9E-3	6.0E-1	7.1E-2	6.0E-2	1.0E-2	1.5E-4	1.3E-8	7.3E-3	**2.3E-3**	5.9E-3	1.6e-03
F8	M.	4.86E-1	4.00E-1	4.56E-1	5.49E-1	2.51E-1	1.15E-3	6.27E-3	3.41E-4	6.84E-4	1.93E-4	**1.07e-04**
	Std.	1.3E-2	6.7E-2	3.1E-2	2.3E-2	1.4E-1	8.1E-5	7.2E-4	1.4E-5	2.4E-5	5.1E-6	**7.1e-06**
F9	M.	4.74E-1	8.87E-1	3.59E-1	4.29E-1	3.65E-1	1.46E+0	1.99E+0	9.65E-3	**1.86E-3**	2.99E-3	6.57e-03
	Std.	2.1E-2	3.3E-1	4.4E-2	2.4E-2	3.4E-2	1.1E-1	0.0E+0	3.4E-3	**7.2E-4**	1.3E-3	1.8e-03
F10	M.	1.05E+0	5.76E-1	3.79E-1	6.39E-1	3.80E-1	1.29E+0	1.76E+0	7.53E-3	**2.35E-3**	3.02E-3	5.49e-03
	Std.	1.5E-1	8.1E-2	8.7E-2	8.6E-2	1.3E-2	1.9E-2	5.2E-8	3.0E-3	**5.9E-4**	8.4E-4	2.1e-03

Table 2. Mean and Standard Deviation Values of DMOEAs over **HVD** for UDF, F Functions with Moderate Environmental Changes ($\tau_t = n_t = 10$).

Prob.	Values	DMOEAs compared by Jiang *et al.* [19]					MOPSO-based Methods without Dynamism handling strategy			Dynamic MOPSO-based Methods		
		DNSGA-II	dCOEA	PPS	MOEA/D	SGEA	MOPSO	dMOPSO	pbMOPSO	Dynamic-MOPSO	**DPb-MOPSO** (Random parameters)	**DPb-MOPSO** (Best orthogonal design)
UDF1	M.	5.14E-1	7.47E-1	7.97E-1	6.12E-1	5.18E-1	2.99E+0	2.99E+0	5.78E-3	5.01E-3	4.65E-3	**4.49e-03**
	Std.	3.2E-2	3.8E-2	5.2E-2	9.4E-2	5.0E-2	1.2E-5	1.1E-7	3.0E-4	1.8E-5	1.7E-4	**2.6e-04**
UDF2	M.	5.51E-1	6.13E-1	4.32E-1	5.42E-1	5.10E-1	2.99E+0	2.99E+0	5.80E-3	5.01E-3	4.66E-3	**4.55e-03**
	Std.	2.4E-2	2.8E-2	1.9E-2	1.7E-2	2.5E-2	9.5E-6	1.1E-7	3.3E-4	1.9E-5	2.7E-4	**2.6e-04**
UDF3	M.	1.22E+0	1.23E+0	1.73E+0	1.22E+0	1.22E+0	5.05E-3	5.04E-3	4.50E-3	4.92E-3	4.63E-3	**4.44e-03**
	Std.	1.9E-3	7.0E-2	3.1E-4	2.4E-3	2.4E-3	1.4E-5	1.1E-7	2.4E-4	7.1E-6	2.4E-4	**2.3e-04**
UDF4	M.	3.47E-1	5.06E-1	3.77E-1	6.41E-1	3.32E-1	2.44E-2	2.39E-2	2.33E-3	6.78E-3	6.97E-4	**4.10e-04**
	Std.	8.3E-2	3.7E-2	2.1E-2	1.9E-1	7.1E-2	1.8E-5	1.6E-7	3.2E-4	1.1E-4	2.8E-4	**2.4e-04**
UDF5	M.	2.78E-1	3.98E-1	2.70E-1	3.65E-1	2.72E-1	2.44E-2	2.39E-2	2.24E-3	6.74E-3	**6.49E-4**	6.76e-04
	Std.	2.5E-2	3.3E-2	1.5E-2	2.7E-2	1.8E-2	1.7E-5	1.5E-7	2.7E4	1.4E-4	**2.8E-4**	3.0e-04
UDF6	M.	9.34E-1	1.26E+0	1.83E+0	1.21E+0	9.77E-1	3.96E+2	3.99E+2	1.50E-2	1.83E-2	**5.86E-3**	1.30e-02
	Std.	1.5E-1	7.2E-2	1.0E-2	1.4E-1	2.0E-1	6.7E-3	3.6E-2	1.8E-2	9.7E-3	**1.6E-3**	8.1e-03
UDF7	M.	2.40E+0	1.91E+0	2.06E+0	2.32E+0	2.06E+0	4.62E+0	1.57E+1	1.93E+0	**1.43E-1**	1.93E+0	1.34e+00
	Std.	7.4E-2	1.7E-1	5.4E-2	2.4E-1	1.2E-1	3.0E-2	5.6E-1	1.2E-1	**1.2E-1**	1.2E+0	1.0e+00
F5	M.	1.25E+0	1.10E+0	4.01E-1	1.19E+0	7.16E-1	4.73E+3	5.23E+3	1.31E+0	**4.86E-2**	2.99E-1	1.79e-01
	Std.	2.5E-2	1.6E-1	9.9E-2	2.9E-2	8.2E-2	9.7E+29	9.7E+2	1.2E+0	**6.4E-2**	2.4E-1	2.6e-01
F6	M.	4.76E-1	9.22E-1	4.92E-1	5.75E-1	3.60E-1	9.86E+3	1.53E+1	1.19E+1	8.87E-2	**5.99E-2**	3.45e-01
	Std.	3.7E-2	1.0E-1	1.5E-1	7.5E-2	2.5E-2	1.4E+30	1.2E+2	1.8E+1	2.5E-1	**5.5E-2**	1.7e+00
F7	M.	6.49E-1	1.22E+0	4.49E-1	6.50E-1	6.05E-1	7.07E+3	1.13E+3	1.05E+1	**3.63E-1**	5.07E+0	1.64e+00
	Std.	1.0E-2	1.5E-1	1.4E-1	2.8E-2	1.5E-2	6.0E+3	0.0E+0	1.5E+1	**6.0E-1**	1.4E+1	2.1e+00
F8	M.	1.06E+0	8.85E-1	1.34E+0	1.06E+0	4.57E-1	1.67E+4	1.34E+4	5.85E-3	3.84E-2	**4.55E-3**	1.02e-02
	Std.	4.6E-2	1.2E-1	1.0E-1	6.6E-2	3.2E-2	1.7E+4	1.1E+4	2.7E-3	4.4E-3	**1.2E-3**	4.6e-03
F9	M.	8.87E-1	1.07E+0	6.88E-1	8.58E-1	5.76E-1	4.98E+3	2.62E+3	1.45E+0	**8.42E-2**	2.22E-1	1.36e-01
	Std.	3.4E-2	1.9E-1	7.7E-2	4.6E-2	7.0E-2	8.4E+29	9.7E-5	1.1E+0	**1.0E-1**	3.5E-1	1.2e-01
F10	M.	1.22E+0	8.58E-1	5.38E-1	1.05E+0	5.77E-1	1.23E+3	1.98E+3	1.33E+0	**6.69E-2**	2.85E-1	1.35e-01
	Std.	5.0E-2	8.8E-2	1.2E-1	5.9E-2	2.3E-2	1.2E+3	0.0E+0	1.0E+0	**6.0E-2**	2.1E-1	1.3e-01

Table 3. Mean and Standard Deviation Values of DMOEAs over **MIGD** for FDA and dMOP Functions with Severe, Moderate and Slight Environmental Changes.

Prob.	(τ_t, n_t)	Values	Transfer Learning-based Methods Tested in [16]						MOPSO-based Methods without Dynamism handling strategy			Dynamic MOPSO-based Methods		
			MMTL-MOEA/D	KF-MOEA/D	PPS-MOEA/D	SVR-MOEA/D	Tr-MOEA/D	RI-MOEA/D	MOPSO	dMOPSO	pbMOPSO	Dynamic-MOPSO	**DPb-MOPSO** (Random parameters)	**DPb-MOPSO** (Best orthogonal design)
FDA1	(5,10)	M.	0.1214	0.4670	0.2485	0.3745	0.3381	0.3166	1.97E-4	5.85E-04	1.13E-2	1.11e-05	1.70e-05	**3.70e-06**
		Std.	1.07E-1	3.38E-1	1.40E-1	3.12E-1	2.14E-1	3.58E-1	3.6E-6	2.2E-5	4.6E-3	4.1e-06	4.9e-06	**1.4e-06**
	(10,10)	M.	0.1199	0.2659	0.2141	0.2332	0.3592	0.2733	1.74E-4	5.52E- 4	5.94E-3	**1.47e-06**	1.63e-06	1.84e-03
		Std.	7.93E-2	1.23E-1	1.22E-1	1.66E-1	3.41E-1	1.83E-1	1.4E-7	3.8E-6	3.3E-3	**1.1e-06**	1.1e-06	1.4e-03
	(20,10)	M.	0.0658	0.1635	0.1018	0.2168	0.1778	0.1959	1.84E-4	5.22E-4	8.80E-3	1.00e-07	**7.35e-08**	**3.31e-07**
		Std.	3.64E-2	9.12E-2	1.25E-1	2.03E-1	2.47E-1	2.36E-1	6.1E-6	1.6E-7	6.7E-3	1.2e-09	**3.0e-09**	**1.4e-07**
FDA2	(5,10)	M.	0.0740	0.1695	0.1023	0.2062	0.1241	0.2127	8.97E-3	4.35E-2	3.46E-3	**2.41e-06**	2.88e-06	5.35e-06
		Std.	3.53E-2	6.51E-2	1.09E-1	1.66E-1	4.72E-2	1.49E-1	7.4E-04	2.9E-5	8.7E-4	**7.2e-07**	1.1e-06	1.1e-06
	(10,10)	M.	0.0842	0.1906	0.1200	0.1965	0.1243	0.2528	8.30E-3	3.61E-2	3.41E-3	**1.70e-06**	2.74e-06	2.60e-06
		Std.	3.34E-2	7.00E-2	2.00E-1	1.31E-1	4.27E-2	1.34E-1	6.1E-4	2.2E-6	5.6E-4	**5.5e-07**	4.4e-07	1.1e-06
	(20,10)	M.	0.0662	0.1335	0.0719	0.1810	0.0785	0.1678	7.92E-3	3.61E-2	7.27E-3	4.23e-06	3.92e-06	**3.23e-06**
		Std.	3.63E-2	4.02E-2	9.86E-2	1.88E-1	3.37E-2	1.44E-1	1.9E-4	2.2E-3	2.0E-3	7.7e-07	8.4e-07	**7.1e-07**
FDA3	(5,10)	M.	0.1428	0.2685	0.3142	0.2250	0.2925	0.3493	2.50E-2	9.52E-2	2.47E-2	9.64e-06	1.35e-05	**6.19e-06**
		Std.	1.11E-1	2.66E-1	2.14E-1	1.81E-1	2.44E-1	4.27E-1	4.9E-5	7.0E-2	2.2E-2	4.3e-06	5.0e-06	**3.4e-06**
	(10,10)	M.	0.0914	0.1429	0.2072	0.1994	0.252	0.2530	3.32E-2	9.50E-2	2.83E-2	8.71e-06	**6.05e-06**	9.53e-06
		Std.	9.77E-2	7.49E-2	1.38E-1	1.93E-1	2.75E-1	3.05E-1	9.7E-6	6.2E-2	2.8E-2	8.1e-06	**2.9e-06**	4.5e-06
	(20,10)	M.	0.0749	0.1349	0.2286	0.1409	0.1442	0.1361	3.28E-2	1.02E-1	1.35E-1	2.66e-05	2.13e-05	**1.76e-06**
		Std.	5.08E-2	1.02E-1	1.76E-1	1.94E-1	8.24E-2	7.58E-2	2.1E-5	6.9E-2	9.8E-2	1.6e-05	1.0e-05	**1.1e-06**
FDA4	(5,10)	M.	0.1523	0.1578	0.2114	0.1866	0.2335	0.1702	2.92E-3	5.93E-2	8.48E-4	9.42e-07	3.56e-07	**2.65e-07**
		Std.	9.67E-2	7.21E-2	1.48E-1	7.83E-2	1.21E-1	4.11E-2	9.6E-5	1.4E-3	4.7E-4	7.3e-08	2.8e-08	**2.5e-08**
	(10,10)	M.	0.1594	0.1311	0.1848	0.1709	0.2180	0.1787	1.81E-3	4.18E-2	1.28E-3	6.35e-07	**1.57e-07**	2.12e-07
		Std.	5.77E-2	4.03E-2	1.75E-1	5.15E-2	1.05E-1	8.33E-2	5.0E-5	3.9E-3	5.3E-4	2.6e-08	**4.4e-09**	1.6e-08
	(20,10)	M.	0.1336	0.125	0.1765	0.1234	0.1998	0.1253	1.92E-3	4.19E-2	5.62E-3	4.81e-07	1.07e-07	**7.56e-08**
		Std.	3.89E-2	4.06E-2	2.02E-1	2.36E-2	9.90E-2	2.66E-2	2.0E-6	2.2E-3	1.8E-3	1.3e-08	3.7e-09	**3.5e-08**
FDA5	(5,10)	M.	0.2081	0.2683	0.2036	0.2120	0.1737	0.2184	3.88E-2	1.42E-1	6.62E-3	9.91e-06	7.36e-06	**1.95e-06**
		Std.	6.47E-2	8.65E-2	7.28E-2	1.05E-1	4.19E-2	1.01E-1	3.1E-4	1.3E-2	1.4E-3	1.7e-06	1.4e-06	**1.0e-06**
	(10,10)	M.	0.1892	0.2369	0.2305	0.1862	0.1752	0.2140	2.87E-2	9.97E-2	1.50E-2	3.24e-06	2.13e-06	**1.45e-06**
		Std.	5.19E-2	7.79E-2	1.04E-1	9.43E-2	4.89E-2	1.01E-1	2.1E-5	8.3E-3	9.4E-3	6.8e-07	2.4e-07	**2.9e-07**
	(20,10)	M.	0.1642	0.1818	0.1895	0.1729	0.1879	0.1968	5.10E-2	1.64E-1	2.35E-2	1.57e-06	**7.34e-07**	1.31e-06
		Std.	6.06E-2	5.76E-2	8.11E-2	9.00E-2	4.56E-2	7.64E-2	1.8E-3	9.6E-3	1.0E-2	1.7e-07	**8.1e-08**	5.6e-07
dMOP1	(5,10)	M.	0.0589	0.1857	0.1269	0.2237	0.2345	0.2421	3.35E-3	3.35E-3	1.76E-4	1.19e-07	**1.18e-07**	1.31e-07
		Std.	3.82E-2	9.13E-2	2.37E-1	8.15E-2	6.53E-2	1.33E-1	5.1E-7	1.0E-6	1.4E-5	3.3e-09	**6.2e-09**	6.5e-09
	(10,10)	M.	0.0543	0.1565	0.0965	0.3266	0.2507	0.2734	2.48E-3	1.59E-2	1.97E-4	9.36e-08	**7.16e-08**	8.59e-08
		Std.	5.52E-2	7.39E-2	2.18E-1	1.99E-1	8.15E-2	1.46E-1	2.6E-7	1.3E-2	2.2E-5	2.0e-09	**2.4e-09**	2.9e-09
	(20,10)	M.	0.0252	0.1145	0.0690	0.1938	0.1204	0.1606	3.47E-3	3.47E-3	1.35E-4	8.41e-08	6.08e-08	**5.98e-08**
		Std.	9.00E-3	5.03E-2	1.95E-1	1.25E-1	9.13E-2	1.63E-1	1.1E-6	4.0E-7	8.6E-6	1.3e-09	2.0e-09	**2.5e-09**
dMOP2	(5,10)	M.	0.0494	0.2258	0.1265	0.1302	0.1311	0.1505	1.77E-4	1.72E-2	1.62E-3	1.29e-06	1.14e-06	**7.23e-07**
		Std.	1.59E-2	1.31E-1	1.34E-1	8.99E-2	6.02E-2	1.58E-1	1.9E-6	1.7E-2	3.5E-3	3.0e-06	2.0e-06	**3.8e-07**
	(10,10)	M.	0.0717	0.1646	0.1102	0.1142	0.1157	0.1586	1.53E-4	1.50E-4	3.70E-4	1.04e-06	1.03e-06	**2.62e-07**
		Std.	4.20E-2	8.01E-2	1.00E-1	8.98E-2	6.03E-2	1.33E-1	1.2E-7	1.9E-8	8.6E-5	5.6e-08	1.3e-07	**4.7e-08**
	(20,10)	M.	0.0261	0.120	0.0771	0.0541	0.0795	0.0609	1.50E-4	1.49E-4	2.81E-4	8.66e-07	1.03e-06	**2.68e-07**
		Std.	8.53E-3	8.70E-2	1.12E-1	4.82E-2	4.89E-2	4.64E-2	2.1E-7	1.6E-7	1.1E-4	7.5e-07	7.6e-07	**5.5e-08**
dMOP3	(5,10)	M.	0.0593	0.1132	0.1136	0.0987	0.1203	0.0729	7.83E-2	8.85E-2	1.10E-1	2.94e-04	7.13e-05	**5.18e-05**
		Std.	3.10E-2	8.72E-2	8.84E-2	7.16E-2	4.29E-2	3.87E-2	2.8E-3	3.7E-3	1.6E-2	1.0e-04	2.0e-05	**1.7e-05**
	(10,10)	M.	0.0683	0.1431	0.0736	0.0897	0.1057	0.0850	1.59E-1	1.88E-1	9.63E-2	2.55e-04	7.46e-05	**7.31e-05**
		Std.	4.26E-2	5.58E-2	6.38E-2	4.56E-2	5.18E-2	5.68E-2	8.8E-3	6.9E-3	2.4E-2	1.1e-04	1.7e-05	**2.1e-05**
	(20,10)	M.	0.0260	0.0730	0.0563	0.0510	0.0575	0.0401	6.38E-1	9.05E-1	1.58E-1	1.58e-04	1.00e-04	**7.55e-05**
		Std.	5.56E-3	4.91E-2	6.87E-2	3.52E-2	3.22E-2	2.57E-2	1.8E-3	3.3E-2	3.1E-2	7.1e-05	1.7e-05	**2.3e-05**

Table 4. Mean and Standard Deviation Values of DMOEAs over **IGD** for FDA and dMOP Functions with Severe, Moderate and Slight Environmental Changes.

Prob.	(τ_t, n_t)	Values	DMOEAs compared by Jiang *et al.* [19]					MOPSO-based Methods without Dynamism handling strategy			Dynamic MOPSO-based Methods		
			DNSGA-II	**dCOEA**	**PPS**	**MOEA/D**	**SGEA**	**MOPSO**	**dMOPSO**	**pbMOPSO**	Dynamic-MOPSO	**DPb-MOPSO** (Random parameters)	**DPb-MOPSO** (Best orthogonal design)
FDA1	(5, 10)	M.	6.40E-1	6.36E-2	2.08E-1	3.56E-1	3.41E-2	**6.05E-5**	1.68E-4	1.13E-2	8.77E-3	1.68E-2	3.33e-03
		Std.	9.8E-2	1.1E-2	8.4E-2	4.9E-2	8.0E-3	**2.0E-6**	8.8E-6	4.6E-3	3.6E-3	8.1E-3	1.2e-03
	(10, 10)	M.	5.82E-2	4.13E-2	4.27E-2	1.21E-1	1.48E-2	5.19E-5	1.45E-4	5.94E-3	1.40E-3	1.72E-3	**2.05e-06**
		Std.	3.8E-3	6.5E-3	1.9E-2	1.1E-2	2.0E-3	1.1E-6	5.1E-6	3.3E-3	8.1E-4	9.7E-4	**1.5e-06**
	(20, 10)	M.	4.14E-2	2.39E-2	1.62E-2	4.04E-2	7.55E-3	**3.90E-5**	1.11E-4	8.80E-3	1.79E-4	1.00E-4	2.98e-04
		Std.	4.2E-3	2.2E-3	7.9E-3	2.2E-3	1.4E-3	**4.8E-7**	2.4E-6	6.7E-3	1.6E-4	1.1E-4	1.2e-04
FDA2	(5, 10)	M.	2.85E-2	7.28E-2	8.13E-2	8.40E-2	1.50E-2	7.00E-3	4.18E-2	3.46E-3	4.01E-3	**3.27E-3**	4.81e-03
		Std.	2.4E-3	3.8E-2	3.0E-2	1.3E-2	1.6E-3	1.2E-3	1.7E-3	8.7E-4	1.0E-3	**8.6E-4**	9.7e-04
	(10, 10)	M.	**1.68E-3**	4.73E-2	6.35E-2	3.38E-2	9.11E-3	4.16E-3	2.32E-2	3.41E-3	3.06E-3	2.88E-3	2.34e-03
		Std.	**9.0E-4**	3.3E-2	1.0E-2	8.8E-3	6.3E-4	4.7E-4	1.4E-3	5.6E-4	1.1E-3	1.1E-3	9.8e-04
	(20, 10)	M.	6.51E-3	3.24E-2	6.27E-2	1.64E-2	6.32E-3	**1.43E-3**	6.46E-3	7.27E-3	1.95E-3	2.54E-3	2.90e-03
		Std.	5.26E-4	4.60E-2	9.07E-3	4.99E-3	4.07E-4	**2.0E-5**	3.1E-4	2.0E-3	6.3E-4	8.1E-4	6.4e-04
FDA3	(5, 10)	M.	2.63E-1	2.63E-1	4.43E-1	2.47E-1	6.25E-2	4.23E-2	5.19E-2	2.47E-2	9.26E-3	1.52E-2	**5.57e-03**
		Std.	6.0E-2	3.5E-2	1.1E-1	2.3E-2	3.8E-2	1.3E-4	3.6E-2	2.2E-2	3.4E-3	5.6E-3	**3.1e-03**
	(10, 10)	M.	1.08E-1	1.95E-1	2.19E-1	1.30E-1	4.03E-2	3.97E-2	5.68E-2	2.83E-2	5.54E-3	**3.96E-3**	8.58e-03
		Std.	3.3E-2	3.2E-2	1.8E-2	2.5E-2	2.9E-2	3.6E-5	4.4E-2	2.8E-2	2.9E-3	**1.7E-3**	4.0e-03
	(20, 10)	M.	9.03E-2	1.26E-1	1.92E-1	5.45E-2	3.52E-2	3.47E-2	4.84E-2	1.35E-1	1.48E-2	8.36E-3	**1.59e-03**
		Std.	2.8E-3	3.1E-2	2.4E-2	8.3E-3	2.9E-2	5.2E-6	3.5E-2	9.8E-2	1.0E-2	5.7E-3	**9.8e-04**
FDA4	(5, 10)	M.	1.49E+0	1.62E-1	3.07E-1	1.36E+0	4.60E-1	6.19E-4	1.28E-2	8.48E-4	9.31E-4	3.44E-4	**2.38e-04**
		Std.	1.2E-1	6.1E-3	1.9E-2	1.6E-1	6.6E-2	1.8E-5	1.7E-3	4.7E-4	1.3E-4	2.7E-5	**2.3e-05**
	(10, 10)	M.	7.63E-1	1.24E-1	2.11E-1	5.77E-1	1.83E-1	6.20E-4	1.25E-2	1.28E-3	5.78E-4	**1.43E-4**	1.91e-04
		Std.	4.4E-2	4.5E-3	2.0E-2	5.4E-2	6.6E-3	1.6E-5	1.8E-3	5.3E-4	4.1E-5	**8.2E-6**	1.4e-05
	(20, 10)	M.	2.62E-1	1.03E-1	1.79E-1	2.22E-1	1.26E-1	3.82E-4	8.17E-3	5.62E-3	4.34E-4	9.60E-5	**6.81e-05**
		Std.	1.6E-2	1.7E-3	3.0E-3	1.3E-2	1.5E-3	1.4E-5	1.2E-3	1.8E-3	1.5E-5	5.7E-6	**3.1e-05**
FDA5	(5, 10)	M.	1.76E+0	4.33E-1	6.55E-1	1.57E+0	5.23E-1	2.50E-2	7.21E-2	6.62E-3	1.05E-2	8.16E-3	**1.75e-03**
		Std.	1.0E-1	4.6E-2	3.1E-2	1.3E-1	3.3E-2	1.8E-4	6.0E-3	1.4E-3	1.8E-3	1.2E-3	**9.1e-04**
	(10, 10)	M.	1.02E+0	3.62E-1	4.80E-1	8.19E-1	3.62E-1	2.16E-2	6.08E-2	1.50E-2	3.30E-3	1.88E-3	**1.31e-03**
		Std.	5.4E-2	4.0E-2	3.5E-2	6.0E-2	8.5E-3	2.0E-4	3.6E-3	9.4E-3	7.1E-4	2.8E-4	**2.6e-04**
	(20, 10)	M.	4.88E-1	3.10E-1	3.71E-1	4.07E-1	3.09E-1	1.42E-2	3.89E-2	2.35E-2	1.77E-3	**7.48E-4**	1.18e-03
		Std.	1.2E-2	2.7E-2	1.2E-2	1.4E-2	2.2E-3	1.3E-4	3.5E-3	1.0E-2	3.2E-4	**6.6E-5**	5.1e-04
dMOP1	(5, 10)	M.	1.31E-1	6.95E-2	4.15E-1	1.36E-2	1.12E-2	4.06E-3	4.39E-3	1.76E-4	1.04E-4	**1.02E-4**	1.18e-04
		Std.	1.1E-2	1.4E-2	7.4E-1	9.0E-3	8.1E-3	2.8E-7	1.8E-3	1.4E-5	3.4E-6	**5.0E-6**	5.9e-06
	(10, 10)	M.	8.83E-3	3.93E-2	5.09E-2	9.39E-3	8.24E-3	2.02E-3	2.29E-3	1.97E-4	8.40E-5	**6.46E-5**	7.73e-05
		Std.	5.0E-3	6.2E-3	9.3E-2	4.3E-3	5.3E-3	2.2E-7	1.4E-3	2.2E-5	1.7E-6	**2.5E-6**	2.6e-06
	(20, 10)	M.	7.39E-3	1.88E-2	4.39E-2	7.17E-3	6.54E-3	2.71E-3	2.71E-3	1.35E-4	7.67E-5	5.51E-5	**5.38e-05**
		Std.	3.2E-3	2.3E-3	8.4E-2	2.7E-3	3.0E-3	1.3E-7	1.0E-7	8.6E-6	1.1E-6	1.6E-6	**2.3e-06**
dMOP2	(5, 10)	M.	6.87E-1	1.20E-1	1.56E-1	4.91E-1	3.02E-2	1.59E-3	4.19E-3	1.62E-3	4.11E-3	3.88E-3	**6.50e-04**
		Std.	7.5E-2	2.0E-2	1.8E-2	4.1E-2	3.4E-3	6.0E-7	6.6E-3	3.5E-3	5.1E-3	6.0E-3	**3.4e-04**
	(10, 10)	M.	1.18E-1	7.32E-2	4.28E-1	1.88E-1	1.21E-2	6.33E-4	2.95E-3	3.70E-4	5.64E-4	6.44E-4	**2.36e-04**
		Std.	9.4E-3	8.9E-3	1.7E-2	1.9E-2	5.7E-4	2.0E-6	8.7E-3	8.6E-5	3.8E-5	7.8E-5	**4.2e-05**
	(20, 10)	M.	1.57E-1	3.46E-2	2.02E-2	5.63E-2	6.32E-3	1.92E-3	2.70E-3	2.81E-4	2.35E-3	2.36E-3	**2.41e-04**
		Std.	6.70E-4	4.32E-3	2.49E-3	3.91E-3	1.74E-4	3.7E-7	4.2E-3	1.1E-4	9.6E-4	8.9E-4	**4.9e-05**
dMOP3	(5, 10)	M.	5.62E-1	4.95E-2	1.76E-1	3.42E-1	1.81E-1	9.13E-1	1.19E+0	1.10E-1	2.18E-1	8.56E-2	**4.66e-02**
		Std.	3.9E-2	4.8E-3	8.0E-2	1.9E-2	9.6E-2	2.9E-1	5.2E-2	1.6E-2	7.2E-2	3.3E-2	**1.5e-02**
	(10, 10)	M.	2.00E-1	**2.95E-2**	1.13E-1	1.68E-1	1.32E-1	2.67E-1	2.95E-1	9.63E-2	1.04E-1	7.69E-2	6.58e-02
		Std.	1.5E-2	**2.4E-3**	1.2E-2	1.0E-2	1.3E-2	2.0E-2	4.2E-3	2.4E-2	3.0E-2	1.3E-2	1.9e-02
	(20 ,10)	M.	1.07E-1	**1.63E-2**	8.99E-2	6.27E-2	8.15E-2	5.04E+0	8.88E+0	1.58E-1	3.47E-1	1.62E-1	6.80e-02
		Std.	8.50E-3	**1.71E-3**	6.74E-3	4.37E-3	1.25E-2	2.7E+0	1.9E-1	3.1E-2	4.1E-2	5.1E-2	2.1e-02

Table 5. Mean and Standard Deviation Values of DMOEAs over **HVD** for FDA and dMOP Functions with Severe, Moderate and Slight Changes.

Prob.	**(τ_t, n_t)**	**Values**	**DMOEAs compared by Jiang et al.** [19]					**MOPSO-based Methods without Dynamism handling strategy**			**Dynamic MOPSO-based Methods**		
			DNSGA-II	**dCOEA**	**PPS**	**MOEA/D**	**SGEA**	**MOPSO**	**dMOPSO**	**pbMOPSO**	Dynamic-MOPSO	**DPb-MOPSO** (Random parameters)	**DPb-MOPSO** (Best orthogonal design)
FDA1	(5, 10)	M.	8.70E-1	1.25E-1	3.87E-1	7.70E-1	8.14E-2	**3.83E-3**	4.48E-3	1.14E-1	1.51E-1	1.56E-1	3.48e-02
		Std.	7.5E-2	2.4E-2	1.0E-1	9.4E-2	2.0E-2	**1.6E-4**	6.3E-5	5.2E-2	5.2E-2	6.9E-2	3.3e-02
	(10, 10)	M.	1.36E-1	8.52E-2	2.97E-1	2.88E-1	3.81E-2	**4.14E-3**	4.56E-3	1.03E-1	6.22E-2	6.65E-2	2.16e-02
		Std.	1.7E-2	2.0E-2	1.6E-2	2.9E-2	1.4E-2	**6.4E-5**	4.0E-5	6.0E-2	3.3E-2	3.7E-2	1.5e-02
	(20, 10)	M.	3.55E-2	5.46E-2	2.84E-1	1.34E-1	2.02E-2	**4.30E-3**	4.65E-3	8.36E-2	6.67E-3	6.57E-3	5.96e-03
		Std.	1.3E-2	1.6E-2	1.5E-2	9.2E-3	1.2E-2	**3.6E-5**	1.7E-5	6.9E-2	7.7E-3	1.3E-2	4.7e-03
FDA2	(5, 10)	M.	4.71E-2	1.85E-1	3.21E-1	1.30E-1	**2.54E-2**	8.99E+0	9.96E+0	6.75E+0	3.04E+0	8.83E+0	2.22e+00
		Std.	1.4E-2	6.4E-2	6.7E-2	2.5E-2	**1.3E-2**	8.5E-1	1.4E+0	9.9E+0	2.9E+0	9.7E+0	4.9e-01
	(10, 10)	M.	2.05E-2	1.24E-1	2.66E-1	6.29E-2	**1.67E-2**	8.54E+0	9.72E+0	1.60E+1	2.55E+0	1.29E+1	2.28e+00
		Std.	1.4E-2	4.6E-2	1.4E-2	1.8E-2	**1.4E-2**	3.2E-1	1.4E+0	6.4E+0	1.6E+0	1.5E+1	5.0e-01
	(20, 10)	M.	1.33E-2	8.64E-2	2.55E-1	3.24E-2	**1.23E-2**	8.43E+0	8.19E+0	8.80E+0	1.23E+0	7.70E+0	4.39e+00
		Std.	1.4E-2	7.0E-2	9.4E-3	1.4E-2	**1.4E-2**	1.8E-2	1.1E+0	1.3E+1	7.4E-1	8.8E+0	8.2e+00
FDA3	(5, 10)	M.	1.54E+0	1.45E+0	1.75E+0	1.66E+0	9.80E-1	4.86E-1	4.27E-1	2.62E-1	1.13E-1	9.59E-2	**5.58e-02**
		Std.	1.6E-1	8.5E-2	1.8E-1	7.8E-2	1.0E-1	8.2E-3	2.6E-1	1.6E-1	6.3E-2	6.4E-2	**4.7e-02**
	(10, 10)	M.	1.09E+0	1.32E+0	1.16E+0	1.12E+0	9.24E-1	5.62E-1	5.27E-1	3.21E-1	8.48E-2	**5.69E-2**	9.99e-02
		Std.	9.9E-2	7.7E-2	4.6E-2	9.3E-2	8.2E-2	6.5E-4	1.9E-1	2.3E-1	5.0E-2	**3.2E-2**	6.3e-02
	(20, 10)	M.	1.04E+0	1.15E+0	1.03E+0	9.47E-1	9.11E-1	5.54E-1	4.13E-1	5.74E-1	2.64E-1	1.26E-1	**3.53e-02**
		Std.	7.9E-2	6.6E-2	7.4E-2	2.2E-2	8.1E-2	2.1E-4	3.6E-1	2.7E-1	1.8E-1	1.1E-1	**2.6e-02**
FDA4	(5, 10)	M.	2.05E+0	3.80E-1	7.77E-1	3.97E+0	1.03E+0	4.17E-2	6.51E+1	3.23E-2	2.13E-2	**3.20E-3**	5.47e-03
		Std.	2.0E-1	2.6E-2	6.8E-2	1.6E+0	1.3E-1	4.6E-3	7.3E+0	4.8E-2	1.1E-2	**1.5E-3**	3.9e-03
	(10, 10)	M.	1.58E+0	2.70E-1	4.34E-1	1.24E+0	2.74E-1	4.26E-2	6.92E+1	3.11E-2	2.28E-2	3.00E-3	**2.08e-03**
		Std.	6.6E-2	3.5E-2	7.2E-2	1.3E-1	2.4E-2	5.0E-3	8.2E+0	4.3E-2	7.6E-3	8.7E-3	**1.2e-03**
	(20, 10)	M.	5.48E-1	1.80E-1	3.34E-1	4.34E-1	1.44E-1	4.41E-2	7.01E+1	1.16E-1	4.57E-2	3.39E-3	**2.03e-03**
		Std.	5.7E-2	2.4E-2	8.3E-3	5.0E-2	2.0E-2	4.0E-3	8.5E+0	6.7E-2	1.2E-2	5.9E-3	**1.5e-03**
FDA5	(5, 10)	M.	6.75E+0	2.76E+0	3.88E+0	7.08E+0	2.70E+0	6.38E-1	9.97E-1	2.78E-1	2.72E-1	1.20E-1	**7.43e-17**
		Std.	1.9E-1	2.8E-1	3.1E-1	1.0E+0	2.2E-1	6.1E-1	5.5E+0	1.2E-1	8.3E-2	7.8E-2	**1.7e-16**
	(10, 10)	M.	5.41E+0	2.37E+0	2.19E+0	4.80E+0	1.88E+0	8.66E-1	5.97E-1	5.18E-1	1.03E-1	1.47E-2	**1.38e-02**
		Std.	1.6E-1	2.7E-1	3.9E-1	2.6E-1	9.3E-2	6.7E-1	3.8E+0	4.5E-1	6.9E-2	9.5E-3	**9.4e-03**
	(20, 10)	M.	2.64E+0	2.02E+0	1.04E+0	2.15E+0	1.78E+0	1.51E+0	1.47E+0	1.24E+0	3.62E-2	**7.10E-3**	1.57e-02
		Std.	1.1E-1	1.8E-1	1.1E-1	1.0E-1	7.1E-2	5.4E-1	5.4E+0	7.2E-1	3.7E-2	**6.3E-3**	1.0e-02
dMOP1	(5, 10)	M.	3.93E-2	1.73E-1	2.86E-1	4.64E-2	3.75E-2	3.29E-1	3.07E-1	3.77E-3	1.21E-3	1.02E-3	**5.22e-04**
		Std.	3.81E-2	3.3E-2	3.6E-1	3.6E-2	2.5E-2	9.9E-5	1.2E-1	5.7E-4	4.3E-4	3.7E-4	**3.2e-04**
	(10, 10)	M.	2.28E-2	1.12E-1	9.27E-2	2.57E-2	1.90E-2	3.29E-1	3.29E-1	2.03E-3	2.53E-3	3.41E-4	**2.81e-04**
		Std.	2.0E-2	2.0E-2	1.3E-1	1.5E-2	1.4E-2	5.9E-5	3.3E-5	1.0E-3	3.4E-4	2.1E-4	**1.7e-04**
	(20, 10)	M.	1.71E-2	5.65E-2	6.02E-2	1.59E-2	1.80E-2	3.29E-1	3.29E-1	3.07E-3	3.49E-3	8.94E-4	**2.11e-04**
		Std.	1.4E-2	8.1E-3	8.1E-2	7.9E-3	1.3E-2	3.6E-5	1.7E-5	5.6E-4	2.2E-4	1.8E-4	**1.6e-04**
dMOP2	(5, 10)	M.	8.06E-1	3.03E-1	3.95E-1	9.04E-1	8.71E-2	1.08E-1	1.98E-2	1.32E-2	1.33E-1	1.08E-1	**3.67e-03**
		Std.	1.1E-1	4.9E-2	3.9E-2	7.3E-2	1.9E-2	8.2E-5	2.2E-1	2.4E-2	8.0E-2	2.5E-2	**7.4e-03**
	(10, 10)	M.	2.90E-1	2.07E-1	1.17E-1	4.46E-1	3.59E-2	1.08E-1	8.61E-2	1.80E-3	1.08E-1	1.16E-1	**9.00e-04**
		Std.	2.5E-2	2.4E-2	4.3E-2	4.2E-2	1.1E-2	6.1E-5	1.2E-1	1.2E-3	2.9E-2	5.2E-3	**6.8e-04**
	(20, 10)	M.	4.50E-2	1.09E-1	5.65E-2	1.98E-1	1.85E-2	1.08E-1	8.57E-2	4.73E-3	7.98E-2	6.65E-2	**6.35e-04**
		Std.	1.2E-2	1.5E-2	6.2E-3	1.4E-2	1.1E-2	3.3E-5	1.2E-1	1.8E-3	4.0E-2	4.7E-2	**6.2e-04**
dMOP3	(5, 10)	M.	9.51E-1	1.05E-1	4.22E-1	7.61E-1	4.07E-1	1.13E+1	1.10E+1	1.89E+0	3.57E-1	1.42E-1	**5.32e-02**
		Std.	3.4E-2	1.6E-2	1.5E-2	5.3E-2	2.4E-2	1.2E+0	1.3E-1	6.2E-1	4.0E-1	1.4E-1	**5.0e-02**
	(10, 10)	M.	4.74E-1	**6.57E-2**	2.79E-1	4.54E-1	3.18E-1	2.49E+0	2.63E+0	1.04E+0	1.74E+1	1.69E+1	1.62e-01
		Std.	2.8E-2	**1.3E-2**	2.7E-2	2.8E-2	2.9E-2	1.1E+0	2.8E-1	5.0E-1	3.0E-1	3.1E+0	1.7e-01
	(20, 10)	M.	2.76E-1	**3.63E-2**	2.21E-1	2.87E-1	2.15E-1	1.63E+1	1.73E+1	2.44E+0	1.58E+1	1.55E+1	1.11e-01
		Std.	2.5E-2	**1.3E-2**	1.5E-2	2.0E-2	3.0E-2	1.9E+0	1.3E-1	7.2E-1	3.0E+0	3.8E+0	6.5e-02

Table 6. Mean and Standard Deviation Values of DMOEAs over **RIGD** for FDA and dMOP Functions.

Prob.	Predict methods	DMOEAs tested by Guo *et al.* [24]					DPb-MOPSO (Best orthogonal design)
		MOEA/D	MOEA/D-DE	SPEA2	MOPSO	NSGAIII	
FDA1	Ave	1.063E-1±3.40E-3	1.105E-1±4.10E-3	2.956E-2±3.49E-4	3.120E-2±9.94E-4	3.612E-1±7.14E-2	**9.37e-05 ± 6.6e-05**
	Adapt	2.810E-2±1.13E-4	3.260E-2±1.50E-3	2.964E-2±4.24E-4	3.096E-2±1.02E-3	2.76E-1±9.54E-2	
	Weight	2.80E-2±5.80E-5	1.2263±0.0091	2.950E-2±4.13E-4	3.116E-2±1.02E-3	3.488E-1±5.29E-2	
FDA2	Ave	1.519E-1±1.50E-3	1.798E-1±7.60E-3	1.766E-1±4.71E-2	1.590E-1±1.51E-3	4.427E-1±1.026E-1	**1.08e-04 ± 2.6e-05**
	Adapt	1.690E-1±5.79E-5	1.448E-1±1.90E-5	1.993E-1±1.03E-1	1.585E-1±6.13E-4	3.818E-1±5.85E-2	
	Weight	1.690E-1±8.47E-5	1.2263±0.0091	1.695E-1±2.17E-2	2.737E-1±7.56E-4	5.349E-1±7.91E-2	
FDA3	Ave	2.024E-1±6.50E-3	3.027E-1±3.25E-2	1.724±0.0000	1.724±0.0000	2.530E-1±2.61E-2	**3.26e-04 ± 1.4e-04**
	Adapt	1.457E-1±1.80E-3	3.336E-1±2.52E-2	1.726±4.55E-16	1.726±4.55E-16	3.940E-1±1.19E-1	
	Weight	1.457E-1±1.40E-3	3.367E-1±2.92E-2	1.724±4.55E-16	1.724±4.55E-16	8.100E-1±3.20E-1	
FDA4	Ave	2.127E-1±6.20E-3	2.301E-1±1.10E-2	9.342E-2±2.03E-3	1.383E-1±2.14E-2	3.059E-1±1.51E-2	**1.06e-05 ± 7.5e-07**
	Adapt	7.420E-2±1.30E-3	9.320E-2±4.60E-3	9.552E-2±4.2E-3	1.257E-1±1.29E02	2.051E-1±3.08E-2	
	Weight	7.370E-2±1.60E-3	8.61E-2±2.30E-3	1.639E-2±2.52E-3	1.829E-2±2.32E-3	1.982E-1±2.45E-3	
FDA5	Ave	3.023E-1±5.50E-3	1.0409±6.06E-2	1.656E-1±3.46E-3	1.978E-1±1.25E-2	4.936E-1±1.94E-1	**6.15e-05 ± 9.1e-06**
	Adapt	1.722E-1±3.00E-3	5.817E-1±5.55E-2	1.657E-1±2.53E-3	1.976E-1±1.52E-2	4.922E-1±1.39E-1	
	Weight	1.729E-1±3.70E-3	5.618E-1±4.98E-2	1.639E-1±2.47E-3	1.942E-1±9.64E-3	3.804E-1±2.33E-1	
dMOP1	Ave	1.642E-1±9.00E-5	1.700E-1±8.56E-4	6.259E-1±1.18E-1	6.496E-1±1.13E-16	8.095E-1±1.64E-1	**4.33e-06 ± 1.6e-07**
	Adapt	1.642E-2±5.69E-5	1.706E-1±1.60E-3	4.258E-1±2.84E-1	6.496E-1±1.13E-1	7.081E-1±1.08E-1	
	Weight	1.642E-2±8.55E-5	1.702E-1±1.10E-4	5.053E-1±2.61E-1	6.496E-1±1.13E-16	6.447E-1±1.89E-1	
dMOP2	Ave	1.915E-2±1.20E-2	1.911E-1±1.14E-2	5.330E-2±2.79E-4	5.419E-2±8.55E-4	2.800E-1±6.69E-2	**1.22e-05 ± 1.4e-06**
	Adapt	5.320E-2±6.66E-4	5.62E-2±1.60E-3	5.325E-2±1.75E-4	5.411E-2±1.03E-3	1.084E-1±2.79E-2	
	Weight	5.320E-2±8.84E-4	5.55E-2±1.30E-3	5.328E-2±3.05E-4	5.422E-2±9.86E-4	1.388E-1±1.80E-2	
dMOP3	Ave	1.3684±3.50E-2	1.3784±6.59E-2	2.963E-1±5.25E-2	2.549E-1±1.12E-3	4.652E-1±8.86E-2	**4.12e-03 ± 1.4e-03**
	Adapt	2.530E-1±1.38E-4	2.600E-1±1.30E-2	3.034E-1±7.40E-2	2.550E-1±1.61E-3	5.220E-1±1.21E-1	
	Weight	2.530E-1±1.65E-4	2.639E-1±2.10E-2	2.980E-1±8.38E-2	2.555E-1±2.34E-3	7.877E-1±1.65E-1	

Table 7. Mean and Standard Deviation Values of DMOEAs over **RGD** for FDA and dMOP Functions.

Prob.	Predict methods	DMOEAs tested by Guo *et al.* [24]					DPb-MOPSO (Best orthogonal design)
		MOEA/D	MOEA/D-DE	SPEA2	MOPSO	NSGAIII	
FDA1	Ave	1.035E-1±3.10E-3	1.968E-1±6.85E-2	2.964E-2±4.27E-04	3.008E-2±5.54E-04	1.794E-1±5.71E-2	**4.11e-04 ± 3.2e-04**
	Adapt	2.810E-2±5.53E-5	2.072E-1±3.34E-2	2.987E-2±6.31E-04	3.0513±5.55E-04	7.87E-2±1.30E-2	
	Weight	2.810E-2±4.54E-5	1.972E-1±5.91E-2	2.963E-2±5.25E-04	3.022±6.16E-04	7.28E-2±1.30E-2	
FDA2	Ave	1.766E-1±1.90E-3	1.768E-1±9.60E-3	2.150E-1±1.04E-2	2.56E8-1±1.11E-2	4.051E-1±1.08E-1	**8.02e-05 ± 3.3e-05**
	Adapt	1.574E-1±1.41E-4	1.468E-1±1.38E-2	2.092E-1±2.66E-2	2.572E-1±1.09E-2	3.382E-1±7.06E-2	
	Weight	1.575E-1±2.37E-4	1.313E-1±2.1E-3	2.166E-1±8.90E-3	2.599E-1±1.23E-2	3.634E-1±5.93E-2	
FDA3	Ave	1.979E-1±6.40E-3	2.001E-1±1.07E-2	9.767E-1±1.39E-16	9.767E-1±1.39E-16	3.067E-1±5.82E-2	**3.93e-04 ± 1.9e-04**
	Adapt	1.439E-1±1.70E-3	1.685E-1±1.54E-2	9.780E-1±3.41E-16	9.780E-1±3.41E-16	3.203E-1±1.17E-1	
	Weight	1.439E-1±1.30E-3	2.067E-1±2.74E-2	9.772E-1±0	9.772E-1±0	3.080E-1±1.58E-1	
FDA4	Ave	4.908E-1±3.90E-3	1.1702±1.15E-1	3.889E-1±6.41E-3	3.041E-1±3.44E-2	5.975E-1±8.9E-3	**1.50e-05 ± 2.2e-06**
	Adapt	4.063E-1±4.95E-4	1.1306±6.79E-2	3.864E-1±9.60E-3	2.999E-1±1.85E-2	5.175E-1±2.73E-2	
	Weight	4.064E-1±4.29E-4	1.1994±6.60E-2	3.857E-1±5.61E-3	3.072E-1±2.64E-2	5.118E-1±2.28E-2	
FDA5	Ave	7.834E-1±35.00E-3	1.7370±1.01E-1	4.316E-1±9.31E-3	3.438E-1±1.62E-2	9.684E-1±1.32E-1	**3.06e-04 ± 1.3e-04**
	Adapt	7.841E-1±3.60E-3	1.6581±1.35E-1	4.295E-1±6.91E-3	3.555E-1±2.62E-2	9.936E-1±9.17E-1	
	Weight	7.841E-1±6.70E-3	1.7339±6.79E-2	4.243E-1±8.30E-3	3.547E-1±1.26E-2	9.853E-1±3.10E-1	
dMOP1	Ave	1.586E-2±1.57E-4	2.085E-1±4.73E-2	1.063E-2±1.81E-2	1.450E-2±6.63E-04	3.811E-1±1.51E-1	**7.97e-06 ± 3.6e-07**
	Adapt	1.387E-2±1.59E-4	1.763E-1±3.03E-2	3.150E-2±3.01E-2	1.457E-2±5.67E-4	1.243E-1±7.64E-2	
	Weight	1.387E-2±1.86E-4	2.121E-1±1.97E-2	2.160E-2±2.75E-2	1.457E-2±5.23E-4	1.676E-1±3.61E-2	
dMOP2	Ave	1.830E-1±1.29E-4	2.116E-1±2.31E-2	5.34E-2±3.16E-4	5.34E-2±3.01E-4	1.866E-1±1.82E-2	**1.70e-05 ± 1.6e-06**
	Adapt	5.290E-2±5.75E-4	9.820E-2±1.09E-2	5.34E-2±2.18E-4	5.34E-2±2.32E-4	8.980E-2±5.70E-3	
	Weight	5.300E-2±7.74E-4	9.79E-2±7.40E-3	5.35E-2±4E-4.03	5.35E-2±2.77E-4	1.357E-1±2.14E-2	
dMOP3	Ave	1.175±2.97E-2	2.5829±5.01E-1	2.528E-1±5.60E-3	2.530E-1±6.01E-3	3.498±2.62E-2	**2.02e-02 ± 1.9e-02**
	Adapt	2.414E-1±1.13E-4	1.8238±6.25E-1	2.649E-1±3.51E-3	2.528E-1±4.80E-3	5.220±1.216E-1	
	Weight	2.414E-1±1.15E-4	1.4660±4.75E-1	2.65E2-1±4.11E-3	2.519E-1±6.12E-3	8.447±3.692E-2	

References

[1] M. Farina, K. Deb, and P. Amato, “Dynamic multiobjective optimization problems: Test cases, approximations, and applications,” *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 5, pp. 425–442, Oct. 2004, doi: 10.1109/TEVC.2004.831456.

[2] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, Apr. 2002, doi: 10.1109/4235.996017.

[3] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proceedings of ICNN’95 - International Conference on Neural Networks*, Nov. 1995, vol. 4, pp. 1942–1948 vol.4. doi: 10.1109/ICNN.1995.488968.

[4] A. Song, W. N. Chen, T. Gu, H. Yuan, S. Kwong, and J. Zhang, “Distributed Virtual Network Embedding System with Historical Archives and Set-Based Particle Swarm Optimization,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 2, pp. 927–942, Feb. 2021, doi: 10.1109/TSMC.2018.2884523.

[5] Y. Wang and L. Li, “Heterogeneous redundancy allocation for series-parallel multi-state systems using hybrid particle swarm optimization and local search,” *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans*, vol. 42, no. 2, pp. 464–474, Mar. 2012, doi: 10.1109/TSMCA.2011.2159585.

[6] H. Chen, M. Li, and X. Chen, “Using diversity as an additional-objective in dynamic multi-objective optimization algorithms,” *2nd International Symposium on Electronic Commerce and Security, ISECS 2009*, vol. 1, pp. 484–487, 2009, doi: 10.1109/ISECS.2009.42.

[7] Z. Zhang, “Multiobjective optimization immune algorithm in dynamic environments and its application to greenhouse control,” *Applied Soft Computing*, vol. 8, no. 2, pp. 959–971, Mar. 2008, doi: 10.1016/J.ASOC.2007.07.005.

[8] C. A. Liu and Y. Wang, “Dynamic multi-objective optimization evolutionary algorithm,” *Proceedings - Third International Conference on Natural Computation, ICNC 2007*, vol. 4, pp. 456–459, 2007, doi: 10.1109/ICNC.2007.340.

[9] K. Deb, U. B. Rao N., and S. Karthik, “Dynamic multi-objective optimization and decision-making using modified NSGA-II: A case study on hydro-thermal power scheduling,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2007, vol. 4403 LNCS, pp. 803–817. doi: 10.1007/978-3-540-70928-2_60.

[10] A. Carlisle and G. Dozier, “Adapting Particle Swarm Optimizationto Dynamic Environments,” *undefined*, 2001.

[11] X. Hu and R. C. Eberhart, “Adaptive particle swarm optimization: Detection and response to dynamic systems,” *Proceedings of the 2002 Congress on Evolutionary Computation, CEC 2002*, vol. 2, pp. 1666–1670, 2002, doi: 10.1109/CEC.2002.1004492.

[12] B. Nasiri and M. R. Meybodi, “History-driven firefly algorithm for optimisation in dynamic and uncertain environments,” *International Journal of Bio-Inspired Computation*, vol. 8, no. 5, pp. 326–339, 2016, doi: 10.1504/IJBIC.2016.079575.

[13] A. Aboud, R. Fdhila, and A. M. Alimi, “MOPSO for dynamic feature selection problem based big data fusion,” in *2016 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2016 - Conference Proceedings*, Feb. 2017, pp. 3918–3923. doi: 10.1109/SMC.2016.7844846.

[14] R. Azzouz, S. Bechikh, and L. ben Said, “Dynamic Multi-objective Optimization Using Evolutionary Algorithms: A Survey,” *Adaptation, Learning, and Optimization*, vol. 20, pp. 31–70, 2017, doi: 10.1007/978-3-319-42978-6_2.

[15] R. Fdhila, C. Walha, T. M. Hamdani, and A. M. Alimi, “Hierarchical design for distributed MOPSO using sub-swarms based on a population Pareto fronts analysis for the grasp planning problem,” *13th International Conference on Hybrid Intelligent Systems, HIS 2013*, pp. 203–208, Oct. 2014, doi: 10.1109/HIS.2013.6920483.

[16] M. Jiang, Z. Wang, L. Qiu, S. Guo, X. Gao, and K. C. Tan, “A Fast Dynamic Evolutionary Multiobjective Algorithm via Manifold Transfer Learning,” *IEEE Transactions on Cybernetics*, vol. 51, no. 7, pp. 3417–3428, May 2020, doi: 10.1109/TCYB.2020.2989465.

[17] C. A. Coello Coello and M. S. Lechuga, “MOPSO: A proposal for multiple objective particle swarm optimization,” *Proceedings of the 2002 Congress on Evolutionary Computation, CEC 2002*, vol. 2, pp. 1051–1056, 2002, doi: 10.1109/CEC.2002.1004388.

[18] A. Aboud, R. Fdhila, and A. M. Alimi, “Dynamic Multi Objective Particle Swarm Optimization Based on a New Environment Change Detection Strategy,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Nov. 2017, vol. 10637 LNCS, pp. 258–268. doi: 10.1007/978-3-319-70093-9_27.

[19] S. Jiang and S. Yang, "A Steady-State and Generational Evolutionary Algorithm for Dynamic Multiobjective Optimization," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 1, pp. 65–82, Feb. 2017, doi: 10.1109/TEVC.2016.2574621.

[20] A. Muruganantham, K. C. Tan, and P. Vadakkepat, "Evolutionary Dynamic Multiobjective Optimization Via Kalman Filter Prediction," *IEEE Transactions on Cybernetics*, vol. 46, no. 12, pp. 2862–2873, Dec. 2016, doi: 10.1109/TCYB.2015.2490738.

[21] L. Cao, L. Xu, E. D. Goodman, C. Bao, and S. Zhu, "Evolutionary Dynamic Multiobjective Optimization Assisted by a Support Vector Regression Predictor," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 2, pp. 305–319, Apr. 2020, doi: 10.1109/TEVC.2019.2925722.

[22] M. Jiang, Z. Huang, L. Qiu, W. Huang, and G. G. Yen, "Transfer Learning-Based Dynamic Multiobjective Optimization Algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 4, pp. 501–514, Aug. 2018, doi: 10.1109/TEVC.2017.2771451.

[23] R. Fdhila, C. Walha, T. M. Hamdani, and A. M. Alimi, "Hierarchical design for distributed MOPSO using sub-swarms based on a population Pareto fronts analysis for the grasp planning problem," *13th International Conference on Hybrid Intelligent Systems, HIS 2013*, pp. 203–208, Oct. 2014, doi: 10.1109/HIS.2013.6920483.

[24] Y. Guo, H. Yang, M. Chen, J. Cheng, and D. Gong, "Ensemble prediction-based dynamic robust multi-objective optimization methods," *Swarm and Evolutionary Computation*, vol. 48, pp. 156–171, Aug. 2019, doi: 10.1016/J.SWEVO.2019.03.015.

[25] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, Dec. 2007, doi: 10.1109/TEVC.2007.892759.

[26] C. K. Goh and K. C. Tan, "A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 1, pp. 103–127, 2009, doi: 10.1109/TEVC.2008.920671.

[27] A. Zhou, Y. Jin, and Q. Zhang, "A Population prediction strategy for evolutionary dynamic multiobjective optimization," *IEEE Transactions on Cybernetics*, vol. 44, no. 1, pp. 40–53, Jan. 2014, doi: 10.1109/TCYB.2013.2245892.

[28] K. B. Lee and J. H. Kim, "DMOPSO: Dual multi-objective particle swarm optimization," *Proceedings of the 2014 IEEE Congress on Evolutionary Computation, CEC 2014*, pp. 3096–3102, Sep. 2014, doi: 10.1109/CEC.2014.6900464.

[29] M. Kim, T. Hiroyasu, M. Miki, and S. Watanabe, "SPEA2+: Improving the Performance of the Strength Pareto Evolutionary Algorithm 2," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3242, pp. 742–751, 2004, doi: 10.1007/978-3-540-30217-9_75.

[30] K. Deb and H. Jain, "An Evolutionary Many-Objective Optimization Algorithm Using Reference-point Based Non-dominated Sorting Approach, Part I: Solving Problems with Box Constraints," 2013, Accessed: May 20, 2022. [Online]. Available: http://www.egr.msu.edu/

[31] H. Li and Q. Zhang, "Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II," *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, vol. 13, no. 2, 2009, doi: 10.1109/TEVC.2008.925798.

[32] C. Raquel and X. Yao, "Dynamic Multi-objective Optimization: A Survey of the State-of-the-Art," *Studies in Computational Intelligence*, vol. 490, pp. 85–106, 2013, doi: 10.1007/978-3-642-38416-5_4.

[33] S. Jiang, S. Yang, X. Yao, K. Tan, M. Kaiser, and N. Krasnogor, "Benchmark Problems for CEC2018 Competition on Dynamic Multiobjective Optimisation," 2018.

[34] R. Azzouz, S. Bechikh, and L. ben Said, "Multi-objective optimization with dynamic constraints and objectives: New challenges for evolutionary algorithms," *GECCO 2015 - Proceedings of the 2015 Genetic and Evolutionary Computation Conference*, pp. 615–622, Jul. 2015, doi: 10.1145/2739480.2754708.

[35] R. Azzouz, S. Bechikh, and L. ben Said, "A dynamic multi-objective evolutionary algorithm using a change severity-based adaptive population management strategy," *Soft Computing 2015 21:4*, vol. 21, no. 4, pp. 885–906, Aug. 2015, doi: 10.1007/S00500-015-1820-4.

[36] S. Biswas, S. Das, P. N. Suganthan, and C. A. C. Coello, "Evolutionary multiobjective optimization in dynamic environments: A set of novel benchmark functions," in *Proceedings of the 2014 IEEE Congress on Evolutionary Computation, CEC 2014*, Sep. 2014, pp. 3192–3199. doi: 10.1109/CEC.2014.6900487.

[37] M. Cámara, J. Ortega, and F. J. Toro, "Parallel processing for multi-objective optimization in dynamic environments," *Proceedings - 21st International Parallel and Distributed Processing Symposium, IPDPS 2007; Abstracts and CD-ROM*, 2007, doi: 10.1109/IPDPS.2007.370433.

[38] E. J. Hughes, "MSOPS-II: A general-purpose many-objective optimiser," in *2007 IEEE Congress on Evolutionary Computation, CEC 2007*, 2007, pp. 3944–3951. doi: 10.1109/CEC.2007.4424985.

[39] S. Seifollahi-Aghmiuni and O. Bozorg Haddad, "Multi Objective Optimization with a New Evolutionary Algorithm," *Water Resources Management 2018 32:12*, vol. 32, no. 12, pp. 4013–4030, Jul. 2018, doi: 10.1007/S11269-018-2034-1.
[40] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, 1997, doi: 10.1109/4235.585893.
[41] C. R. Raquel and P. C. Naval, "An effective use of crowding distance in multiobjective particle swarm optimization," *GECCO 2005 - Genetic and Evolutionary Computation Conference*, pp. 257–264, 2005, doi: 10.1145/1068009.1068047.
[42] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, "Combining convergence and diversity in evolutionary multiobjective optimization," *Evolutionary Computation*, vol. 10, no. 3, pp. 263–282, Sep. 2002, doi: 10.1162/106365602760234108.
[43] Genichi. Taguchi, Rajesh. Jugulum, and Shin. Taguchi, "Computer-Based Robust Engineering : an Essential for DFSS.," p. 240, 2004.
[44] S. Gao, M. Zhou, Y. Wang, J. Cheng, H. Yachi, and J. Wang, "Dendritic Neuron Model with Effective Learning Algorithms for Classification, Approximation, and Prediction," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 2, pp. 601–614, Feb. 2019, doi: 10.1109/TNNLS.2018.2846646.
[45] Y. N. Guo, X. Zhang, D. W. Gong, Z. Zhang, and J. J. Yang, "Novel Interactive Preference-Based Multiobjective Evolutionary Optimization for Bolt Supporting Networks," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 4, pp. 750–764, Aug. 2020, doi: 10.1109/TEVC.2019.2951217.
[46] J. J. Durillo and A. J. Nebro, "JMetal: A Java framework for multi-objective optimization," *Advances in Engineering Software*, vol. 42, no. 10, pp. 760–771, Oct. 2011, doi: 10.1016/j.advengsoft.2011.05.014.
[47] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, Mar. 2011, doi: 10.1016/J.SWEVO.2011.02.002.
[48] N. Rokbani *et al.*, "Bi-heuristic ant colony optimization-based approaches for traveling salesman problem," *Soft Computing*, vol. 25, no. 5, pp. 3775–3794, Mar. 2021, doi: 10.1007/s00500-020-05406-5.
[49] J. J. Ji, Y. N. Guo, X. Z. Gao, D. W. Gong, and Y. P. Wang, "Q-Learning-Based Hyperheuristic Evolutionary Algorithm for Dynamic Task Allocation of Crowdsensing," *IEEE Transactions on Cybernetics*, 2021, doi: 10.1109/TCYB.2021.3112675.
[50] A. M. LYAPUNOV, "The general problem of the stability of motion," *http://dx.doi.org/10.1080/00207179208934253*, vol. 55, no. 3, pp. 531–534, 2007, doi: 10.1080/00207179208934253.
[51] S. Nichele, " Introduction to the Modeling and Analysis of Complex Systems. H. Sayama (Ed.). (2015, Open SUNY Textbooks). Free open access PDF, 498 pp. ISBN 978-1-942341-06-2 (deluxe color edition). ISBN 978-1-942341-08-6 (print edition). ISBN 978-1-942341-09-3 (ebook). ," *Artificial Life*, vol. 22, no. 3, pp. 424–427, Aug. 2016, doi: 10.1162/ARTL_R_00209.
[52] B. B and D. C, "Stochastic neural network model for spontaneous bursting in hippocampal slices," *Phys Rev E Stat Nonlin Soft Matter Phys*, vol. 66, no. 5 Pt 1, p. 18, Nov. 2002, doi: 10.1103/PHYSREVE.66.051908.
[53] S. Bhattacharya, A. Konar, S. Das, and S. Y. Han, "A Lyapunov-Based Extension to Particle Swarm Dynamics for Continuous Function Optimization," *Sensors (Basel)*, vol. 9, no. 12, p. 9977, Dec. 2009, doi: 10.3390/S91209977.

Ahlem Aboud she is currently a PhD student in computer science in ISITCom, University of Sousse and with the REsearch Group on Intelligent Machines (REGIM-lab) at ENIS Sfax, University of Sfax, Tunisia. She received his engineering degree in 2015 from the ISIMS Sfax, Tunisia. She is currently IEEE student member since October 2015. His research interests include, dynamic multi-objective optimization problem, evolutionary computation, collective intelligence methods, and Machine Learning Algorithms.

Nizar Rokbani (IEEE SM), He is an Assit Prof of Industrial Computing with Institute of Applied Science and technology of Sousse, Tunisia where he is a lecturer since 2014. He graduated in Electrical Engineering from the National Engineering School of Tunis, ENIT in 1995. He obtained a Master degree in industrial computing in 2003 from the National Engineering School of Sfax, ENIS, and a PHD in

Electrical Engineering in 2013. He was an automation process engineer until 1998, he was then a technologic trainer at the institute of technological studies of Gabes, occupied a post of training adviser engineer at the national agency of vocational training from 2003 to 2006. His research interests include applications of intelligent techniques such as Swarm intelligence, computational intelligence, fuzzy logic, evolutionary algorithms to robotic systems and industrial processes. Dr. Nizar Rokbani is an IEEE member & volunteer, He participated in funding several society chapters, including RAS and SMC 2009, and OES 2016, he is also the founding counselor of ISSAT Sousse Sb 2015, and Polytech Sousse Sb, 2016. He is The IEEE RAS Tunisia Chair 2015-20120, the chapter was awarded: R8 Outstanding Society Chapter 2016, Best RAS Society Chapter award for 2017, and Outstanding Society chapter award of Tunisia Section for 2017. In 2019, He received the the Meritorious Achievement Award in Pre-University Education of the IEEE Educational Activities Board (EAB) for his efforst in promoting STEM & Robotics education in tunisia primary schools.

Raja Fdhila she holds an engineering degree in computer science and a Master's degree in automation and signal processing from the National School of Engineers of Tunis in 2005 and 2007 respectively. She obtained her doctorate in computer systems engineering from ENIS Sfax in 2017. She is member of the REGIM-lab at ENIS Sfax. She works mainly in the field of artificial intelligence, and is particularly interested in multi-objective optimization, collective intelligence, multi-agent systems. She has been an IEEE member since 2008.

Dr Abdulrahman Qahtani is currently Assistant professor in computer science department at Taif University. Dr. Qahtani received his PhD from Southampton University in 2015. He has extensive experience in software engineering and development process in a distributed domain. His research focused on customization process across organizational boundaries. Recently, he applied machine learning algorithms on software engineering data to predict time and cost estimation for multi-clients' projects.

Omar Almutiry received the PhD degree from the University of Southampton, UK, in 2017. Since 2018, he has been with the Applied Computer Science College, King Saud University, Saudi Arabia, where he is currently working as an Assistant Professor with the Department of Applied Computer Sciences. He is also assistant to the General Director of the Almuzahmiyah branch for academic affairs and development. His research areas include data science, health informatics, deep learning applications in healthcare and medical fields.

Habib Dhahri was born in Sidi Bouzid, Tunisia in 1975. He graduated in Computer Science in 2001, obtained his PhD in Computer Science in 2013 from the National Engineering School of Sfax (ENIS). He is now an assistant professor in Computer Science at the King Saud University. His area of interest includes computational intelligence, soft computing techniques and Machine Learning for Healthcare Data. He has authored and co-authored more than 30 publications in journals and conferences. He also serves as a reviewer for several international scientific journals.

Amir Hussain he received his B.Eng (highest 1st Class Honours with distinction) and Ph.D degrees, from the University of Strathclyde, Glasgow, U.K., in 1992 and 1997, respectively. Following postdoctoral and academic positions at the Universities of West of Scotland (1996-98), Dundee (1998-2000) and Stirling (2000-18) respectively, he joined Edinburgh Napier University (in Scotland, UK) in 2018 as Professor and founding Head of the Data Science and Cyber Analytics (DSCA) Research Group (managing 20 academics and research staff). As part of the latter, he is also founding Head of the Cognitive Big Data and Cybersecurity (CogBiD) Research Lab

Adel M. Alimi (S’91–M’96–SM’00) was born in Sfax, Tunisia, in 1966. He received the Degree in electrical engineering in 1990, the Ph.D. degree in electrical and computer engineering from the Polytechnic University of Montreal, Montr ´ eal, QC, ´ Canada, in 1995, and the HDR degree in electrical and computer engineering from the National Engineering School of Sfax, Sfax, in 2000. He is currently a professor of electrical and computer engineering with the University of Sfax, Sfax. His current research interests include applications of intelligent methods (neural networks, fuzzy logic, and evolutionary algorithms) to pattern recognition, robotic systems, vision systems, industrial processes, intelligent pattern recognition, learning, analysis, and intelligent control of large-scale complex systems. Prof. Alimi is a member of IAPR, INNS, and PRS. He is a member and an Associate Editor of the Editorial Board of many international scientific journals, including Pattern Recognition Letters, Neurocomputing, Neural Processing Letters, International Journal of Image and Graphics, Neural Computing and Applications, International Journal of Robotics and Automation, and International Journal of Systems Science. He was a Guest Editor of several special issues of international journals, including Fuzzy Sets and Systems, Soft Computing, the Journal of Decision Systems, Integrated Computer-Aided Engineering, and Systems Analysis Modeling and Simulation.