Multi-guide Particle Swarm Optimisation for Dynamic Multi-objective Optimisation Problems

Paweł Joćko

Submitted in partial fulfilment
of the requirements for the degree of

Master of Science

Department of Computer Science
Brock University
St. Catharines, Ontario

Abstract

This study investigates the suitability of, and adapts, the multi-guide particle swarm optimisation (MGPSO) algorithm for dynamic multi-objective optimisation problems (DMOPs). The MGPSO is a multi-swarm approach, originally developed for static multi-objective optimisation problems (SMOPs), where each subswarm optimises one of the objectives. It uses a bounded archive that is based on a crowding distance archive implementation. Compared to static optimization problems, DMOPs pose a challenge for meta-heuristics because there is more than one objective to optimise, and the location of the Pareto-optimal set (POS) and the Pareto-optimal front (POF) can change over time. To efficiently track the changing POF in DMOPs using MGPSO, six archive management update approaches, eight archive balance coefficient initialization strategies, and six quantum particle swarm optimisation (QPSO) variants are proposed. To evaluate the adapted MGPSO for DMOPs, a total of twenty-nine well-known benchmark functions and six performance measures were implemented. Three experiments were run against five different environment types with varying temporal and spatial severities. The best strategies from each experiment were then compared with the other dynamic multi-objective optimisation algorithms (DMOAs). An extensive empirical analysis shows that the adapted MGPSO achieves very competitive, and often better, performance compared to existing DMOAs.

Acknowledgements

I am grateful to my supervisor, Prof. Beatrice Ombuki-Berman for her invaluable advice, continuous support, and patience during my MSc study. I would like to express my gratitude to Prof. Andries Engelbrecht for his immense knowledge in the field of optimisation, which was extremely helpful throughout the entire process of writing the thesis. My gratitude also extends to the Faculty of Computer Science for the financial support I received throughout the years. Finally, I would like to thank my parents and my brother for their love and encouragement in the past few years. It would be impossible for me to complete my study without them.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

"Every new beginning comes from some other beginning's end." - Seneca

Many real-world problems are dynamic in nature and typically require more than one objective to be optimised. However, these objectives are often in conflict with one another, where improving one objective deteriorates the other. This implies that there does not exist a single solution to a multi-objective problem, but rather a set of optimal trade-off solutions. The goal, therefore, is to find and track a set of ever changing optimal trade-off solutions. Such problems are called dynamic multi-objective optimisation problems (DMOPs), where each DMOP has either two or three objective functions to optimise. When an optimisation problem has more than three objectives, it is referred to as many-objective optimisation problem.

Dynamic multi-objective optimisation (DMOO) algorithms have many real world practical applications in the context of finance [1], scheduling [2, 3, 4], planning [5], and resource allocation [6, 7]. For over two decades, many *state-of-the-art* dynamic multi-objective optimisation algorithms (DMOAs) have been proposed for solving DMOPs. Some of the most notable designs include genetic algorithms such as the dynamic non-dominated sorting algorithm II (DNSGA-II) [2], steady-state and generational evolutionary algorithm (SGEA) [8], dynamic co-evolutionary algorithm (dCOEA) [9], and multi-swarm based dynamic vector evaluated particle swarm optimisation (DVEPSO) algorithm [10]. One of the main goals of these algorithms is to develop an efficient and computationally inexpensive environment change strategy so that they can quickly track the ever-changing Pareto-optimal set (POS) and Pareto-optimal front (POF). The POS refers to the non-dominated set of the decision variables in the entire feasible search space, whereas POF refers to a set of corresponding optimal solutions in the space of objective functions.

Section 1.1 provides the motivation for this thesis. The main objective and the sub-objectives are discussed in Section 1.2, followed by a list of the contributions made throughout this study in Section 1.3. Lastly, an outline for the remainder of the thesis is given in Section 1.4.

1.1 Motivation

More recently, the multi-guide particle swarm optimisation (MGPSO) algorithm was proposed by Scheepers, Engelbrecht, and Cleghorn [11] for solving static multi-objective optimisation problems (SMOPs). The MGPSO is simple to implement and is computationally efficient. This multi-swarm approach introduced an archive guide, selected from a bounded archive, to balance trade-offs between conflicting objectives. The empirical study has shown that MGPSO is very competitive when compared with the other multi-objective algorithms (MOAs) [11]. Given the promising results of the MGPSO for SMOPs, this paper adapts the MGPSO for DMOPs. More specifically, this study considers archive management strategies to allow MGPSO to track changing POFs efficiently. This paper also examines the effect that these various archive management approaches have on the performance of the MGPSO, with the goal of determining the one that works best on a large variety of DMOPs. The best approach is then used in a comparative study with the other *state-of-the-art* DMOAs. The problems considered in this study have only two or three objectives, and the exact time of the changes is assumed to be known beforehand.

This study also considers alternative archive balance coefficient initialization strategies to allow MGPSO to more efficiently control the influence that the social guide and the archive guide have on the particle's velocity. Once these strategies are defined, a series of experiments are conducted to determine the best strategy for solving DMOPs. The MGPSO in these experiments also includes the re-evaluation of non-dominating solutions archive management strategy. The best balance coefficient initialization strategy is then used in a comparative study with the other DMOAs.

To help MGPSO deal with the diversity loss when solving DMOPs, quantum particle swarm optimisation (QPSO) strategies are explored. Overall, six QPSO variants are considered in the experimental analyses. Two distinct experiments, one where 50% of neutral particles are converted into quantum particles and the other one at 10% quantum proportion are conducted. The best performing strategies from both experiments are then then used in a comparative study with other DMOAs as well as the MGPSO without any quantum particles.

1.2 Objectives

The main objective of this study is to adapt the multi-guide particle swarm optimization algorithm to solve dynamic multi-objective optimization problems. In working towards this goal, the following sub-objectives have been identified:

- to provide an overview of the dynamic multi-objective optimization background that subsequent chapters build upon.
- to discuss the particle swarm optimization algorithms and how they govern the movement of particles to solve single and multi-objective optimization problems.
- to investigate and propose various archive management update approaches to efficiently re-populate the bounded archive with diverse non-dominated solutions.
- to investigate current archive balance coefficient initialization strategies and propose new strategies that take into the account the dynamic nature of the problems.
- to investigate current QPSO techniques used for DMOPs and to propose new variants that take advantage of the bounded archive.
- to perform an extensive empirical and sensitivity analyses of the above mentioned strategies and to study their influence on the performance of the MGPSO.
- to determine the best strategies that allow MGPSO to efficiently track the POS and the POF.
- to compare the performance of the adapted MGPSO with the other *state-of-the-art* DMOO algorithms.

1.3 Contributions

The main contributions of this study are

- The introduction of the adapted MGPSO algorithm for DMOO.
- The finding that the proposed algorithm is capable of solving DMOPs.
- The introduction of six archive management approaches.

- The finding that utilising a local search algorithm as an archive management approach is efficient.
- The introduction of eight balance coefficient update strategies.
- The finding that re-initializing the balance coefficient at each environment change is preferable.
- The introduction of the MGPSO with two QPSO strategies.
- The addition of two alternative sampling methods for both QPSO variants.
- The finding that self-adaptive QPSO outperforms PCX QPSO for most DMOPs.
- The finding that smaller proportion of quantum particles is preferable.
- The finding that the adapted MGPSO is highly competitive and oftentimes outperforms the other DMOAs.

1.4 Thesis Outline

- Chapter 2 covers formal definitions for dynamic multi-objective optimisation on which the subsequent chapters build upon. It includes sections on Pareto-optimal set and Pareto-optimal front, the definition for the dynamic multi-objective optimisation problems, and provides an overview of various dynamic environment types. The formal definitions of the current dynamic multi-objective optimisation benchmark functions and the current performance measures used to evaluate the performance of the dynamic multi-objective optimisation algorithms are given.
- Chapter 3 covers the original particle swarm optimisation algorithm used for solving single-objective optimisation problems, and then the modified, multi-guide particle swarm optimisation used for solving multi-objective optimisation problems is discussed.
- Chapter 4 introduces the multi-guide particle swarm optimization adapted for the dynamic multi-objective optimisation problems. Each change to the to the original algorithm is discussed in detail. It covers the bounded archive update approaches, quantum particle swarm optimisation implementation, and new archive balance coefficient initialization strategies.

- Chapter 5 covers the experimental set-up. It includes sections on performance measures, benchmark functions, dynamic environment types, dynamic multi-objective optimisation algorithms, and a step-by-step process on how to rank the performance of the algorithm that takes into the account the tracking ability of DMOAs.

- Chapter 6 considers six archive management strategies and the effect that these approaches have on the performance of the MGPSO is examined. The goal is to determine the approach that works best on various DMOPs as well as various frequencies of change and severities of change. The approach that performs best is then selected for a comparative study with the other DMOAs.

- Chapter 7 presents a parameter sensitivity analysis of the balance coefficient parameter of the MGPSO. Overall, nine initialization strategies are considered and the one that performs best on twenty-nine benchmark functions and various environment types is considered for the final comparison with the other *state-of-the-art* DMOAs. The MGPSO considered in these experiments uses the re-evaluation of non-dominating solutions archive management strategy.

- Chapter 8 presents the experimental results for the MGPSO with the self-adaptive quantum particles and the parent-centric crossover particles. Varying proportions of quantum particles are also considered. Then, the best QPSO approach is used in a comparative study with the other DMOAs. The MGPSO considered in these experiments uses the re-evaluation of non-dominating solutions archive management strategy and the best performing balance coefficient strategy from the previous chapter.

- Chapter 9 provides a summary of all the findings and conclusions of the presented work. Ideas for future research, based on the presented work, are also given.

Chapter 2

Formal Definitions

This chapter provides definitions that are required for the rest of the thesis. Section 2.1 provides formal definitions with regards to DMOO, followed by Section 2.2 and Section 2.3 that describe DMOPs and performance measures, respectively.

2.1 Dynamic Multi-objective Optimisation

Section 2.1.1 covers the theory and formal definitions introduced by Helbig and Engelbrecht [12] with regards to DMOO. It provides definitions for vector domination and Pareto-optimality, and the main goal when solving DMOPs is given. Section 2.1.2 describes four environment types of DMOPs as well as two control parameters that allow changes to DMOPs with respect to the spatial and temporal severity.

2.1.1 Optimisation Definitions

Let the n_x-dimensional search space (also referred to as the *decision space*) be represented by $S \subseteq \mathbb{R}^{n_x}$ and the feasible space represented by $F \subseteq S$, where $F = S$ for boundary constrained optimisation problems. Let $\mathbf{x} = (x_1, x_2, \ldots, x_{n_x}) \in S$ represent a vector of the decision variables, i.e. a *decision vector*, and let a single objective function be defined as $f_k : \mathbb{R}^{n_x} \to \mathbb{R}$. Then $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_{n_k}(\mathbf{x})) \in O \subseteq \mathbb{R}^{n_x}$ represents an *objective vector* containing n_k objective function evaluations, and O is the *objective space*. A boundary constrained DMOP is then defined as:

$$\begin{aligned} &\textit{minimise} \;\; \mathbf{f}(\mathbf{x}, \mathbf{W}(t)) \\ &\textit{subject to} \;\; \mathbf{x} \in [\mathbf{x}_{min}, \mathbf{x}_{max}]^{n_x} \end{aligned} \tag{2.1}$$

where $\mathbf{W}(t)$ is a matrix of time-dependent control parameters of objective functions at time t, $\mathbf{W}(t) = (\mathbf{w}_1(t), \ldots, \mathbf{w}_{n_k}(t))$, n_x is the number of decision variables, $\mathbf{x} = (x_1, \ldots, x_{n_k}) \in \mathbb{R}^{n_x}$ and $\mathbf{x} \in [\mathbf{x}_{min}, \mathbf{x}_{max}]^{n_x}$ refers to the domain of $\mathbf{x}$, with $\mathbf{x}_{min}$ and $\mathbf{x}_{max}$ referring to the lower and upper bounds of the feasible values for decision variables $\mathbf{x}$.

Two solutions are compared using vector domination, and are defined as follows:

Definition 2.1.1 (Vector Domination)**.** Let f_k be an objective function. Then, a decision vector $\mathbf{x}_1$ dominates another decision vector $\mathbf{x}_2$, denoted by $\mathbf{x}_1 \prec \mathbf{x}_2$, if and only if

- $\mathbf{x}_1$ is at least as good as $\mathbf{x}_2$ for all the objectives, i.e. $f_k(\mathbf{x}_1) \leq f_k(\mathbf{x}_2)$, $\forall k = 1, \ldots, n_k$; and
- $\mathbf{x}_1$ is strictly better than $\mathbf{x}_2$ for at least one objective, i.e. $\exists l = 1, \ldots, n_k$: $f_l(\mathbf{x}_1) < f_l(\mathbf{x}_2)$.

The best decision vectors are referred to as being Pareto-optimal, defined as:

Definition 2.1.2 (Pareto-optimal)**.** A decision vector $\mathbf{x}^*$ is Pareto-optimal if there does not exist a decision vector $\mathbf{x} \neq \mathbf{x}^* \in F$ that dominates it, i.e. $\nexists k : f_k(\mathbf{x}) \prec f_k(\mathbf{x}^*)$. If $\mathbf{x}^*$ is Pareto-optimal, the objective vector, $\mathbf{f}(\mathbf{x}^*)$, is also Pareto-optimal.

The Pareto-optimal set (POS) is the set that contains all the Pareto-optimal decision vectors, defined as:

Definition 2.1.3 (Pareto-optimal Set)**.** The POS is formed by the set of all Pareto-optimal decision vectors, i.e.

$$POS = \{\mathbf{x}^* \in F \mid \nexists \mathbf{x} \in F : \mathbf{x} \prec \mathbf{x}^*\} \tag{2.2}$$

The set of corresponding objective vectors is referred to as the Pareto-optimal front (POF):

Definition 2.1.4 (Pareto-optimal Front)**.** For the objective vector $\mathbf{f}(\mathbf{x})$ and the POS, the POF, $POF \subseteq O$, is defined as

$$POF = \{\mathbf{f} = (f_1(\mathbf{x}^*), f_2(\mathbf{x}^*), \ldots, f_{n_k}(\mathbf{x}^*)) \mid \mathbf{x}^* \in POS\} \tag{2.3}$$

When solving a DMOP, the goal of a dynamic multi-objective optimisation algorithm (DMOA) is to track the POF over time, i.e. for each time step t, to find

$$\begin{aligned} POF(t) = \{\mathbf{f}(t) = (f_1(\mathbf{x}^*, \mathbf{w}_1(t)), f_2(\mathbf{x}^*, \mathbf{w}_2(t)), \ldots, \\ f_{n_k}(\mathbf{x}^*, \mathbf{w}_{n_k}(t))) \mid \mathbf{x}^* \in POS(t)\} \end{aligned} \tag{2.4}$$

2.1.2 Dynamic Environment Types

Farina, Deb, and Amato [13] classified dynamic environments for DMOPs into four distinct environment types, namely:

- **Type I environment:** The optimal variables in the decision space (POS) changes, but the optimal objective values in the objective space (POF) remains unchanged.
- **Type II environment:** Both the POS and the POF change.
- **Type III environment:** The POS remains unchanged, but the POF changes.
- **Type IV environment:** Both the POS and the POF remain unchanged, although other regions of the fitness landscape may change.

The time t of each DMOP is calculated by $t = \frac{1}{n_t}\left\lfloor\frac{\tau}{\tau_t}\right\rfloor$ where τ is the current iteration number, τ_t is the number of iterations for which t remains fixed, and n_t is the number of distinct steps in t. Both n_t and τ_t were introduced by Branke, Salihoglu, and Uyar [14] to allow changes to DMOPs with respect to the spatial severity and the temporal severity, respectively. Spatial severity, often referred to as severity of change, measures the distance between the current and the previous POF; big n_t value makes small changes to the POF, whereas a low n_t value makes the changes bigger. On the other hand, the temporal severity (or frequency of change) determines how often the environment changes; big τ_t value changes the environment slowly compared to a small τ_t value that results in changes being more frequent. For example, $n_t = 1$ and $\tau_t = 10$ modifies a DMOP so that the severity of changes is large and the frequency of changes happens fast. When $n_t = 20$ and $\tau_t = 50$, the spatial severity is very small and the frequency of changes is slow.

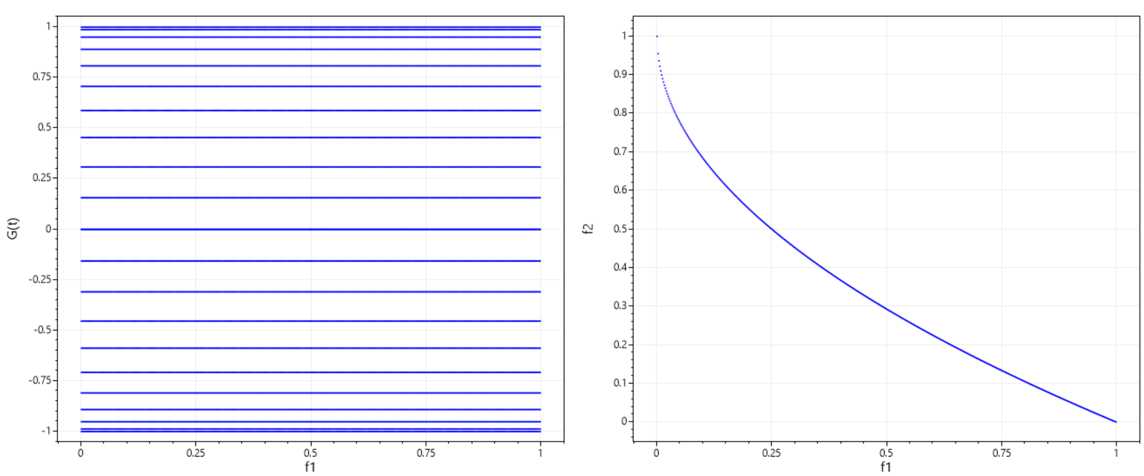
Over the years, many benchmark functions for DMOO have been proposed to evaluate the performance of DMOAs. In the next section, twenty-nine DMOPs are formally defined and their main characteristics are given.

2.2 Dynamic Multi-objective Optimisation Problems

Based on the analysis of DMOPs by Helbig and Engelbrecht [12], this section discusses twenty-nine benchmark functions used to evaluate the performance of DMOO algorithms. The FDA and DIMP test functions are discussed in Section 2.2.1 and in Section 2.2.2, respectively. Sections 2.2.3 covers the dMOP test suite, followed by the DMOPs with isolated and deceptive POF in Section 2.2.4. The HE test suite is covered in Section 2.2.5 and some of the more recently proposed functions are discussed in Section 2.2.6. Section 2.2.7 provides a summary of each DMOP considered in this study.

2.2.1 FDA Test Suite

Farina, Deb, and Amato [13] developed the first DMOPs based on the ZDT [15] and DTLZ [16] benchmark functions for SMOPs. The DMOPs inside the FDA test suite were constructed to have the POS or the POF change over time (Type I or Type III environment), or to have both the POS and the POF change over time (Type II environment). The FDA1-FDA3 are bi-objective DMOPs, whereas FDA4 and FDA5 are tri-objective DMOPs. The POF of these DMOPs are either convex, non-convex or changes from convex to concave over time.



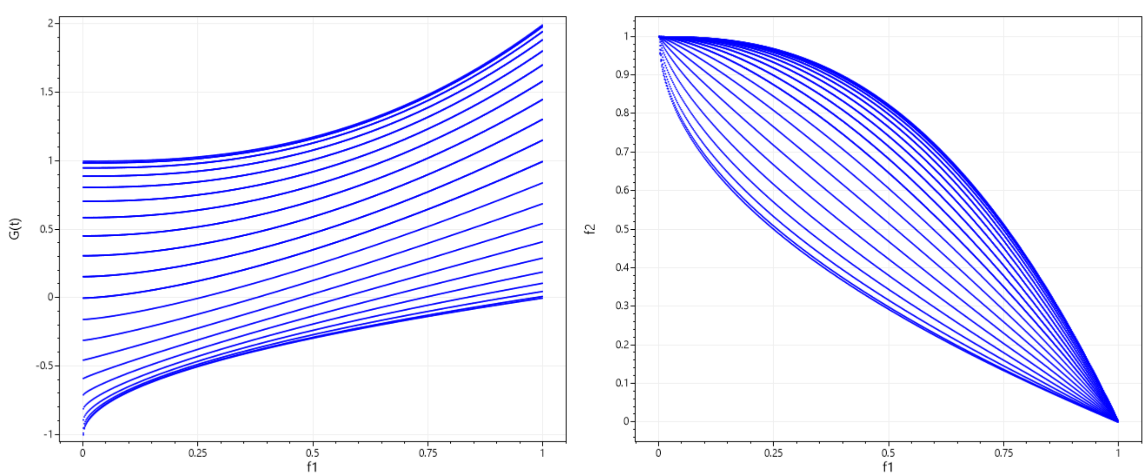
(a) POS **(b)** POF

Figure 2.1 POS and POF of FDA1 with $n_t = 10$ and $\tau_t = 10$ for 1000 iterations.

$$
\text{FDA1} = \begin{cases} Minimize: \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x}_{\mathsf{I}}), f_2(f_1(\mathbf{x}_{\mathsf{I}}), g(\mathbf{x}_{\mathsf{II}}, t))) \\ f_1(\mathbf{x}_{\mathsf{I}}) = x_1 \\ g(\mathbf{x}_{\mathsf{II}}, t) = 1 + \sum_{x_i \in \mathbf{x}_{\mathsf{II}}} (x_i - G(t))^2 \\ f_2(f_1, g) = g \cdot \left(1 - \sqrt{\frac{f_1}{g}}\right) \\ where: \\ G(t) = \sin(0.5\pi t),\ t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ \mathbf{x}_{\mathsf{I}} \in [0, 1];\ \mathbf{x}_{\mathsf{II}} = (x_2, \ldots, x_n) \in [-1, 1]^{n-1} \end{cases} \tag{2.5}
$$

The FDA1 is a Type I DMOP where variables in the decision space (i.e. POS) change over time, but the values in the objective space (i.e. POF) remain the same. It has a convex POF as illustrated in Figure 2.1b and the POS is depicted in Figure 2.1a. The number of decision variables, n_x, is set to 20 as suggested by Farina, Deb, and Amato [13]. The POS and POF of FDA1 is given by

$$
\begin{aligned} &POS(t): 0 \leq x_1 \leq 1,\ x_i = G(t), \forall i = 2, \ldots, n \\ &POF(t): f_2 = 1 - \sqrt{f_1},\ 0 \leq f_1 \leq 1 \end{aligned} \tag{2.6}
$$



(a) POS **(b)** POF

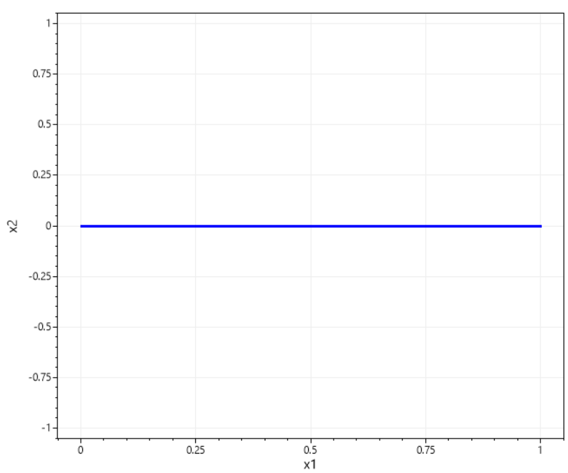
Figure 2.2 The POS and POF of ZJZ with $n_t = 10$ and $\tau_t = 10$ for 1000 iterations.

Another characteristic of the FDA test suite is that there is linear correlation between the decision variables. Since there is no reason that the POS of a real world problem to be this simple, Zhou et al. [17] modified FDA1 to incorporate dependencies between the decision variables. The modified FDA1, called ZJZ, is defined as follows

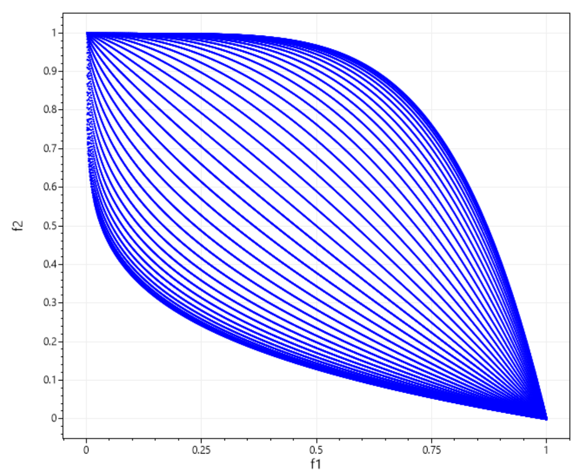
$$\text{ZJZ} = \begin{cases} Minimize : \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x}_{\mathsf{I}}), f_2(f_1(\mathbf{x}_{\mathsf{I}}), g(\mathbf{x}_{\mathsf{II}}, t), t)) \\ f_1(\mathbf{x}_{\mathsf{I}}) = x_1 \\ g(\mathbf{x}_{\mathsf{II}}, t) = 1 + \sum_{x_i \in \mathbf{x}_{\mathsf{II}}} \left(x_i - G(t) - x_1^{H(t)}\right)^2 \\ f_2(f_1, g, t) = g \cdot \left(1 - \left(\frac{f_1}{g}\right)^{H(t)}\right) \\ where : \\ G(t) = \sin(0.5\pi t) \\ H(t) = 1.5 + G(t),\ t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ \mathbf{x}_{\mathsf{I}} \in [0, 1];\ \mathbf{x}_{\mathsf{II}} = (x_2, \dots, x_n) \in [-1, 2]^{n-1} \end{cases} \tag{2.7}$$

For ZJZ, the values of both the POS and the POF change over time, as illustrated in Figure 2.2a and Figure 2.2b, respectively. The POF changes from convex to concave and the number of decision variables, n_x, is set to 10. The POS and the POF is given by

$$\begin{aligned} POS(t) &: 0 \leq x_1 \leq 1,\ x_i = G(t) + x_1^{H(t)}, \forall i = 2, \dots, n \\ POF(t) &: f_2 = 1 - f_1^{H(t)},\ 0 \leq f_1 \leq 1 \end{aligned} \tag{2.8}$$



(a) POS



(b) POF

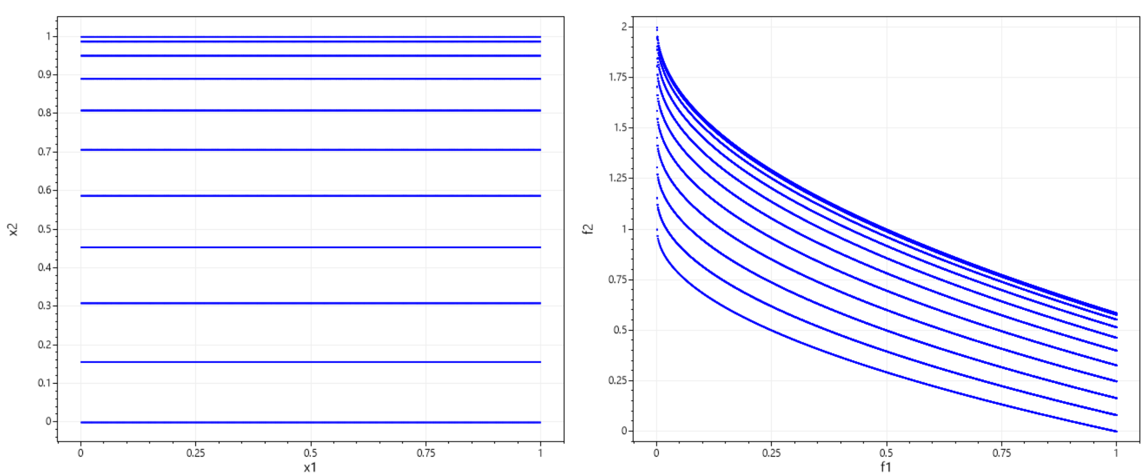
Figure 2.3 The POS and POF of FDA2$_{Cam}$ with $n_t = 10$ and $\tau_t = 10$ for 1000 iterations.

For FDA2 DMOP, the POF changes from a convex to a concave shape only for specific values of the decision variables, making it very difficult to find convex POF instead of concave POF. The following modifications to FDA2 have been proposed by Cámara, Ortega, and Toro [18] to address this issue

$$
\text{FDA2}_{Cam} = \begin{cases} Minimize: \mathbf{f}(\mathbf{x},t) = (f_1(\mathbf{x}_\mathsf{I}), f_2(\mathbf{x}, g(\mathbf{x}_\mathsf{II}), h(\mathbf{x}_\mathsf{III}, f_1(\mathbf{x}_\mathsf{I}), t))) \\ f_1(\mathbf{x}_\mathsf{I}) = x_1 \\ f_2(\mathbf{x}, g, h) = g \cdot h \\ g(\mathbf{x}_\mathsf{II}) = 1 + \sum_{x_i \in \mathbf{x}_\mathsf{II}} x_i^2 \\ h(\mathbf{x}_\mathsf{III}, f_1, t) = 1 - \left(\frac{f_1}{g}\right)^{H_2(t)} \\ where: \\ H(t) = z^{-\cos(\pi t/4)},\ t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ H_2(t) = H(t) + \sum_{x_i \in \mathbf{x}_\mathsf{III}} (x_i - H(t)/2)^2 \\ \mathbf{x}_\mathsf{I} \in [0,1];\ \mathbf{x}_\mathsf{II}, \mathbf{x}_\mathsf{III} \in [-1,1]^{n-1} \end{cases} \tag{2.9}
$$

The FDA2_{Cam} is a Type III DMOP since values of the solutions move only in the objective space. The change to the original FDA2 was made to functions $h(\mathbf{x})$ and $H(t)$ and authors suggest a value of $z = 5$. The sizes are $|\mathbf{x}_\mathsf{II}| = |\mathbf{x}_\mathsf{III}| = 15$, so there are 31 decision variables. Its POF changes from convex to concave, as illustrated in Figure 2.3b, and it has a rather simple POS, as depicted in Figure 2.3a. The POS and POF is given by

$$
\begin{aligned} &POS(t): 0 \leq x_1 \leq 1, x_i = 0,\ \forall x_i \in \mathbf{x}_\mathsf{II} \quad and \quad x_i = \frac{H(t)}{2},\ \forall x_i \in \mathbf{x}_\mathsf{III} \\ &POF(t): f_2 = 1 - f_1^{H(t)},\ 0 \leq f_1 \leq 1 \end{aligned} \tag{2.10}
$$



(a) POS **(b)** POF

Figure 2.4 The POS and POF of FDA3_{Cam} with $n_t = 10$ and $\tau_t = 10$ for 1000 iterations.

The underlying problems with FDA3 have also lead to several modifications being suggested. This study considers the modification by Cámara, Ortega, and Toro [18], formally defined as

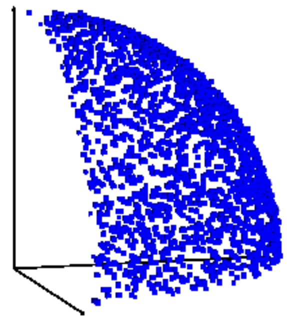
$$\text{FDA3}_{Cam} = \begin{cases} Minimize : \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x}_\mathsf{I}, t), f_2(f_1(\mathbf{x}_\mathsf{I}, t), g(\mathbf{x}_\mathsf{II}, t))) \\ f_1(\mathbf{x}_\mathsf{I}, t) = x_1^{F(t)} \\ g(\mathbf{x}_\mathsf{II}, t) = 1 + G(t) + \sum_{x_i \in \mathbf{x}_\mathsf{II}} (x_i - G(t))^2 \\ f_2(f_1, g) = g \cdot \left(1 - \sqrt{\frac{f_1}{g}}\right) \\ where : \\ G(t) = |\sin(0.5\pi t)| \\ F(t) = 10^{2\sin(0.5\pi t)},\ t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ \mathbf{x}_\mathsf{I} = [0, 1];\ \mathbf{x}_\mathsf{II} = [-1, 1]^{n-1} \end{cases} \tag{2.11}$$

In this function, both the decision and the objective space changes over time. The suggested modification was made only to $f_1(\mathbf{x})$, and $|\mathbf{x}_\mathsf{II}| = 29$ as suggested by Coello, Dhaenens, and Jourdan [19]. In total, there are 30 decision variables. Both the POS and POF is depicted in Figure 2.4a and Figure 2.4b, respectively. The POS and POF of FDA3_{Cam} is given by

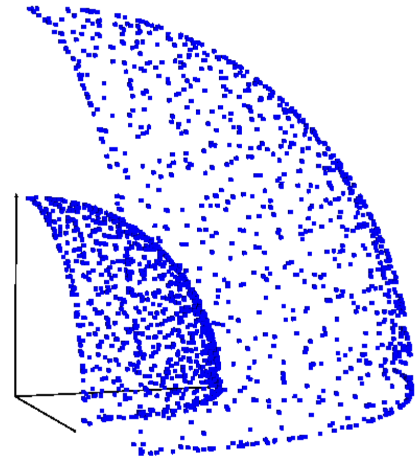
$$\begin{aligned} &POS(t) : 0 \leq x_1 \leq 1,\ x_i = (1 + G(t))\left(1 - \sqrt{\frac{f_1}{1 + G(t)}}\right), \forall i = 2, \ldots, n \\ &POF(t) : f_2 = 1 - \sqrt{f_1},\ 0 \leq f_1 \leq 1 \end{aligned} \tag{2.12}$$

$$\text{FDA4} = \begin{cases} Minimize : \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x}, g(\mathbf{x}_\mathsf{II}, t)), \ldots, f_k(\mathbf{x}, g(\mathbf{x}_\mathsf{II}, t))) \\ f_1(\mathbf{x}, g, t) = (1 + g(\mathbf{x}_\mathsf{II}, t)) \prod_{i=1}^{M-1} \cos\left(\frac{x_i \pi}{2}\right) \\ f_k(\mathbf{x}, g, t) = (1 + g(\mathbf{x}_\mathsf{II}, t)) \left(\prod_{i=1}^{M-1} \cos\left(\frac{x_i \pi}{2}\right)\right) \\ \sin\left(\frac{y_{M-k+1}\pi}{2}\right),\ \forall k = 2, \ldots, M-1 \\ \vdots \\ f_m(\mathbf{x}, g, t) = (1 + g(\mathbf{x}_\mathsf{II}, t)) \prod_{i=1}^{M-1} \sin\left(\frac{x_1 \pi}{2}\right) \\ where : \\ g(\mathbf{x}_\mathsf{II}) = \sum_{x_i \in \mathbf{x}_\mathsf{II}} (x_i - G(t))^2 \\ G(t) = |\sin(0.5\pi t)|,\ t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ \mathbf{x}_\mathsf{II} = (x_M, \ldots, x_n);\ x_i \in [0, 1],\ \forall i = 1, \ldots, n \end{cases} \tag{2.13}$$

For FDA4, values in the decision space change over time, but the values in the



(a) POF of FDA4



(b) POF of FDA5

Figure 2.5 The POF of FDA4 and FDA5 with $n_t = 1$ and $\tau_t = 10$ for 1000 iterations.

objective space remain the same. Therefore, it is a Type I DMOP. It has a non-convex POF, as illustrated in Figure 2.5a. The POS is similar to FDA1 and is depicted in Figure 2.1a. The $M = 3$ and the number of decision variables is set to 12. Thus, FDA4 has three objectives to optimise. The POS and POF is given by

$$\begin{aligned} &POS(t) : 0 \leq x_1, x_2 \leq 1,\ x_i = G(t), \forall i = 3, \ldots, n \\ &POF(t) : f_1 = \cos(u)\cos(v), f_2 = \cos(u)\sin(v), f_3 = \sin(u),\ 0 \leq u, v \leq \pi/2 \end{aligned} \tag{2.14}$$

$$\text{FDA5} = \begin{cases} Minimize : \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x}, g(\mathbf{x_{II}}, t)), \ldots, f_k(\mathbf{x}, g(\mathbf{x_{II}}, t))) \\ f_1(\mathbf{x}, g, t) = (1 + g(\mathbf{x_{II}}, t)) \prod_{i=1}^{M-1} \cos\left(\frac{y_i \pi}{2}\right) \\ f_k(\mathbf{x}, g, t) = (1 + g(\mathbf{x_{II}}, t)) \left(\prod_{i=1}^{M-1} \cos\left(\frac{y_i \pi}{2}\right)\right) \\ \sin\left(\frac{y_{M-k+1}\pi}{2}\right),\ \forall k = 2, \ldots, M-1 \\ \vdots \\ f_m(\mathbf{x}, g, t) = (1 + g(\mathbf{x_{II}}, t)) \prod_{i=1}^{M-1} \sin\left(\frac{y_1 \pi}{2}\right) \\ where : \\ g(\mathbf{x_{II}}, t) = G(t) + \sum_{x_i \in \mathbf{x_{II}}} (x_i - G(t))^2 \\ G(t) = |\sin(0.5\pi t)|,\ t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ y_i = x_y^{F(t)},\ \forall i = 1, \ldots, (M-1) \\ F(t) = 1 + 100\sin^4(0.5\pi t) \\ \mathbf{x_{II}} = (x_M, \ldots, x_n);\ x_i \in [0, 1],\ \forall i = 1, \ldots, n \end{cases} \tag{2.15}$$

FDA5 has a non-convex POF, where both POS and POF changes over time. Therefore, it is a Type II DMOP. The POF is depicted in Figure 2.5b and the POS is similar to FDA1 as in Figure 2.1a. The $M = 3$ and the number of decision variables is set to 12. Thus, FDA5 has three objectives to optimise. The POS and POF of FDA5 is given by

$$\begin{aligned}&POS(t): 0 \leq x_1, x_2 \leq 1,\ x_i = G(t), \forall i = 3, \dots, n\\&POF(t): f_1 = \cos(u)\cos(v), f_2 = \cos(u)\sin(v), f_3 = \sin(u),\ 0 \leq u, v \leq \pi/2\end{aligned} \tag{2.16}$$

2.2.2 DIMP Test Suite

A major drawback of the FDA test suite is that all DMOPs objective functions consist of decision variables with the same rate of change over time. To overcome this shortcoming, Koo, Goh, and Tan [20] proposed two DMOPs, DIMP1 and DIMP2, where each decision variable has different rate of change. For both problems, only the first variable, x_1, remains unchanged since it controls the spread of the solutions. The DIMP1 DMOP is defined as follows

$$\text{DIMP1} = \begin{cases} Minimize: \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x_I}), f_2(f_1(\mathbf{x_I}), g(\mathbf{x_{II}}, t)))\\ f_1(\mathbf{x_I}) = x_1\\ g(\mathbf{x_{II}}, t) = 1 + \sum_{x_i \in \mathbf{x_{II}}} (x_i - G_i(t))^2\\ f_2(f_1, g) = g \cdot \left(1 - \left(\frac{f_1}{g}\right)^2\right)\\ where:\\ G_i(t) = \sin^2\left(0.5\pi t + 2\pi\left(\frac{i}{n+1}\right)\right),\ t = \frac{1}{n_t}\left\lfloor\frac{\tau}{\tau_t}\right\rfloor\\ \mathbf{x_I} = (x_1) \in [0, 1];\ \mathbf{x_{II}} = (x_2, \dots, x_n) \in [-1, 1]^{n-1} \end{cases} \tag{2.17}$$

The POS of DIMP1 changes over time and is similar to the POS of FDA1 (refer to Figure 2.1a). The POF remains the same, as illustrated in Figure 2.6b. Therefore, it belongs to a Type I environment. The POS and POF is given by

$$\begin{aligned}&POS(t): 0 \leq x_1 \leq 1,\ x_i = G(t), \forall i = 2, \dots, n\\&POF(t): f_2 = 1 - f_1^2,\ 0 \leq f_1 \leq 1\end{aligned} \tag{2.18}$$

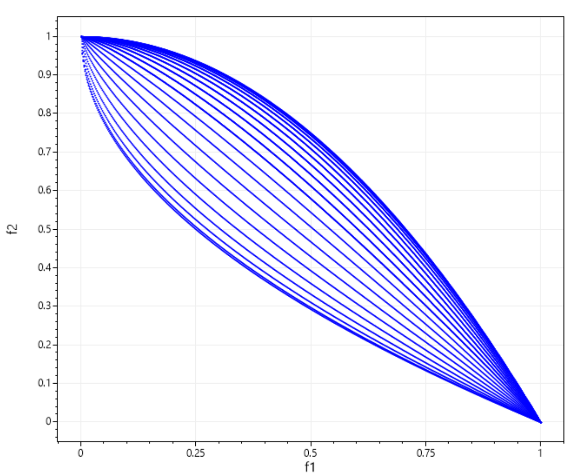
$$\text{DIMP2} = \begin{cases} Minimize: \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x_I}), f_2(f_1(\mathbf{x_I}), g(\mathbf{x_{II}}, t))) \\ f_1(\mathbf{x_I}) = x_1 \\ g(\mathbf{x_{II}}, t) = 1 + 2(n-1) + \sum_{x_i \in \mathbf{x_{II}}} \left[(x_i - G_i(t))^2 - 2\cos\left(3\pi(x_i - G_i(t))\right)\right] \\ f_2(f_1, g) = g \cdot \left(1 - \sqrt{\frac{f_1}{g}}\right) \\ where: \\ G_i(t) = \sin^2\left(0.5\pi t + 2\pi\left(\frac{i}{n+1}\right)\right), \ t = \frac{1}{n_t}\left\lfloor\frac{\tau}{\tau_t}\right\rfloor \\ \mathbf{x_I} = (x_1) \in [0, 1]; \ \mathbf{x_{II}} = (x_2, \ldots, x_n) \in [-2, 2]^{n-1} \end{cases} \tag{2.19}$$

The DIMP2 DMOP also belongs to a Type I environment, since its POS changes over time (refer to Figure 2.1a), but its POF remains the same (refer to Figure 2.1b). Both the POS and POF of DIMP2 is similar to the POS and POF of the FDA1 DMOP. The POS and POF is given by

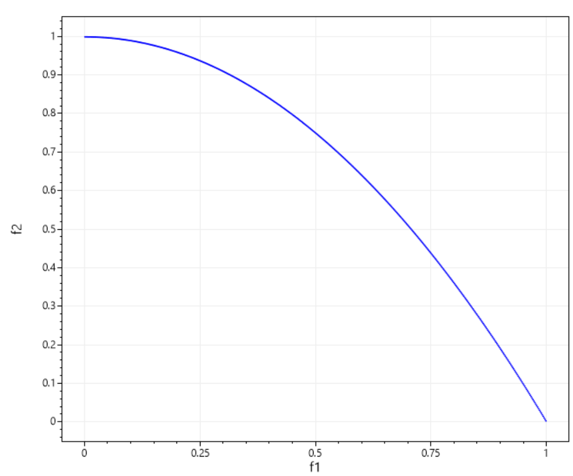
$$\begin{aligned} POS(t) &: 0 \leq x_1 \leq 1, \ x_i = G(t), \forall i = 2, \ldots, n \\ POF(t) &: f_2 = 1 - \sqrt{f_1}, \ 0 \leq f_1 \leq 1 \end{aligned} \tag{2.20}$$

2.2.3 dMOP Test Suite

Goh and Tan [9] presented three DMOPs of Type I, Type II, and Type III, called dMOP1, dMOP2 and dMOP3, respectively. The dMOP1 benchmark function is defined as follows



(a) POF of dMOP1



(b) POF of DIMP1

Figure 2.6 The POF of dMOP1 and DIMP1 with $n_t = 10$ and $\tau_t = 10$ for 1000 iterations.

$$
\text{dMOP1} = \begin{cases} Minimize: \mathbf{f}(\mathbf{x},t) = (f_1(\mathbf{x}_{\mathsf{I}}), f_2(f_1(\mathbf{x}_{\mathsf{I}}), g(\mathbf{x}_{\mathsf{II}}), t)) \\ f_1(\mathbf{x}_{\mathsf{I}}) = x_1 \\ g(\mathbf{x}_{\mathsf{II}}) = 1 + 9\sum_{x_i \in \mathbf{x}_{\mathsf{II}}} (x_i)^2 \\ f_2(f_1, g, t) = g \cdot \left(1 - \left(\frac{f_1}{g}\right)^{H(t)}\right) \\ where: \\ H(t) = 0.75\sin(0.5\pi t) + 1.25,\ t = \frac{1}{n_t}\left\lfloor\frac{\tau}{\tau_t}\right\rfloor \\ x_i \in [0,1];\ \mathbf{x}_{\mathsf{I}} = (x_1);\ \mathbf{x}_{\mathsf{II}} = (x_2, \ldots, x_n) \end{cases} \tag{2.21}
$$

The dMOP1 has a convex POF, as illustrated in Figure 2.6a, where the values in the objective space change. However, the values in the decision space remain the same and the POS is similar to the one from FDA2_{Cam} (refer to Figure 2.3a). The number of decision variables for this Type III problem should be set to $n_x = 10$ as suggested by Goh and Tan [9]. The POS and POF is given by

$$
\begin{aligned} &POS(t): 0 \leq x_1 \leq 1,\ x_i = 0, \forall i = 2, \ldots, n \\ &POF(t): f_2 = 1 - f_1^{H(t)},\ 0 \leq f_1 \leq 1 \end{aligned} \tag{2.22}
$$

$$
\text{dMOP2} = \begin{cases} Minimize: \mathbf{f}(\mathbf{x},t) = (f_1(\mathbf{x}_{\mathsf{I}}), f_2(f_1(\mathbf{x}_{\mathsf{I}}), g(\mathbf{x}_{\mathsf{II}}, t), t)) \\ f_1(\mathbf{x}_{\mathsf{I}}) = x_1 \\ g(\mathbf{x}_{\mathsf{II}}, t) = 1 + 9\sum_{x_i \in \mathbf{x}_{\mathsf{II}}} (x_i - G(t))^2 \\ f_2(f_1, g, t) = g \cdot \left(1 - \left(\frac{f_1}{g}\right)^{H(t)}\right) \\ where: \\ H(t) = 0.75\sin(0.5\pi t) + 1.25 \\ G(t) = \sin(0.5\pi t),\ t = \frac{1}{n_t}\left\lfloor\frac{\tau}{\tau_t}\right\rfloor \\ x_i \in [0,1];\ \mathbf{x}_{\mathsf{I}} = (x_1);\ \mathbf{x}_{\mathsf{II}} = (x_2, \ldots, x_n) \end{cases} \tag{2.23}
$$

The dMOP2 is Type II problem where POF changes from convex to concave (refer to Figure 2.6a), and POS changes in a similar way to FDA1 (refer to Figure 2.1a). The number of decision variables should be set to 10 as suggested by Goh and Tan [9]. The POS and POF is given by

$$
\begin{aligned} &POS(t): 0 \leq x_1 \leq 1,\ x_i = G(t), \forall i = 2, \ldots, n \\ &POF(t): f_2 = 1 - \sqrt{f_1},\ 0 \leq f_1 \leq 1 \end{aligned} \tag{2.24}
$$

$$
\text{dMOP3} = \begin{cases} Minimize : \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x}_{\mathsf{I}}), f_2(f_1(\mathbf{x}_{\mathsf{I}}), g(\mathbf{x}_{\mathsf{II}}, t))) \\ f_1(\mathbf{x}_{\mathsf{I}}) = x_r \\ g(\mathbf{x}_{\mathsf{II}}, t) = 1 + 9 \sum_{x_i \in \mathbf{x}_{\mathsf{II}} \setminus x_r} (x_i - G(t))^2 \\ f_2(f_1, g) = g \cdot \left(1 - \sqrt{\frac{f_1}{g}}\right) \\ where : \\ G(t) = |\sin(0.5\pi t)|, \ t = \frac{1}{n_t} \left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ x_i \in [0, 1]; \ r = \bigcup (1, 2, \ldots, n) \end{cases} \tag{2.25}
$$

$$
\text{dMOP3}_{mod} = \begin{cases} Minimize : \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x}_{\mathsf{I}}), f_2(f_1(\mathbf{x}_{\mathsf{I}}), g(\mathbf{x}_{\mathsf{II}}, t))) \\ f_1(\mathbf{x}_{\mathsf{I}}) = x_r \\ g(\mathbf{x}_{\mathsf{II}}, t) = 1 + \sum_{x_i \in \mathbf{x}_{\mathsf{II}} \setminus x_r} (x_i - G(t))^2 \\ f_2(f_1, g) = g \cdot \left(1 - \sqrt{\frac{f_1}{g}}\right) \\ where : \\ r = 1 + \lfloor (n-1) G(t) \rfloor \\ G(t) = |\sin(0.5\pi t)|, \ t = \frac{1}{n_t} \left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ x_i \in [0, 1] \end{cases} \tag{2.26}
$$

Both dMOP3 and dMOP3$_{mod}$ have a convex POF (similar to FDA1 from Figure 2.1b) that remains the same, and a simple POS (similar to FDA3$_{Cam}$ from Figure 2.4a) that changes over time. Therefore, they belong to a Type I environment. The switch of the position-related variable, x_r, is a challenging dynamic, as it can cause severe diversity loss to population. The dimension, n_x, should be set to 10 for both problems as suggested by Goh and Tan [9]. The POS and POF of dMOP3 is given by

$$
\begin{aligned} POS(t) &: 0 \leq x_1 \leq 1, \ x_i = G(t), \forall i = 2, \ldots, n \\ POF(t) &: f_2 = 1 - \sqrt{f_1}, \ 0 \leq f_1 \leq 1 \end{aligned} \tag{2.27}
$$

The POS and POF of dMOP3$_{mod}$ is given by

$$
\begin{aligned} POS(t) &: 0 \leq x_r \leq 1, \ x_{i \neq r} = G(t), \forall i = 1, \ldots, n \\ POF(t) &: f_2 = 1 - \sqrt{f_1}, \ 0 \leq f_1 \leq 1 \end{aligned} \tag{2.28}
$$

2.2.4 Isolated and Deceptive Test Problems

Helbig and Engelbrecht [12] have made the following modifications to dMOP2 and FDA5 functions, so that they have an isolated POF and a deceptive POF. When a DMOP has an isolated POF, the majority of the search space is fairly flat, making it more difficult to solve because no useful information is provided with regards to the location of the POF. In order to convert FDA5 and dMOP2 DMOPs such that they have an isolated POF, a new function, y^*, creates flat regions by mapping the decision variables to new values. It is defined as

$$y^*(x_i, A, B, C) = A + min(0, \lfloor x_i - B \rfloor)\frac{A(B - x_i)}{B} - min(0, \lfloor C - x_i \rfloor)\frac{(1 - A)(x_i - C)}{1 - C} \tag{2.29}$$

where values for A, B, and C are $G(t)$, 0.001, and 0.05, respectively [12]. The modified DMOPs with an isolated POF are referred to as FDA5$_{iso}$ and dMOP2$_{iso}$ for the rest of the paper.

On the other hand, a DMOP with a deceptive POF is a multi-modal problem, since there exist more than one optima. It is more difficult to solve because the search space favors the deceptive optimum and the algorithm might get stuck in the local POF. In order to convert FDA5 and dMOP2 into DMOPs with a deceptive POF, a different definition for the y^* function is used. It is defined as

$$\begin{gathered} y^*(x_i, A, B, C) = 1 + (|x_i - A| - B) \\ \left(\frac{\lfloor x_i - A + B \rfloor \left(1 - C + \frac{A-B}{B}\right)}{A - B} + \frac{1}{B} + \frac{\lfloor A + B - x_i \rfloor \left(1 - C + \frac{1-A-B}{B}\right)}{1 - A - B}\right) \end{gathered} \tag{2.30}$$

where values for A, B, and C are $G(t)$, 0.001, and 0.05, respectively [12]. The modified DMOPs with a deceptive POF are referred to as FDA5$_{dec}$ and dMOP2$_{dec}$ for the rest of the paper.

The modified FDA5 DMOPs, FDA5$_{iso}$ and FDA5$_{dec}$, are defined as

$$\begin{matrix} \text{FDA5}_{iso} \\ \text{FDA5}_{dec} \end{matrix} = \begin{cases} Minimize: \mathbf{f}(\mathbf{x},t) = (f_1(\mathbf{x}, g(\mathbf{x_{II}},t)), \ldots, f_k(\mathbf{x}, g(\mathbf{x_{II}},t))) \\ f_1(\mathbf{x},g,t) = (1+g(\mathbf{x_{II}},t)) \prod_{i=1}^{M-1} \cos\left(\frac{y_i \pi}{2}\right) \\ f_k(\mathbf{x},g,t) = (1+g(\mathbf{x_{II}},t)) \left(\prod_{i=1}^{M-1} \cos\left(\frac{y_i \pi}{2}\right)\right) \\ \sin\left(\frac{y_{M-k+1}\pi}{2}\right), \ \forall k = 2, \ldots, M-1 \\ \vdots \\ f_m(\mathbf{x},g,t) = (1+g(\mathbf{x_{II}},t)) \prod_{i=1}^{M-1} \sin\left(\frac{y_1 \pi}{2}\right) \\ where: \\ g(\mathbf{x_{II}},t) = \sum_{x_j \in \mathbf{x_{II}}} (x_i - G(t))^2 \\ G(t) = |\sin(0.5\pi t)|, \ t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ F(t) = 1 + 100 \sin^4(0.5\pi t) \\ y_i = x_y^{F(t)}, \ \forall i = 1, \ldots, (M-1) \\ y_j = y^*(x_j, A, B, C), \ \forall x_i \in \mathbf{x_{II}} \\ \mathbf{x_{II}} = (x_M, \ldots, x_n); \ x_i \in [0,1], \ \forall i = 1, \ldots, n \end{cases} \quad (2.31)$$

where the POS and POF of FDA5$_{iso}$ and FDA5$_{dec}$ is given by

$$\begin{aligned} &POS(t): 0 \leq x_1, x_2 \leq 1, \ x_i = G(t), \forall i = 3, \ldots, n \\ &POF(t): f_1 = \cos(u)\cos(v), f_2 = \cos(u)\sin(v), f_3 = \sin(u), \ 0 \leq u, v \leq \pi/2 \end{aligned} \quad (2.32)$$

The modified dMOP2 DMOPs, dMOP2$_{iso}$ and dMOP2$_{dec}$, are defined as

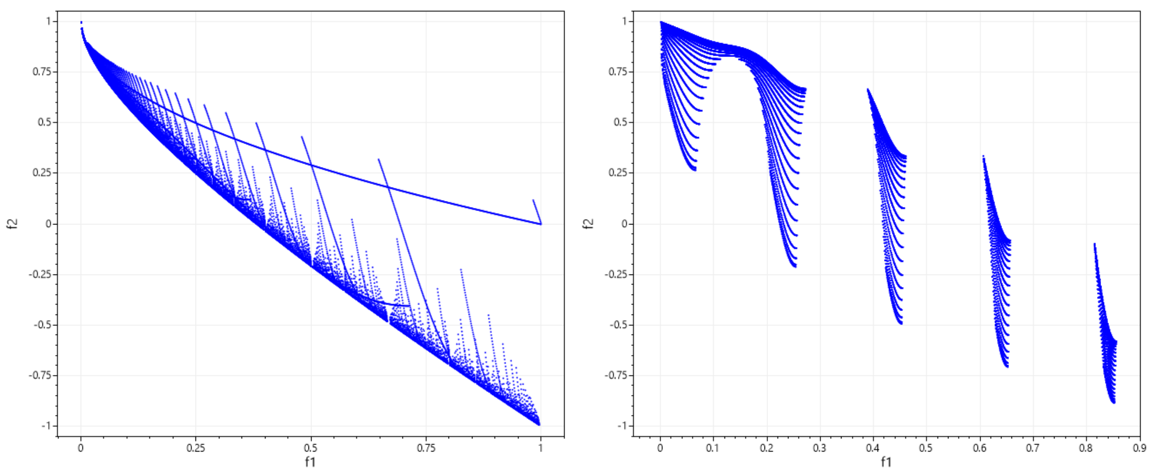
$$\begin{matrix} \text{dMOP2}_{iso} \\ \text{dMOP2}_{dec} \end{matrix} = \begin{cases} Minimize: \mathbf{f}(\mathbf{x},t) = (f_1(\mathbf{x_I}), f_2(f_1(\mathbf{x_I}), g(\mathbf{x_{II}},t),t)) \\ f_1(\mathbf{x_I}) = x_1 \\ g(\mathbf{x_{II}},t) = 1 + 9\sum_{x_i \in \mathbf{x_{II}}} (x_i - G(t))^2 \\ f_2(f_1,g,t) = g \cdot \left(1 - \left(\frac{f_1}{g}\right)^{H(t)}\right) \\ where: \\ y_i = y^*(x_i, A, B, C), \ \forall x_i \in \mathbf{x_{II}} \\ H(t) = 0.75\sin(0.5\pi t) + 1.25 \\ G(t) = \sin(0.5\pi t), \ t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ x_i \in [0,1]; \ \mathbf{x_I} = (x_1); \ \mathbf{x_{II}} = (x_2, \ldots, x_n) \end{cases} \quad (2.33)$$

where the POS and POF of dMOP2$_{iso}$ and dMOP2$_{dec}$ is given by

$$
\begin{aligned}
POS(t) &: 0 \leq x_1 \leq 1,\ x_i = G(t), \forall i = 2, \ldots, n \\
POF(t) &: f_2 = 1 - f_1^{H(t)},\ 0 \leq f_1 \leq 1
\end{aligned}
\tag{2.34}
$$

2.2.5 HE Test Suite

The FDA and dMOP test suites contain DMOPs with only a continuous POF. More recently, Helbig and Engelbrecht [21] presented two DMOPs with a discontinuous POF, namely HE1 and HE2. These two functions are based on the ZDT3 [15] MOP. The HE1 DMOP is defined as



(a) POF of HE1 **(b)** POF of HE2

Figure 2.7 The POF of HE1 and HE2 with $n_t = 10$ and $\tau_t = 10$ for 1000 iterations.

$$
\text{HE1} = \begin{cases}
Minimize : \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x}_\mathsf{I}), f_2(f_1(\mathbf{x}_\mathsf{I}), g(\mathbf{x}_\mathsf{II}), t)) \\
f_1(\mathsf{x}_\mathsf{I}) = x_1 \\
g(\mathbf{x}_\mathsf{II}) = 1 + \frac{9}{n-1} \sum_{x_i \in \mathbf{x}_\mathsf{II}} x_i \\
f_2(f_1, g, t) = g \cdot \left(1 - \sqrt{\frac{f_1}{g}} - \frac{f_1}{g} \sin\left(10\pi t f_1\right)\right) \\
where : \\
t = \frac{1}{n_t} \left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\
x_i \in [0, 1];\ \mathbf{x}_\mathsf{I} = (x_1);\ \mathbf{x}_\mathsf{II} = (x_2, \ldots, x_n)
\end{cases}
\tag{2.35}
$$

The HE1 DMOP has a discontinuous POF with various disconnected continuous sub-regions, as illustrated in Figure 2.7a. The POS is rather simple and is similar to the POS of FDA2$_{Cam}$ (refer to Figure 2.3a). Therefore, HE1 belongs to a Type III environment. The dimension, n_x, is set to 30 as suggested by Helbig and Engelbrecht

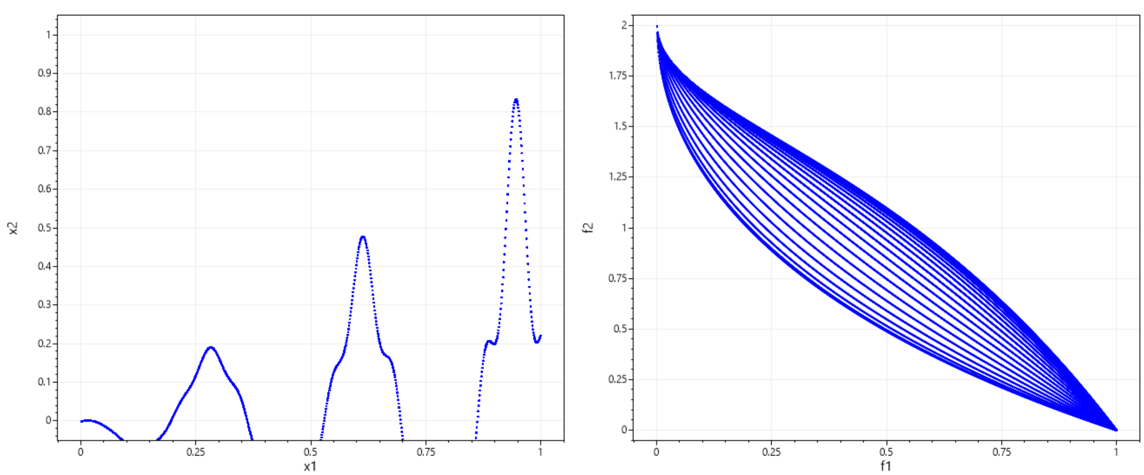
[21] and the POS and the POF is given by

$$
\begin{aligned}
&POS(t): 0 \leq x_1 \leq 1,\ \ x_i = 0, \forall i = 2, \ldots, n \\
&POF(t): f_2 = 1 - \sqrt{f_1} - f_1 \sin\left(10\pi t f_1\right),\ \ 0 \leq f_1 \leq 1
\end{aligned}
\tag{2.36}
$$

$$
\text{HE2} = \begin{cases}
Minimize: \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x}_{\mathsf{I}}), f_2(f_1(\mathbf{x}_{\mathsf{I}}), g(\mathbf{x}_{\mathsf{II}}), t)) \\
f_1(\mathbf{x}_{\mathsf{I}}) = x_i \\
g(\mathbf{x}_{\mathsf{II}}) = 1 + \frac{9}{n-1} \sum_{x_i \in \mathbf{x}_{\mathsf{II}}} x_i \\
f_2(f_1, g, t) = g \cdot \left(1 - \left(\sqrt{\frac{f_1}{g}}\right)^{H(t)} - \left(\frac{f_1}{g}\right)^{H(t)} \sin\left(10\pi f_1\right)\right) \\
where: \\
H(t) = 0.75 \sin\left(0.5\pi t\right) + 1.25,\ \ t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\
x_i \in [0, 1];\ \ \mathbf{x}_{\mathsf{I}} = (x_1);\ \ \mathbf{x}_{\mathsf{II}} = (x_2, \ldots, x_n)
\end{cases}
\tag{2.37}
$$

The HE2 DMOP also has a discontinuous POF with various disconnected continuous sub-regions, as depicted in Figure 2.7b. The POS remains unchanged and is similar to the POS of FDA2$_{Cam}$ (refer to Figure 2.3a) Therefore, HE2 belongs to a Type III environment. The dimension, n_x, is set to 30 as suggested by Helbig and Engelbrecht [21] and the POS and the POF is given by

$$
\begin{aligned}
&POS(t): 0 \leq x_1 \leq 1,\ \ x_i = -, \forall i = 2, \ldots, n \\
&POF(t): f_2 = 1 - \sqrt{f_1}^{H(t)} - f_1^{H(t)} \sin\left(0.5\pi f_1\right),\ \ 0 \leq f_1 \leq 1
\end{aligned}
\tag{2.38}
$$



(a) POS **(b)** POF

Figure 2.8 The POS and POF of HE7 with $n_t = 10$ and $\tau_t = 10$ for 1000 iterations.

Most of the DMOPs introduced so far have a rather simplistic POS. To address this issue, Helbig and Engelbrecht [12] introduced the following two DMOPs, HE7 and HE9, to have a complex POS. The POS for both of these problems is defined by non-linear curves and are different for each decision variable. The HE7 DMOP is defined as

$$\text{HE7} = \begin{cases} Minimize: \mathbf{f}(\mathbf{x},t) = (f_1(\mathbf{x}), f_2(f_1(\mathbf{x}), g(\mathbf{x}), t)) \\ f_1(\mathbf{x}) = x_1 + \frac{2}{|J_1|} \cdot \\ \qquad \sum_{j\in J_1}\left(x_j - [0.3x_1^2\cos\left(24\pi x_1 + \frac{4j\pi}{n}\right) + 0.6x_1]\cos\left(6\pi x_1 + \frac{j\pi}{n}\right)\right)^2 \\ g(\mathbf{x}) = 2 - \sqrt{x_1} + \frac{2}{|J_2|} \cdot \\ \qquad \sum_{j\in J_2}\left(x_j - [0.3x_1^2\cos\left(24\pi x_1 + \frac{4j\pi}{n}\right) + 0.6x_1]\sin\left(6\pi x_1 + \frac{j\pi}{n}\right)\right)^2 \\ f_2(f_1, g, t) = g \cdot \left(1 - \left(\frac{f_1}{g}\right)^{H(t)}\right) \\ where: \\ H(t) = 0.75\sin(0.5\pi t) + 1.25,\ t = \frac{1}{n_t}\left\lfloor\frac{\tau}{\tau_t}\right\rfloor \\ J_1 = \{j \mid \text{j is odd and } 2 \le j \le n\} \\ J_2 = \{j \mid \text{j is even and } 2 \le j \le n\} \\ x_1 \in [0,1];\ x_i \in [-1,1];\ \forall i = 2,3,\ldots,n \end{cases} \tag{2.39}$$

The HE7 DMOP belongs to a Type III environment, since the POF changes over time (refer to Figure 2.8b), but the POS remains the same (refer to Figure 2.8a). The number of decision variables is set to 30 and the POS and POF is given by

$$POS: x_j = \begin{cases} a\cos\left(\frac{6\pi x_1 + \frac{j\pi}{n}}{3}\right),\ j \in J_1 \\ a\cos\left(6\pi x_1 + \frac{j\pi}{n}\right),\ j \in J_2 \\ with: \\ a = \left[0.3x_1^2\cos\left(24\pi x_1 + \frac{4j\pi}{n}\right) + 0.6x_1\right] \end{cases}$$

$$POF: y = (2 - \sqrt{x_1})\left[1 - \left(\frac{x_1}{2-\sqrt{x_1}}\right)^{H(t)}\right]$$

The HE9 DMOP is defined as

$$\text{HE9} = \begin{cases} Minimize : \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x}), f_2(f_1(\mathbf{x}), g(\mathbf{x}), t)) \\ f_1(\mathbf{x}) = x_1 + \frac{2}{|J_1|} \sum_{j \in J_1} \left(4 \sum_{j \in J_1} y_j^2 - \prod_{j \in J_1} \cos\left(\frac{20 y_j \pi}{\sqrt{j}}\right) + 2.0\right) \\ g(\mathbf{x}) = 2 - \sqrt{x_1} + \frac{2}{|J_2|} \sum_{j \in J_2} \left(4 \sum_{j \in J_2} y_j^2 - \prod_{j \in J_2} \cos\left(\frac{20 y_j \pi}{\sqrt{j}}\right) + 2.0\right) \\ f_2(f_1, g, t) = g \cdot \left(1 - \left(\frac{f_1}{g}\right)^{H(t)}\right) \\ where : \\ H(t) = 0.75 \sin(0.5\pi t) + 1.25, \; t = \frac{1}{n_t} \left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ J_1 = \{j \mid \text{j is odd and } 2 \leq j \leq n\} \\ J_2 = \{j \mid \text{j is even and } 2 \leq j \leq n\} \\ y_j = x_j - x_1^{0.5\left(1.0 + \frac{3(j-2)}{n-2}\right)}, \; \forall j = 2, 3, \ldots, n \\ x_1 \in [0, 1]; \; x_i \in [-1, 1]; \; \forall i = 2, 3, \ldots, n \end{cases} \tag{2.40}$$

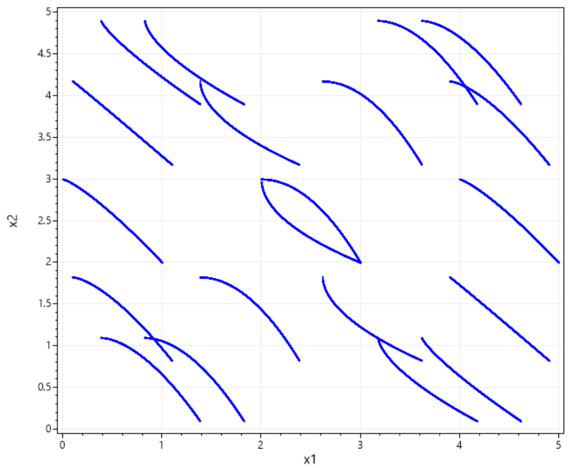
The HE9 also belongs to a Type III environment and its POF is similar to HE7 (refer to Figure 2.8a). The number of decision variables is set to 30 and the POS and POF is given by

$$\begin{aligned} POS &: x_j = x_1^{0.5\left(\frac{3(j-2)}{n-2}\right)}, \; \forall j = 2, 3, \ldots, n \\ POF &: y = (2 - \sqrt{x_1}) \left[1 - \left(\frac{x_1}{2 - \sqrt{x_1}}\right)^{H(t)}\right] \end{aligned} \tag{2.41}$$

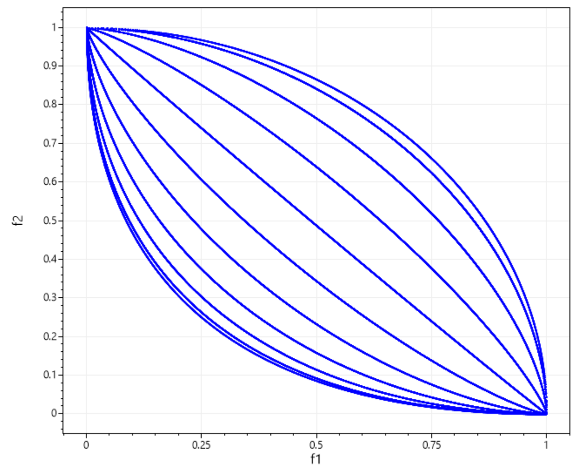
2.2.6 Recently Proposed Test Problems

This section contains recently proposed DMOPs that belong to a Type II environment. The F5-F7 DMOPs were introduced by Zhou, Jin, and Zhang [22] and DF4-DF9 DMOPs were introduced by Jiang et al. [23]. All of the following DMOPs have the number of decision variables, n_x, set to 10.

The F5 DMOP, unlike benchmark functions from the FDA test suite, has non-linear correlation between decision variables and is defined as



(a) POS



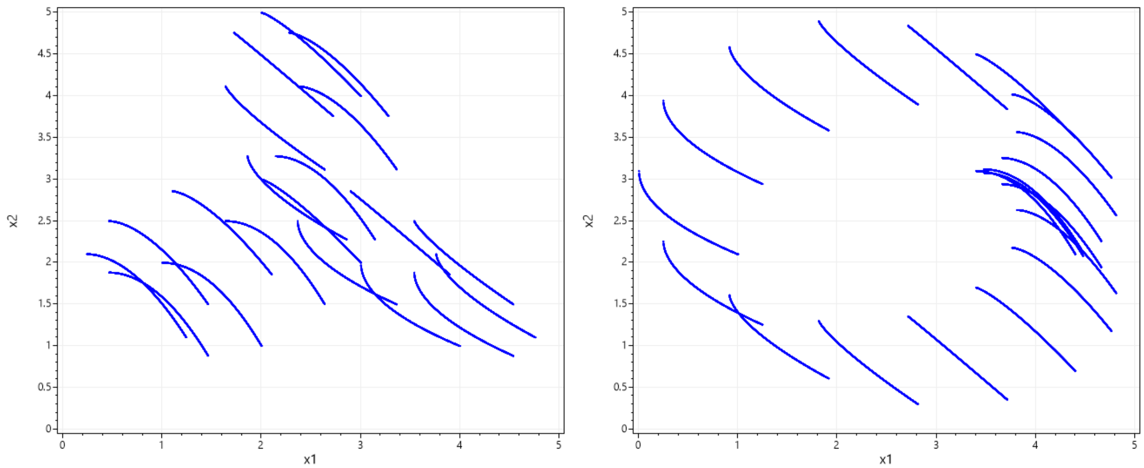
(b) POF

Figure 2.9 The POS and POF of F5 with $n_t = 10$ and $\tau_t = 10$ for 1000 iterations.

$$
\text{F5} = \begin{cases} Minimize : \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x}, t), f_2(\mathbf{x}, t)) \\ f_1(\mathbf{x}, t) = |x_1 - a|^H + \sum_{i \in I_1} y_i^2 \\ f_2(\mathbf{x}, t) = |x_1 - a - 1|^H + \sum_{i \in I_2} y_i^2 \\ where : \\ H(t) = 1.25 + 0.75 \sin(\pi t),\ t = \frac{1}{n_t} \left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ y_i = x_i - b - 1 + |x_1 - a|^{H + \frac{i}{n}} \\ a = 2\cos(\pi t) + 2,\ b = 2\sin(2\pi t) + 2 \\ I_1 = \{i \mid \text{i is odd and } 1 \le i \le n\} \\ I_2 = \{i \mid \text{i is even and } 1 \le i \le n\} \\ x_i \in [0, 5];\ \forall i = 1, 2, \dots, n \end{cases} \tag{2.42}
$$

The POF and POS change over time, as illustrated in Figure 2.9b and 2.9a, respectively. The POF and POS is given by

$$
\begin{aligned} &POS(t) : a \le x_1 \le a + 1,\ x_i = b + 1 - |x_1 - a|^{H + \frac{1}{n}},\ \forall i = 2, \dots, n \\ &POF(t) :\ f_1 = s^H, f_2 = (1 - s)^H, 0 \le s \le 1 \end{aligned} \tag{2.43}
$$



(a) POS of F6 **(b)** POS of F7

Figure 2.10 The POS of F6 and F7 with $n_t = 10$ and $\tau_t = 10$ for 1000 iterations.

$$
\text{F6} = \begin{cases}
Minimize: \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x}, t), f_2(\mathbf{x}, t)) \\
f_1(\mathbf{x}, t) = |x_1 - a|^H + \sum_{i \in I_1} y_i^2 \\
f_2(\mathbf{x}, t) = |x_1 - a - 1|^H + \sum_{i \in I_2} y_i^2 \\
where: \\
H(t) = 1.25 + 0.75 \sin(\pi t),\ t = \frac{1}{n_t} \left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\
y_i = x_i - b - 1 + |x_1 - a|^{H + \frac{i}{n}} \\
a = 2\cos(\pi t) + 2,\ b = 2\sin(2\pi t) + 2 \\
I_1 = \{i \mid \text{i is odd and } 1 \le i \le n\} \\
I_2 = \{i \mid \text{i is even and } 1 \le i \le n\} \\
x_i \in [0, 5];\ \forall i = 1, 2, \ldots, n
\end{cases} \tag{2.44}
$$

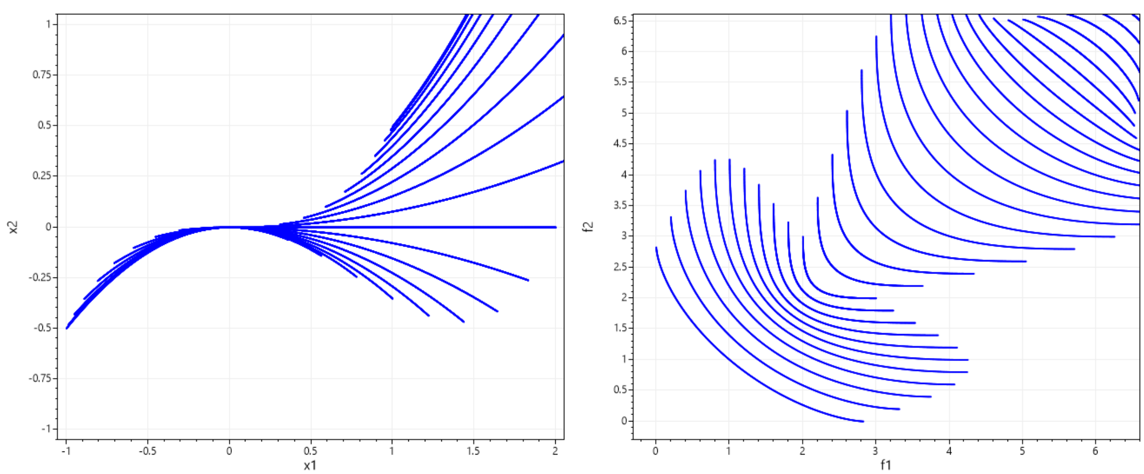
The POF and POS change over time, as illustrated in Figure 2.9b and 2.10a, respectively. The POF and POS is given by

$$
\begin{aligned}
&POS(t): a \le x_1 \le a + 1,\ x_i = b + 1 - |x_1 - a|^{H + \frac{1}{n}},\ \forall i = 2, \ldots, n \\
&POF(t):\ f_1 = s^H, f_2 = (1 - s)^H, 0 \le s \le 1
\end{aligned} \tag{2.45}
$$

$$
\text{F7} = \begin{cases}
Minimize : \mathbf{f}(\mathbf{x},t) = (f_1(\mathbf{x},t), f_2(\mathbf{x},t)) \\
f_1(\mathbf{x},t) = |x_1 - a|^H + \sum_{i \in I_1} y_i^2 \\
f_2(\mathbf{x},t) = |x_1 - a - 1|^H + \sum_{i \in I_2} y_i^2 \\
where : \\
H(t) = 1.25 + 0.75 \sin(\pi t), \;\; t = \frac{1}{n_t} \left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\
y_i = x_i - b - 1 + |x_1 - a|^{H + \frac{i}{n}} \\
a = 2\cos(\pi t) + 2, \;\; b = 2\sin(2\pi t) + 2 \\
I_1 = \{ i \,|\, \text{i is odd and } 1 \le i \le n \} \\
I_2 = \{ i \,|\, \text{i is even and } 1 \le i \le n \} \\
x_i \in [0,5]; \;\; \forall i = 1, 2, \dots, n
\end{cases} \tag{2.46}
$$

The POF and POS change over time, as illustrated in Figure 2.9b and 2.10b, respectively. The POF and POS is given by

$$
\begin{aligned}
&POS(t) : a \le x_1 \le a + 1, \;\; x_i = b + 1 - |x_1 - a|^{H + \frac{1}{n}}, \;\; \forall i = 2, \dots, n \\
&POF(t) : \;\; f_1 = s^H, f_2 = (1 - s)^H, 0 \le s \le 1
\end{aligned} \tag{2.47}
$$



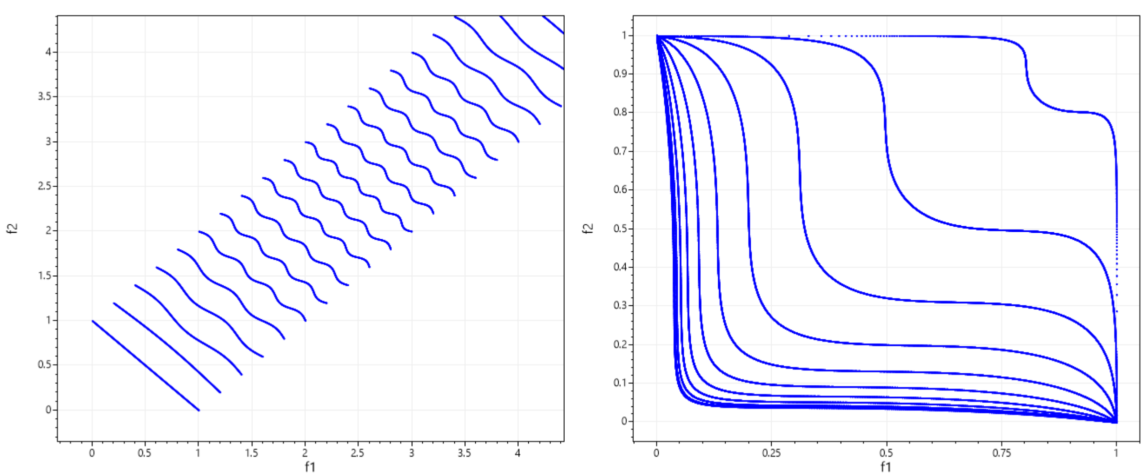
(a) POS **(b)** POF

Figure 2.11 The POS and POF of DF4 with $n_t = 10$ and $\tau_t = 10$ for 1000 iterations.

$$
\text{DF4} = \begin{cases} Minimize : \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x}, g(\mathbf{x}, t), t), f_2(x_1, g(\mathbf{x}, t), t)) \\ f_1(\mathbf{x}, g, t) = g \cdot |x_1 - a|^{H(t)} \\ g(\mathbf{x}, t) = 1 + \sum_{i=2}^{n} \left(x_i - \frac{ax_1^2}{i}\right)^2 \\ f_2(x_1, g, t) = g \cdot |x_1 - a - b|^{H(t)} \\ where : \\ H(t) = 1.5 + a, \ t = \frac{1}{n_t} \left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ a = \sin(0.5\pi t), \ b = 1 + |\cos(0.5\pi t)| \\ x_i \in [-2, 2]; \ \forall i = 1, 2, \ldots, n \end{cases} \tag{2.48}
$$

The DF4 has dynamics on both the POS and POF, as illustrated in Figure 2.11a and in Figure 2.11b. The length and position of the POS changes over time and the POF segments change from convex to concave. The POS and POF is given by

$$
\begin{aligned} POS(t) :& \ a \leq x_1 \leq a + b, \ x_i = \frac{ax_1^2}{i}, \ \forall i = 2, \ldots, n \\ POF(t) :& \ f_2 = \left(b - f_1^{\frac{1}{H(t)}}\right)^{H(t)}, \ 0 \leq f_1 \leq b^{H(t)} \end{aligned} \tag{2.49}
$$



(a) POF of DF5 **(b)** POF of DF6

Figure 2.12 The POF of DF5 and DF6 with $n_t = 10$ and $\tau_t = 10$ for 1000 iterations.

$$\text{DF5} = \begin{cases} Minimize: \mathbf{f}(\mathbf{x},t) = (f_1(\mathbf{x}, g(\mathbf{x},t),t), f_2(x_1, g(\mathbf{x},t),t)) \\ f_1(\mathbf{x},g,t) = g \cdot (x_1 + 0.02\sin(w_t \pi x_1)) \\ g(\mathbf{x},t) = 1 + \sum_{i=2}^{n} (x_i - G(t))^2 \\ f_2(x_1,g,t) = g \cdot (1 - x_1 + 0.02\sin(w_t \pi x_1)) \\ where: \\ G(t) = \sin(0.5\pi t),\ w_t = \lfloor 10G(t) \rfloor,\ t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ x_1 \in [0,1];\ x_i \in [-1,1];\ \forall i = 2,3,\dots,n \end{cases} \tag{2.50}$$

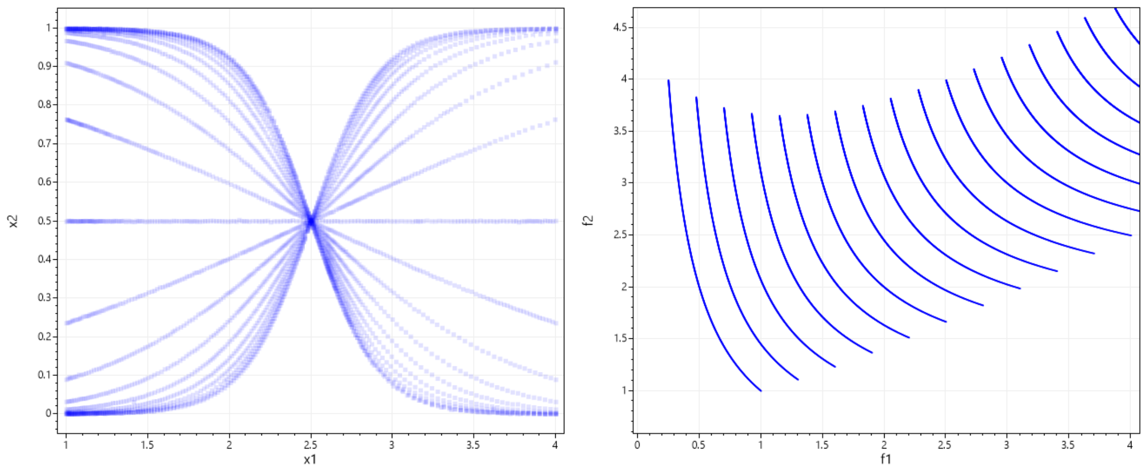
The POS of DF5 is rather simple and is similar to FDA1 (refer to Figure 2.1a). The POF has time-varying curves, as depicted in Figure 2.12a, where it starts linear and then contains several locally concave/convex segments. The POS and POF is given by

$$\begin{aligned} &POS(t):\ a \le x_1 \le 1,\ x_i = G(t),\ \forall i = 2,\dots,n \\ &POF(t):\ f_1 + f_2 = 1 + 0.04\sin\left(w^t \pi \frac{f_1 - f_2 + 1}{2}\right),\ 0 \le f_1 \le 1 \end{aligned} \tag{2.51}$$

$$\text{DF6} = \begin{cases} Minimize: \mathbf{f}(\mathbf{x},t) = (f_1(\mathbf{x}, g(\mathbf{x},t),t), f_2(x_1, g(\mathbf{x},t),t)) \\ f_1(\mathbf{x},g,t) = g \cdot (x_1 + 0.1\sin(3\pi x_1))^{\alpha_t} \\ g(\mathbf{x},t) = 1 + \sum_{i=2}^{n} (|G(t)| y_i^2 - 10\cos(2\pi y_i) + 10) \\ f_2(x_1,g,t) = g \cdot (1 - x_1 + 0.1\sin(3\pi x_1))^{\alpha_t} \\ where: \\ G(t) = \sin(0.5\pi t),\ t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ y_i = x_i - G(t),\ \alpha_t = 0.2 + 2.8|G(t)|, \\ x_1 \in [0,1];\ x_i \in [-1,1];\ \forall i = 2,3,\dots,n \end{cases} \tag{2.52}$$

For DF6, the POF geometry is different from the previous DMOPs as it contains knee regions and long tails. The POS is similar to FDA1 POS, as illustrated in Figure 2.1a, and the POF is depicted in Figure 2.12b. The POS and POF is given by

$$\begin{aligned} &POS(t):\ a \le x_1 \le 1,\ x_i = G(t),\ \forall i = 2,\dots,n \\ &POF(t):\ f_1^{\frac{1}{\alpha_t}} + f_2^{\frac{1}{\alpha_t}} = 1 + 0.2\sin\left(3\pi \frac{f_1^{\frac{1}{\alpha_t}} - f_2^{\frac{1}{\alpha_t}} + 1}{2}\right),\ 0 \le f_1 \le 1 \end{aligned} \tag{2.53}$$



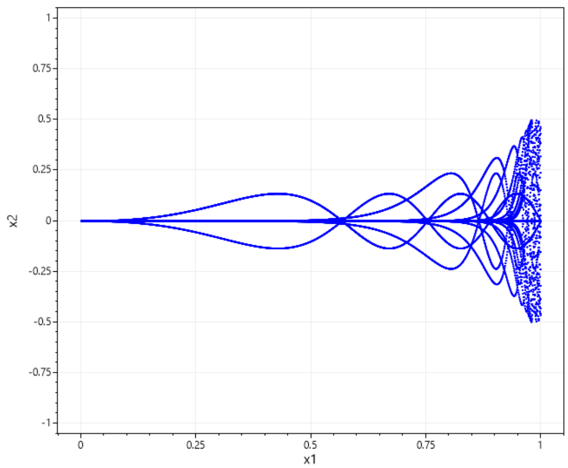
(a) POS **(b)** POF

Figure 2.13 The POS and POF of DF7 with $n_t = 10$ and $\tau_t = 10$ for 1000 iterations.

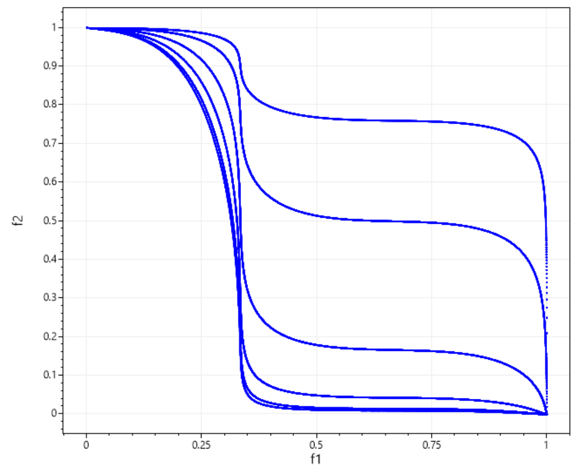
$$\text{DF7} = \begin{cases} Minimize: \mathbf{f}(\mathbf{x},t) = (f_1(\mathbf{x}, g(\mathbf{x},t),t), f_2(x_1, g(\mathbf{x},t),t)) \\ f_1(\mathbf{x},g,t) = g \cdot \left(\frac{1+t}{x_1}\right) \\ g(\mathbf{x},t) = 1 + \sum_{i=2}^{n} \left(x_i - \frac{1}{1+e^{\alpha_t(x_1-2.5)}}\right)^2 \\ f_2(x_1,g,t) = g \cdot \left(\frac{x_1}{1+t}\right) \\ where: \\ \alpha_t = 5\cos(0.5\pi t),\ t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ x_1 \in [1,4];\ x_i \in [0,1];\ \forall i = 2,3,\ldots,n \end{cases} \tag{2.54}$$

The POF of DF7, as depicted in Figure 2.13b, changes over time. The POS, as illustrated in Figure 2.13a, also changes over time but has a centroid that remains unchanged. The POS and POF is given by

$$\begin{aligned} POS(t):&\ 0 \le x_1 \le 1,\ x_i = \frac{1}{1+e^{\alpha_t(x_1-0.5)}},\ \forall i = 2,\ldots,n \\ POF(t):&\ f_2 = \frac{1}{f_1},\ \frac{1+t}{4} \le f_1 \le (1+t) \end{aligned} \tag{2.55}$$



(a) POS



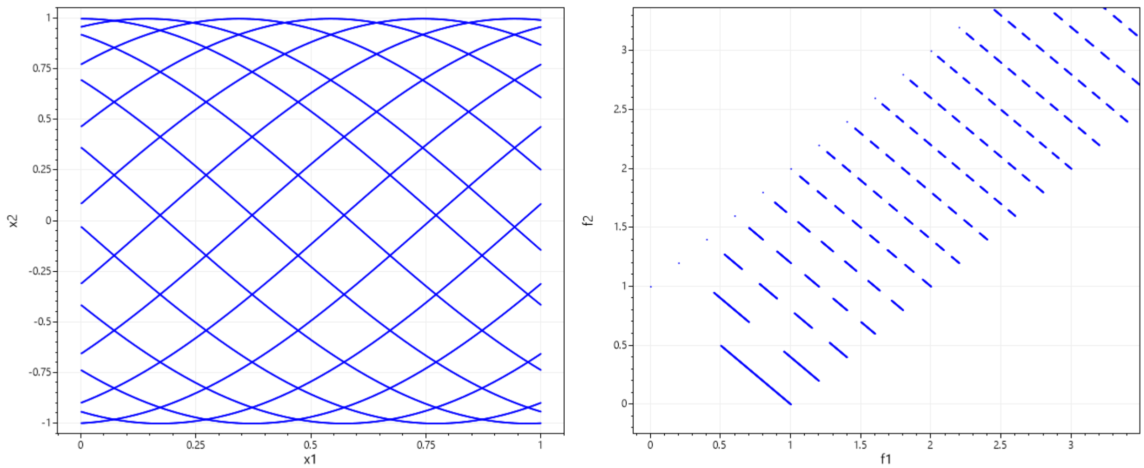
(b) POF

Figure 2.14 The POS and POF of DF8 with $n_t = 10$ and $\tau_t = 10$ for 1000 iterations.

$$\text{DF8} = \begin{cases} Minimize: \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x}, g(\mathbf{x}, t)), f_2(x_1, g(\mathbf{x}, t))) \\ f_1(\mathbf{x}, g) = g \cdot (x_1 + 0.1 \sin(3\pi x_1)) \\ g(\mathbf{x}, t) = 1 + \sum_{i=2}^{n} \left(x_i - \frac{G(t) \sin(4\pi x_1^{\beta_t})}{1+|G(t)|} \right)^2 \\ f_2(x_1, g) = g \cdot (1 - x_1 + 0.1 \sin(3\pi x_1)) \\ where: \\ G(t) = \sin(0.5\pi t), \; t = \frac{1}{n_t} \left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\ \alpha_t = 2.25 + 2\cos(2\pi t), \; \beta_t = 100 G^2(t) \\ x_1 \in [0, 1]; \; x_i \in [-1, 1]; \; \forall i = 2, 3, \dots, n \end{cases} \tag{2.56}$$

Similar to DF7, the POS of DF8 varies over time and has a stationary POS centroid (refer to Figure 2.14a). The POF of DF8, as illustrated in Figure 2.14a, contains knee regions and long tails, which can be a challenging property [24, 25]. The POS and POF is given by

$$\begin{aligned} &POS(t): \; 0 \leq x_1 \leq 1, \; x_i = \frac{G(t) \sin(G(t) \sin(4\pi x_1^{\beta_t}))}{1 + |G(t)|}, \; \forall i = 2, \dots, n \\ &POF(t): \; f_1 + f_2^{\frac{1}{n_t}} = 1 + 0.2 \sin\left(3\pi \frac{f_1 - f_2^{\frac{1}{\alpha_t}} + 1}{2}\right), \; 0 \leq f_1 \leq 1 \end{aligned} \tag{2.57}$$



(a) POS **(b)** POF

Figure 2.15 The POS and POF of DF9 with $n_t = 10$ and $\tau_t = 10$ for 1000 iterations.

$$
\text{DF9} = \begin{cases}
Minimize : \mathbf{f}(\mathbf{x}, t) = (f_1(\mathbf{x}, g(\mathbf{x}, t), t), f_2(x_1, g(\mathbf{x}, t), t)) \\
f_1(\mathbf{x}, g, t) = g \cdot \left(x_1 + max\left\{0, \left(\frac{1}{2N_t} + 0.1\right)\sin\left(2N_t\pi x_1\right)\right\}\right) \\
g(\mathbf{x}, t) = 1 + \sum_{i=2}^{n}\left(x_i - \cos\left(4t + x_1 + x_{i_1}\right)\right)^2 \\
f_2(x_1, g, t) = g \cdot \left(1 - x_1 + max\left\{0, \left(\frac{1}{2N_t} + 0.1\right)\sin\left(2N_t\pi x_1\right)\right\}\right) \\
where : \\
N_t = 1 + \lfloor 10|\sin\left(0.5\pi t\right)|\rfloor,\ t = \frac{1}{n_t}\left\lfloor \frac{\tau}{\tau_t} \right\rfloor \\
x_1 \in [0, 1];\ x_i \in [-1, 1];\ \forall i = 2, 3, \ldots, n
\end{cases} \tag{2.58}
$$

For DF9, the POS and the POF change over time. The POS of DF9 is illustrated in Figure 2.15a, and the POF, as depicted in Figure 2.15b, has a time-varying disconnected segments. The POS and POF is given by

$$
\begin{aligned}
&POS(t):\ 0 \leq x_1 \leq 1,\ x_i = \cos\left(4t + x_1 + x_{i-1}\right),\ \forall i = 2, \ldots, n \\
&POF(t):\ f_2 = 1 - f_1,\ f_1 \in \bigcup_{i=1}^{N_t}\left[\frac{2i-1}{2N_t}, \frac{i}{N_t}\right]\bigcup\{0\}
\end{aligned} \tag{2.59}
$$

2.2.7 Summary of Dynamic Multi-objective Optimisation Problems

This study considers DMOPs that belong only to the first three environment types that were formally defined in Section 2.1.2. All of the twenty-nine DMOPs discussed in this chapter have been classified in Table 2.1, where the number of decision variables, n_x, is given for each DMOP. The number of decision variables was selected based on the suggested values from the original authors of these problems.

Table 2.1 Benchmark functions classified into three dynamic environment types, where n_x is the number of decision variables

Type I	n_x	**Type II**	n_x	**Type III**	n_x
FDA1	20	ZJZ	20	FDA2$_{Cam}$	31
FDA4	12	FDA3$_{Cam}$	30	dMOP1	10
DIMP1	10	FDA5	12	HE1	30
DIMP2	10	FDA5$_{iso}$	12	HE2	30
dMOP3	10	FDA5$_{dec}$	12	HE7	30
dMOP3$_{mod}$	10	dMOP2	10	HE9	30
		dMOP2$_{iso}$	10		
		dMOP2$_{dec}$	10		
		F5-F7	10		
		DF4-DF9	10		

2.3 Dynamic Multi-objective Optimisation Performance Measures

It is useful to have a metric that provides an overall estimate of the performance at the end of the experiment. These metrics, or performance measures, allow for an easy comparison between various algorithms or parameter setting combinations. Based on the analysis of the performance measures by Helbig and Engelbrecht [26], six performance measures were selected for this study: Variational Distance (VD), Spacing (S), Maximum Spread (MS), the number of non-dominated solutions (NS), Accuracy (acc), and Stability ($stab$). They can be categorized into four distinct types, that is

- **Accuracy Performance Measures**: These performance measures are used to measure the accuracy of POF^* found by a DMOO algorithm. They estimate how close the found POF^* is to the true POF' (i.e. VD).

- **Diversity Performance Measures**: These performance measures are used to measure the diversity of the non-dominated solutions contained in POF^*. Diversity can be measured either by measuring how evenly the solutions are spread along POF^* (i.e. S), the extent of POF^* (i.e. MS), or by the number of non-dominated solutions (i.e. NS).
- **Robustness Performance Measures**: These performance measures quantify the robustness of the algorithm (i.e. *stab*). They measure how well the algorithm recovers after the environment change.
- **Combined Performance Measures**: These performance measures assess the overall quality of the non-dominated solutions of the approximated POF^*. They are called combined metrics because they take into account both the accuracy and diversity of the found solutions (i.e. *acc*).

Formal definitions and an in-depth description of each performance measure is given below.

2.3.1 Variational Distance

Zhou et al. [17] adapted the generational distance (GD) for DMOO and dubbed it the variational distance. The VD measures the accuracy of POF^* found by a DMOO algorithm by estimating how close the found POF^* is to the true POF'. The VD is formally defined as:

$$VD = \frac{1}{n_c} \sum_{i=1}^{n_c} VD(t) \tag{2.60}$$

with

$$VD(t) = \frac{\sqrt{n_{POF^*} \sum_{i=1}^{n_{POF^*}} d_i^2(t\%\tau_t)}}{n_{POF^*}}$$

where n_{POF^*} is the number of solutions in POF^* and d_i is the Euclidean distance in the objective space between solution i of POF^* and the nearest member of POF'. The current iteration number is t, τ_t is the frequency of change, and n_c is the number of environment changes. The performance measure is calculated every iteration just before a change in the environment occurs. Prior knowledge of when changes occur is required and the POF^* needs to be normalised.

2.3.2 Number of Non-dominated Solutions

The number of non-dominated solutions in POF^* is most likely the easiest performance measure to calculate, but it should be noted that it does not provide any information with regards to the quality of the solutions. For example, one algorithm might become stuck in a local POF optimum and have a higher NS than the algorithm that successfully tracked the true POF' but has a smaller NS. This could lead to an incorrect conclusion that the algorithm with higher NS value is better. The NS metric can still be useful when used in tandem with other performance measures.

2.3.3 Spacing

The metric of spacing [27] was designed to be used with other performance measures. It has a low computational cost, and provides useful information about how evenly the non-dominated solutions are distributed along POF^* [26]. It is formally defined as:

$$S(t) = \sqrt{\frac{1}{n_{POF^*} - 1} \sum_{i=1}^{n_{POF^*}} (d_i - \overline{d})^2}, \ S = \frac{1}{n_c} \sum_{i=1}^{n_c} S(t) \tag{2.61}$$

with

$$d_i = min_{j=1,\ldots,n_{POF^*};j \neq i} \left\{ \sum_{k=1}^{n_k} |f_k(\mathbf{x}) - f_{kj}(\mathbf{x})| \right\}$$

and

$$\overline{d} = \frac{1}{n_{POF^*}} \sum_{i=1}^{n_{POF^*}} d_i$$

where n_{POF^*} is the number of non-dominated solutions found and d_i is the minimum value of the sum of the absolute difference in objective function values between the i-th solution in POF^* and any other solution in POF^*, and $\overline{d}$ is the average of all d_i values. The lower the S value is, the more uniformly spread the solutions in POF^* are. However, this does not mean that the solutions are necessarily good, since they can be uniformly spaced in POF^*, but not close to the true POF' [10]. The POF^* needs to be normalised prior to the calculation of the S metric.

It should be noted that hypervolume (HV) performance measure is sometimes referred to as S-metric [10]. However, in this paper, the S performance measure refers to the metric of Spacing from Equation 2.61.

2.3.4 Maximum Spread

The maximum spread measure, introduced by Zitzler [28], measures the length of the diagonal of the hyperbox that is created by the extreme function values of the non-dominated solutions in POF^* [26].

Goh and Tan [9] introduced an adapted version of MS for DMOPs to measure how well POF^* covers POF' by taking into account the proximity of POF^* to the POF'. The MS is defined as:

$$MS = \frac{1}{n_c} \sum_{i=1}^{n_c} MS(t) \tag{2.62}$$

with

$$MS(t) = \sqrt{\frac{1}{n_k} \sum_{k=1}^{n_k} \left[\frac{min\{\overline{POF_k^*}, \overline{POF_k'}\} - max\{\underline{POF_k^*}, \underline{POF_k'}\}}{\overline{POF_k'} - \underline{POF_k'}} \right]^2}$$

where $\overline{POF_k^*}$ and $\underline{POF_k^*}$ are the maximum and minimum value of the k-th objective in POF^*, respectively. A high MS' value (i.e. $MS = 1$) indicates a good spread of solutions. However, when the algorithm losses track of the true POF', the MS value might be bigger than one (i.e. $MS \geq 1$), which will rank higher than the algorithm that tracked the true POF' properly and obtained a value closer to 1.

2.3.5 Accuracy

Cámara, Ortega, and Toro [29] introduced the accuracy measure that uses the hypervolume difference (HVD) [17] when the true POF' is known. It is called the alternative accuracy acc_{alt} measure to not confuse it with the acc measure introduced by Weicker [30]. The alternative accuracy measure is formally defined as:

$$acc_{alt} = \frac{1}{n_c} \sum_{i=1}^{n_c} acc_{alt}(t) \tag{2.63}$$

with

$$acc_{alt}(t) = |HV(POF'(t)) - HV(POF^*(t))|$$

where $acc_{alt}(t)$ is the absolute HVD at time t. The absolute values ensure that $acc_{alt}(t) \geq 0$. However, when the true POF' is unknown, the acc_{alt} cannot be used. The mean of acc_{alt} can be calculated by averaging all the computed values by the number of changes n_c. The acc_{alt} will be referred to as acc for the rest of the paper.

2.3.6 Stability

Cámara, Ortega, and Toro [31] adapted the stability measure for DMOPs by quantifying the effect of the changes in the environment on the accuracy of the algorithm. Stability is defined as:

$$stab(t) = max\{0, acc(t-1) - acc(t)\}, \ stab = \frac{1}{n_c}\sum_{i=1}^{n_c} stab(t) \tag{2.64}$$

where a low *stab* value indicates good performance. The stability measure indicates how well the DMOA recovers after an environment change. To calculate the *stab* measure, the accuracy (*acc* defined in Equation 2.63) from the previous environment change has to be known. Therefore, if there are n_c changes, there will be $n_c - 1$ stability measures calculated after the experiment.

2.4 Summary

Many real life problems are dynamic in nature where more than one objectives needs to be optimised. Choosing the right set of benchmark functions to test the performance of the newly proposed DMOAs can be challenging, and to help with this task, researchers have developed many dynamic multi-objective optimisation problems and performance measures. This chapter covered the formal definitions for DMOO and discussed the concepts of the POS and the POF that subsequent chapters build upon. Then, twenty-nine DMOPs and six performance measures that will be used in the experimental section of the thesis were discussed.

Next chapter provides an overview of the particle swarm optimisation algorithm for SOPs and the recently proposed multi-guide particle swarm optimisation algorithm for SMOPs.

Chapter 3

Overview of Particle Swarm Optimisation Algorithms

This chapter covers the particle swarm optimisation (PSO) algorithm for single-objective optimisation and the multi-guide PSO for multi-objective optimisation. Section 3.1 covers the original PSO implementation, followed by the MGPSO implementation in Section 3.2

3.1 Particle Swarm Optimisation for Single-objective Optimisation

The particle swarm optimisation algorithm by Kennedy and Eberhart [32], originally developed for simulating social behavior of bird flocks, was later adapted as an effective computational method for single objective optimisation problems. The PSO algorithm solves problems by maintaining a swarm of particles, where each particle represents a candidate solution, and then iteratively improves the solutions by moving them around the search space. In the original implementation of the PSO, the movement of particles is governed by the particle's personal best (*pbest*) position, and the neighbourhood best (*nbest*) position of the pre-defined neighbourhood that the particle belongs to. In other words, the *pbest* position contains the cognitive information of each particle, and the *nbest* position contains the social information of the neighbourhood. When the neighbourhood of each particle is the entire swarm, then the *nbest* position is referred to as the global best (*gbest*) position. The pseudo-code of the PSO algorithm is given in Algorithm 1.

Algorithm 1 PSO Algorithm

```
create and initialise a swarm
while stopping condition has not been reached do
    for each particle in a swarm do
        set pbest position
    end for
    set nbest position
    for each particle in a swarm do
        calculate new velocity
        calculate new position
    end for
end while
```

Initialising the Swarm

The following process covers the initialisation of the PSO algorithm with inertia weight [33]. The PSO algorithm is initialized with a single swarm that contains n_s particles. Each of the particle's position is then initialised within the domain of the problem under consideration, as in

$$x_j(0) = x_{min,j} + r_j(x_{max,j} - x_{min,j}),\ \forall j = 1, 2, \ldots, n_x \tag{3.1}$$

where $r_j \sim U(0, 1)$, n_x is the number decision variables, $\mathbf{x}$, and x_j is the j-th dimension of $\mathbf{x}$. The $x_{min,j}$ and $x_{max,j}$ refer to the minimum and maximum feasible values in each dimension, j.

The *pbest* of each particle in swarm is set to the initial position of the particle, i.e. $\mathbf{y}_i(0) = \mathbf{x}_i(0)$. The *nbest* values, $\hat{\mathbf{y}}_i(0)$, are then determined by considering all the *pbest* positions of all particles from their respective neighbourhoods, and velocity vector of each particle is set to $\mathbf{v}_i(0) = 0$. The cognitive component (c_1), social component (c_2), and inertia weight (ω) are typically fine-tuned based on the problem under consideration, but $c_1 = c_2 = 1.49$, and $\omega = 0.72$ are common values that lead to a convergent behavior.

Calculating a New Velocity and Position of a Particle

Movement of particles is primarily based on the cognitive component, social component, inertia weight, and the previous position of a particle. Following is the formal definition of a velocity calculation

$$\mathbf{v}_i(t + 1) = \omega\mathbf{v}_i(t) + c_1\mathbf{r}_1(t)[\mathbf{y}_i(t) - \mathbf{x}_i(t)] + c_2\mathbf{r}_2(t)[\hat{\mathbf{y}}_i(t) - \mathbf{x}_i(t)] \tag{3.2}$$

where $\mathbf{v}_i(t)$ and $\mathbf{x}_i(t)$ are the velocity and position of particle i at time step t, respectively; $\hat{\mathbf{y}}_i$ represents the *nbest* of particles $\hat{\imath}$, with particle i in their neighbourhood. The $\mathbf{y}_i(t)$ represents the *pbest* at time t; $c_1\mathbf{r}_1(t)[\mathbf{y}_i(t) - \mathbf{x}_i(t)]$ is the cognitive component and $c_2\mathbf{r}_2(t)[\hat{\mathbf{y}}_i(t) - \mathbf{x}_i(t)]$ is the social component. Both c_1 and c_2 are positive acceleration coefficients that control the influence of the cognitive and social components, respectively. The random vectors, $\mathbf{r}_1$ and $\mathbf{r}_2$ are sampled from a uniform distribution, i.e. $\mathbf{r}_1, \mathbf{r}_2 \sim U(0,1)^{n_x}$. The inertia weight, ω, influences the contribution of the previous flight magnitude on the new velocity.

Once the new velocity of a particle has been calculated, its new position is determined by adding the velocity to its current position, as follows

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}_i(t+1) \tag{3.3}$$

For a boundary constraint problem, there exists a possibility where a particle will move outside the boundaries of the search space, which can be problematic since the true POS can only be found within the bounds of the search space. Therefore, an effective strategy is required to overcome the boundary violations. The following are some of the common strategies that have been identified by Chu, Gao, and Sorooshian [34] and Engelbrecht [35] to address this issue. The first strategy involves an absorption technique where the dimension of the particle that violates the constraint is set to the boundary of that dimension. Another basic strategy involves generating a random value within the bounds of the feasible values for the dimension that violated the boundary constraint. The above techniques can be used by any DMOA, but there is a strategy that can only be used by a PSO-based algorithm, called *pbest* selection. In *pbest* selection, the particle is allowed to exit the feasible region, but *pbest* updates are disallowed when the position is infeasible.

Calculating Personal Best and Neighbourhood Best Positions

For minimisation problems, the *pbest* at time $t+1$ is calculated as

$$\mathbf{y}_i(t+1) = \mathbf{x}_i(t+1) \;\text{ if } f(\mathbf{x}_i(t+1)) < f(\mathbf{y}_i(t)) \tag{3.4}$$

where the fitness function is represented by $f: \mathbb{R}^{n_x} \rightarrow \mathbb{R}$. The *nbest* is calculated by computing the *pbest* found so far by all particles in the neighborhood. Then, *nbest* is going to be replaced by the value that corresponds to a minimum, m, among all of the *pbest* of particles in that neighborhood, as long as m is smaller than the current *nbest* value. Furthermore, the *pbest* and *nbest* values can be updated in either *synchronous* or *asynchronous* manner [36]. In *synchronous* updates, the *nbest* positions are updated after the *pbest* of every particle is calculated, whereas with *asynchronous* updates, the *nbest* is calculated after each particle's *pbest* update.

Topologies in a PSO are used to allow communication between particles. They can be viewed as social networks that, depending on the implementation, can control the flow of information between particles. The *gbest* and *lbest* implementations are described below, but there exist other variants of neighborhood topologies and the reader is referred to [37, 38, 39, 40] for more information and experimentation on the topologies used in a PSO. In a *gbest* PSO, all particles are connected to one another with a *star* topology, where each particle can see the *pbest* of the every particle in the swarm. In a local best (*lbest*) topology, each particle is divided into a predefined number of neighbourhoods where the particle can only communicate with its neighbors. However, in *lbest* topology a particle can still be a part of more than one neighbourhood. The main implication of the *lbest* topology is that there is a slower information exchange compared to the *star* topology. Some of the notable examples of the PSO with *lbest* topologies include the Von Neumann or *ring* topologies.

The other topologies and PSO variants are out of scope of this thesis. The reader is referred to [35, 41] for a more thorough overview of the various networks used in a PSO as well as the other variants of the PSO algorithm.

3.2 Multi-guide Particle Swarm Optimisation for Multi-objective Optimisation

The multi-guide particle swarm optimisation (MGPSO) algorithm [11], as illustrated in Algorithm 2, makes use of multiple swarms and assigns each objective function to a corresponding subswarm. The quality of particles in a subswarm is evaluated using the objective function assigned to that subswarm. Similar to the inertia weight PSO by Shi and Eberhart [33], the personal best particle (*pbest*) and the neighbourhood best particle (*nbest*) positions within a subswarm are also updated based on the corresponding objective function. However, the MGPSO expands the velocity update equation to include an archive guide to find solutions that balance the objective

Algorithm 2 Multi-guide Particle Swarm Optimisation [11]

for each objective $k = 1, \ldots, n_k$ **do**
    Let f_k be the objective function;
    Create and initialise a swarm, S_k, to contain n_{s_k} particles;
    for each particle $i = 1, \ldots, n_{s_k}$ **do**
        Initialise position $\mathbf{x}_{ki}(0)$ uniformly within a predefined hypercube of dimension n_x;
        Initialise the *pbest* position as $\mathbf{y}_{ki}(0) = \mathbf{x}_{ki}(0)$;
        Determine the *nbest*, $\hat{\mathbf{y}}_{ki}(0)$;
        Initialise the velocity as $\mathbf{v}_{ki}(0) = \mathbf{0}$;
        Initialise $\lambda_{ki} \sim \mathrm{U}(0, 1)$;
    end for
end for
Let $t = 0$;
repeat
    for each objective $k = 1, \ldots, n_k$ **do**
        for each particle $i = 1, \ldots, n_{s_k}$ **do**
            if $f_k(\mathbf{x}_{ki}(t)) < f_k(\mathbf{y}_{ki}(t))$ **then**
                $\mathbf{y}_{ki}(t) = \mathbf{x}_{ki}(t)$;
            end if
            for particles $\hat{i}$ with particle i in their neighborhood **do**
                if $f_k(\mathbf{y}_{ki}(t)) < f_k(\hat{\mathbf{y}}_{k\hat{i}}(t))$ **then**
                    $\hat{\mathbf{y}}_{k\hat{i}}(t) = \mathbf{y}_{ki}(t)$;
                end if
            end for
            Update the archive with the solution $\mathbf{x}_{ki}(t)$;
        end for
    end for
    for each objective $k = 1, \ldots, n_k$ **do**
        for each particle $i = 1, \ldots, n_{s_k}$ **do**
            Select a solution, $\hat{\mathbf{a}}_{ki}(t)$, from the archive using tournament selection;

$$\begin{aligned}\mathbf{v}_{ki}(t+1) = \omega \mathbf{v}_{ki}(t) &\\ &+ c_1 \mathbf{r}_1(t)(\mathbf{y}_{ki}(t) - \mathbf{x}_{ki}(t)) \\ &+ \lambda_{ki} c_2 \mathbf{r}_2(t)(\hat{\mathbf{y}}_{ki}(t) - \mathbf{x}_{ki}(t)) \\ &+ (1 - \lambda_{ki}) c_3 \mathbf{r}_3(t)(\hat{\mathbf{a}}_{ki}(t) - \mathbf{x}_{ki}(t));\end{aligned}$$

            $\mathbf{x}_{ki}(t+1) = \mathbf{x}_{ki}(t) + \mathbf{v}_{ki}(t+1)$;
        end for
    end for
    $t = t + 1$;
until stopping condition is true

functions. The MGPSO velocity update equation is defined as

$$\begin{aligned}\mathbf{v}_{ki}(t+1) = &\ \omega\mathbf{v}_{ki}(t) + c_1\mathbf{r}_1(t)(\mathbf{y}_{ki}(t) - \mathbf{x}_{ki}(t)) \\ &+ \lambda_{ki}c_2\mathbf{r}_2(t)(\hat{\mathbf{y}}_{ki}(t) - \mathbf{x}_{ki}(t)) \\ &+ (1-\lambda_{ki})c_3\mathbf{r}_3(t)(\hat{\mathbf{a}}_{ki}(t) - \mathbf{x}_{ki}(t))\end{aligned} \tag{3.5}$$

where $\mathbf{v}_{ki}(t)$ and $\mathbf{x}_{ki}(t)$ are the velocity and position of particle ki at time step t respectively; $\hat{\mathbf{y}}_{ki}$ represents the *nbest* of neighbourhood $\hat{\imath}$ and $\mathbf{y}_{ki}(t)$ represents the *pbest* at time t; $c_1\mathbf{r}_1(t)[\mathbf{y}_{ki}(t) - \mathbf{x}_{ki}(t)]$ is the cognitive component of the velocity and $c_2\mathbf{r}_2(t)[\hat{\mathbf{y}}_{ki}(t) - \mathbf{x}_{ki}(t)]$ is the social component of the velocity; c_1 and c_2 are positive acceleration coefficients that influence the contributions of the cognitive and social components respectively; $\mathbf{r}_1, \mathbf{r}_2 \sim U(0,1)^{n_x}$ are vectors of random values sampled from an uniform distribution with n_x representing the number of decision variables or the dimension of the search space [42]; ω represents the inertia weight, which controls the influence of previous step sizes and direction on the new velocity.

The MGPSO introduced the following parameters to the velocity equation: c_3 is an archive acceleration coefficient, $\mathbf{r}_3$ is a random vector, where each element of $\mathbf{r}_3$ is sampled from an uniform distribution in $(0,1)$; $\hat{\mathbf{a}}_{ki}(t)$ is the archive guide for a particle ki, sampled from the archive at iteration t; and λ_{ki} is the archive balance coefficient for particle ki. The control parameter, λ_{ki} balances the influence of the social and archive guides on the particle's velocity and thus the amount of exploitation of the already found candidate solutions in POF. Smaller λ_{ki} values increase the influence of the archive guide while simultaneously decreasing the influence of the social/neighbourhood guide. Scheepers, Engelbrecht, and Cleghorn [11] proposed that random values be sampled for the archive balance coefficient to increase the stochasticity of the search process and to eliminate the need for tuning the archive balance coefficient.

The archive guide, $\hat{\mathbf{a}}_{ki}(t)$, is selected for each particle i in each subswarm k using the following procedure: Three non-dominated solutions are randomly selected from the archive, and the solution with the largest crowding distance [43] in the archive is selected as the archive guide. In other words, the least populated solution is selected to diversify sparsely populated regions along the found POF.

Once the new velocity of a particle has been calculated, its new position is determined by adding the velocity to its current position as follows

$$\mathbf{x}_{ki}(t+1) = \mathbf{x}_{ki}(t) + \mathbf{v}_{ki}(t+1) \tag{3.6}$$

3.2.1 Bounded Archive

The MGPSO uses a bounded archive [44] that limits the number of solutions that can be kept in the archive. The pseudo-code for the bounded archive maintenance is provided in Algorithm 3

When a particle is considered for addition to the archive, which happens right after the evaluation of the particle as in Algorithm 2, the archive must check for the dominance relation between the particles already in the archive and the new particle that is being considered for addition to the archive. If the new particle is dominated by any of the solutions in the archive, it is discarded. Otherwise, the particle is added to the archive, and any archive solution that becomes dominated is removed from the archive. If the archive is full, crowding distance [43] is used to determine the most crowded solution in the archive, which is then removed. This process ensures that only non-dominated solutions are stored in the archive.

Algorithm 3 Bounded Archive Maintenance

```
for each particle, p, in swarm do
    if p is not dominated by any solution in the archive
            and p is not similar to any solutions in the archive then
        if p dominates any solution in the archive then
            Remove all archive solutions that are dominated by p;
            Add p to the archive;
        else
            Add p to the archive;
            if archive size exceeds the archive size limit then
                Select the most crowded solution, a, from the archive;
                Remove solution a from the archive;
            end if
        end if
    end if
end for
```

3.3 Summary

This chapter defined the original PSO algorithm used for single-objective optimisation and the multi-guide PSO used for multi-objective optimisation. In the next chapter, several components to adapt the multi-guide particle swarm optimisation algorithm for dynamic multi-objective optimisation are proposed.

Chapter 4

Proposed Multi-guide Particle Swarm Optimisation for Dynamic Multi-objective Optimisation

This thesis presents an adapted multi-guide particle swarm optimization algorithm capable of solving dynamic multi-objective optimization problems. The MGPSO, proposed by Scheepers, Engelbrecht, and Cleghorn [11], is a multi-swarm approach, where each subswarm optimizes one of the objectives. It takes advantage of the bounded archive to keep track of the changing Pareto optimal front (POF) of non-dominated solutions and adds an archive guide to the velocity update equation to facilitate convergence to a new POF. This work extends the original algorithm by introducing six archive management update approaches and eight alternative archive balance coefficient initialization strategies. Moreover, two recently proposed quantum PSO algorithms, self-adaptive QPSO [45] and parent centric crossover QPSO [46], are incorporated into the MGPSO to deal with the diversity loss.

Section 4.1 introduces the archive update strategies to re-initialize the archive when the change to the environment occurs. Section 4.2 covers eight alternative initialization approaches to set the archive balance coefficient parameter. Self-adaptive QPSO and parent centric corssover QPSO and are discussed in Section 4.3. Lastly, Section 4.4 contains the summary of the chapter.

4.1 Archive Management

The objective of this section is to provide an overview of the environment changes that occur in DMOO and then to develop effective environment change mechanisms for the bounded archive used by the MGPSO. Specifically, Section 4.1.1 describes in more detail the consequences of changes in objective functions for DMOO and Section 4.1.2 proposes six new archive management update strategies.

4.1.1 Consequences of the Environment Changes

One of the main goals of DMOAs is to develop an effective and computationally efficient environment change strategy so that the DMOAs can quickly track the ever-changing Pareto-optimal set and Pareto-optimal front. Dynamic environments are challenging because DMOAs have limited time to explore and exploit the decision space. It is very likely that when a change to the environment occurs, the new true POS and POF will be different from the last true POS and POF, and so the algorithm needs to adapt quickly to these changes. The extent of changes to both POS and POF depend on the type of the environment that the DMOP belongs to. The DMOPs and different environment types are described in Section 2.2.

One question remains: How is the change to the environment detected? In most real-world scenarios, changes to the environment are usually visible with the use of sensors, agents, etc. However, when the exact time of changes is unknown, an environment change detection strategy is required. The most common strategy is to re-evaluate 10% of the population as in Deb, Rao, and Sindhya [2], Goh and Tan [9], and Zhang et al. [47], but this strategy may fail if the change did not occur in any of the selected individuals. Other strategies do exist, but this paper does not take into account any of the approaches since the exact time of changes is assumed to be known beforehand.

4.1.2 Proposed Archive Management Approaches

The archive guide, as described in Section 3.2, is selected from the non-dominated solution in the bounded archive. Since the original implementation of the MGPSO was developed for SMOPs, the algorithm did not have to worry about any environment change approaches. The following six archive management update approaches are considered, namely

Clear the archive

When the environment changes, the archive is completely cleared from all solutions. This approach is clearly inefficient since the solutions in the archive may not become dominated, or could be repaired with little effort to become non-dominated. In other words, the algorithm looses valuable information about the objective space. For instance, the MGPSO algorithm uses the velocity update equation to govern the movement of the particles towards better areas of the search space. As in Equation (3.5), one of the main components in the calculation is the archive guide. So, by clearing the archive, none of the particles will use the information from the archive to improve the results since all previously gained knowledge about the potentially good areas of the decision space is now lost. This approach is referred to as **cl** for the rest of the paper.

Re-evaluate solutions

When a change in the environment is detected, some of the solutions in the archive that were non-dominated might have become dominated since DMOPs change over time. This means that there has to be an effective archive management strategy implemented to remove dominated solutions and to keep the non-dominated solutions in the archive. This adapted MGPSO uses the following archive management strategy: When a change to the environment occurs, re-evaluate all of the archive's solutions and remove the solutions that became dominated. This strategy is referred to as **re** for the rest of the paper.

Local search with a fixed step size

When a change is detected, all solutions in the archive are optimised through a hill-climbing process where successor solutions are created for each particle in the archive. These new solutions are then added to the archive and all of the particles in the archive are re-evaluated for dominance. This process repeats four times to keep the computational complexity of the algorithm low. Furthermore, preliminary results have shown that the MGPSO performs well when the local search repeats four times. The pseudo-code for the local search is provided in Algorithm 4.

The successor solutions are created as follows: Let $\mathbf{x}_{\mathsf{I}}$ represent a vector of decision variables used in a calculation of the first objective, and $\mathbf{x}_{\mathsf{II}}$ represent a vector of decision variables used in a calculation of the other objective(s). Then, successor solutions are generated, for each solution in the archive, by moving decision variables in $\mathbf{x}_{\mathsf{I}}$ and $\mathbf{x}_{\mathsf{II}}$ by a % of the domain's extent, i.e. a % of $(\mathbf{x}_{max} - \mathbf{x}_{min})$, as follows

- First successor: Add $\mathbf{s}_{\mathsf{I}}$ to $\mathbf{x}_{\mathsf{I}}$ and leave $\mathbf{x}_{\mathsf{II}}$ unchanged
- Second successor: Subtract $\mathbf{s}_{\mathsf{I}}$ from $\mathbf{x}_{\mathsf{I}}$ and leave $\mathbf{x}_{\mathsf{II}}$ unchanged

- Third successor: Leave $\mathbf{x_I}$ unchanged and add $\mathbf{s_{II}}$ to $\mathbf{x_{II}}$
- Fourth successor: Leave $\mathbf{x_I}$ unchanged and subtract $\mathbf{s_{II}}$ from $\mathbf{x_{II}}$
- Fifth successor: Add $\mathbf{s_I}$ to $\mathbf{x_I}$ and add $\mathbf{s_{II}}$ to $\mathbf{x_{II}}$
- Sixth successor: Add $\mathbf{s_I}$ to $\mathbf{x_I}$ and subtract $\mathbf{s_{II}}$ from $\mathbf{x_{II}}$
- Seventh successor: Subtract $\mathbf{s_I}$ from $\mathbf{x_I}$ and add $\mathbf{s_{II}}$ to $\mathbf{x_{II}}$
- Eight successor: Subtract $\mathbf{s_I}$ from $\mathbf{x_I}$ and subtract $\mathbf{s_{II}}$ from $\mathbf{x_{II}}$

where $\mathbf{s_I}$ and $\mathbf{s_{II}}$ are vectors representing a % of the domain's extent for decision variables in $\mathbf{x_I}$ and $\mathbf{x_{II}}$, respectively. The percentages selected for this study are 2%, 5%, and 10%. These approaches are referred to as $\mathbf{h_2}$, $\mathbf{h_5}$, and $\mathbf{h_{10}}$ for the rest of the paper.

The rationale behind this algorithm is that hill-climbing is known for quickly finding better solutions when the current solutions are bad. In most cases, this is exactly what happens when an environment change occurs. The previously found POF will not be close to the new true POF, and so hill-climbing will re-populate the archive with more feasible solutions.

Algorithm 4 Local Search

```
repeat
    Initialize successors, S, to an empty set
    Copy the bounded archive to A
    Clear the bounded archive
    for each particle, p, in A do
        Generate eight successor solutions using the process described in Section 4.1.2
        Add the new successors to S
    end for
    Add each successor to A using Algorithm 3
until stopping condition is true
```

Local search with a decreasing step size

This approach is similar to the one defined in the previous section. However, a % of the domain's extent is not fixed anymore. During the first iteration, it is set to 10% and then the value is decreased by a factor of 2 with each iteration. The implication of this strategy is that the successor solutions will initially move large distances in the the search space, then gradually slow down until the last iteration is finished. The idea is to combine the strengths of the $\mathbf{h_2}$, $\mathbf{h_5}$, and $\mathbf{h_{10}}$ approaches into a single strategy. This hill-climbing archive management strategy is referred to as $\mathbf{h_d}$ for the rest of the paper.

4.1.3 Worst-case Computational Complexity Analysis

This section discusses the worst-case computational complexity of the different archive management strategies.

The original MGPSO used a bounded archive size equal to the total number of particles, $n_s = n_k n_{s_k}$. With reference to Section 3.2.1, the worst-case computational complexity of the original archive management strategy is calculated as the cost to evaluate the dominance relation and the cost of the crowding distance calculation [43]:

$$O(n_s n_k) + O(n_k n_s \log n_s) = O(n_k n_s \log n_s) \tag{4.1}$$

This cost is incurred for each particle considered for addition to the archive.

The archive management strategies proposed in Section 4.1.2 add an additional cost at each environment change. This additional cost for each archive management strategy is as follows:

- Clear the archive: $O(n_s)$, because the only additional action is to delete all of the solutions that are currently in the archive.
- Re-evaluate solutions: $O(n_s^2 n_k)$, due to the evaluation of the dominance relation for each pair of solutions in the archive.
- Local search strategies: Let n_l be the number of local search iterations and let n'_a be the number of successors created for each solution in the archive. Then both local search strategies have a cost of $O(n_l n_s n'_a + n'_a n_k n_s \log n_s)$. However, note that small constant values are used for the local search parameters, i.e. $n_l = 4$ and $n'_a = 8$, in which case the cost becomes $O(n_k n_s \log n_s)$.

The re-evaluate strategy is therefore the most costly, followed by the local search strategies. Clear the archive has linear cost in the size of the archive.

Next section covers the current λ_i initialization strategies and then introduces alternative strategies adapted for DMOPs.

4.2 Balance Coefficient

The objective of this section is to provide an overview of the balance coefficient initialization approaches from the previous studies and then to introduce alternative initialization approaches adapted for dynamic multi-objective optimisation. Specifically, Section 4.2.1 describes in more detail the original λ_i initialization approach and then explores other strategies developed by Erwin and Engelbrecht [48]. Section 4.2.2 introduces new initialization approaches adapted for DMOO.

4.2.1 Current Balance Coefficient Initialization Approaches

The initial implementation of the MGPSO used only one way to initialize the balance coefficient (λ_i) parameter. The λ_i is a critical part of the update velocity calculation from Equation 3.5 as it controls the weighted contribution trade-off between the social guide and the archive guide.

The standard approach (*std*) to deal with the balance coefficients is to randomly initialize λ_i, per particle as a constant, that is

$$\lambda_i(0) \sim U(0,1) \tag{4.2}$$

The following five approaches to initialize the λ_i parameter were recently proposed by Erwin and Engelbrecht [48]. The goal of the study was to determine if the alternative initialization strategies result in a more diverse POF.

Random Update (r)

The random strategy samples a new balance coefficient at every iteration. This value is used by all particles. Thus, every particle will have the same balance coefficient at each iteration, that is

$$\lambda_i(t) = \lambda_i(t) \sim U(0,1) \tag{4.3}$$

Random Update Per Particle (r_i)

A different λ_i assigned to every particle, re-sampled at every iteration, as follows

$$\lambda_i(t) \sim U(0,1) \tag{4.4}$$

Random Update Per Particle Per Dimension (r_{ij})

A different balance coefficient, λ_{ij} is assigned to each dimension for every particle, re-sampled at every iteration

$$\lambda_{ij}(t) \sim U(0,1) \tag{4.5}$$

This is the most stochastic strategy since every decision variable of the particle used in the velocity update calculation is initialized differently.

Linearly Decreasing (ld)

For every particle, the balance coefficient is initialized to 1.0. Thereafter, λ_i updated as follows

$$\lambda_i(t+1) = \lambda_i(t) - \frac{1.0}{n} \tag{4.6}$$

where n is the max number of iterations. When λ_i is set to 1.0, the archive component has no influence over the movement of particles. The implication of this strategy is that with each iteration, the archive component is slowly increasing the control of the way particles move.

Linearly Increasing (li)

Similar to the linearly decreasing strategy, but the balance coefficient is initialized to 0.0 and increases linearly over time:

$$\lambda_i(t+1) = \lambda_i(t) + \frac{1.0}{n} \tag{4.7}$$

The implication of this strategy is that initially, the archive component has a very large influence of the way particles move, but with each iteration it becomes less important.

It was determined that, for bi-objective SMOPs, the linearly increasing (li) strategy significantly outperformed other initialization approaches [48]. The MGPSO was able to reach high levels of diversity for all of the tested MOO benchmark functions early on. As a consequence, this allowed the algorithm to exploit the feasible search areas more efficiently as the social guide's influence increased.

4.2.2 Balance Coefficient Initialization Approaches Adapted for DMOPs

The standard, random initialization strategy was shown to lead to a good performance [11] on static multi-objective optimization problems, and a more thorough sensitivity analysis on initialization strategies was conducted by Erwin and Engelbrecht [48] where it was determined that there are better ways to initialize λ_i parameter.

Based on the λ_i sensitivity analysis from [48], three alternative approaches are introduced that take into account the changing behaviour of DMOPs, namely:

Standard Approach, Re-initialized Every Environment Change (std_{τ_t})

The standard update approach (std), initializes the balance coefficient for every particle once during the initialization phase. The following change adapts the parameter for dynamic problems by re-sampling it again

$$\lambda_i(t) \sim U(1, 0) \tag{4.8}$$

at time t right after the environment change has occured.

Linearly Decreasing, Re-initialized Every Environment Change (ld_{τ_t})

For every particle, the balance coefficient is initialized to 1.0. Thereafter, λ_i is updated as follows

$$\lambda_i(t+1) = \lambda_i(t) - \frac{1.0}{\tau_t} \tag{4.9}$$

where τ_t is the frequency of change that controls how often the environment changes. After the change to the environment is detected, the balance coefficient is re-initialized to 1.0.

Linearly Increasing, Re-initialized Every Environment Change (li_{τ_t})

Similar to the linearly decreasing strategy, but the balance coefficient is initialized to 0.0 and increases linearly over time

$$\lambda_i(t+1) = \lambda_i(t) + \frac{1.0}{\tau_t} \tag{4.10}$$

After the change to the environment is detected, the balance coefficient is re-initialized to 0.0. Both ld, and li were adapted because the environment changes every τ_t

iterations, so it makes more sense to linearly decrease or increase λ_i with regards to the frequency of change rather than the maximum iteration number.

The last two approaches had to be adapted for dynamic environments because linearly decreasing/increasing the archive balance coefficient over the maximum iteration number could result in a low performance when it comes to exploiting the feasible regions of the search space. If decreasing strategy is picked, the archive balance coefficient will have very little influence on the velocity update equation during the first half of the run. This might work for static MOPs, but for dynamic MOPs, the changes happen often and the archive guide needs to contribute to the movement of the particle so that it can effectively exploit the search space.

Next section covers the original QPSO implementation and describes six QPSO variants to be added into the MGPSO.

4.3 Quantum Particle Swarm Optimization Strategies

This section discusses quantum particle swarm optimisation algorithm in detail and introduces its two variants. Specifically, Section 4.3.1 describes the original QPSO implementation, followed by self-adaptive QPSO in Section 4.3.2 and parent-centric crossover QPSO in Section 4.3.3. Both of these QPSO approaches are incorporated into the MGPSO to determine whether they can increase its performance when solving DMOPs. Lastly, the alternative sampling methods for both QPSO strategies are proposed in Section 4.3.4.

4.3.1 Original Quantum Particle Swarm Optimization

The original PSO was never intended to solve dynamic multi-objective optimisation problems and, as such, several modifications have been made to the original implementation to help with this task. Some of the most notable examples of PSO variants capable of solving DMOPs include DVEPSO [10], QPSO [49, 46, 45], and MGPSO described in this paper [11]. To this day, original quantum PSO by Blackwell and Branke [49] remains a popular choice, as it is easy to understand and has a low computational complexity. The QPSO converts a percentage of vanilla particles, referred to as quantum particles, to move in a manner similar to electrons orbiting the nucleus of an atom. These quantum particles use a position update that is different to the one from Equation 3.3. Instead, they are sampled from a probability distribution, centered at *nbest* in a following way

$$\mathbf{x}_i(t+1) \sim pd(\hat{\mathbf{y}}_i, r_{cloud}) \tag{4.11}$$

where *pd* is a probability distribution and r_{cloud} is a constant determining the size of the quantum radius within which a quantum particle is allowed to move. Large r_{cloud} values correspond to a large quantum cloud, allowing for more exploration of the decision space. On the other hand, small r_{cloud} values make the quantum radius smaller, which results in more exploitation. However, finding the perfect r_{cloud} value is impossible due to the dynamic nature of the problems being solved. For example, small r_{cloud} restricts the exploration ability to that radius value, whereas large r_{cloud} could result in unnecessary exploration due to the search space being covered by a large cloud. The probability distribution can also be thought as an additional parameter, since many different distributions exist that can be used as a sampling method. Any particle that is not updated using the above equation is called a neutral particle and follows the standard position calculation from Equation 3.3.

4.3.2 Self-adaptive Quantum Particle Swarm Optimization

To mitigate the issue of a problem dependent tuning of the r_{cloud} parameter, Pamparà and Engelbrecht [45] proposed a self-adapting QPSO that automatically adapts the r_{cloud}. The quantum cloud radius is calculated by taking the maximum between the diversity of neutral and quantum particles, as follows

$$D(t) = \frac{1}{n_s} \sum_{i=1}^{n_s} \sqrt{\sum_{j=1}^{n_x} (x_{ij}(t) - \overline{x}_j(t))^2} \tag{4.12}$$

where n_s is the number of the neutral, or quantum particles considered in the diversity calculation; the particle's decision vector must be within the problem domain. The average j-th dimension of the entire swarm, $\overline{x}_j(t)$, is calculated as

$$\overline{x}_j(t) = \frac{\sum_{i=1}^{n_s} x_{ij}(t)}{n_s} \tag{4.13}$$

The resulting diversity value is then used as the r_{cloud} value. The r_{cloud} value is fed into a random distribution as the deviation, from which quantum particle positions are sampled.

The main idea of this approach is to first start with a large r_{cloud} value and then reduce it to a smaller one over time to favour exploitation of the feasible regions of the search space. However, when the change to the environment occurs, r_{cloud} value automatically becomes larger - allowing for more exploration of the decision space by the quantum particles.

4.3.3 Parent-centric Crossover Quantum Particle Swarm Optimization

Harrison, Ombuki-Berman, and Engelbrecht [46] examined the use of a parent-centric crossover (PCX) in PSO as a way to deal with the diversity loss when solving DMOPs. The PCX operator [50] is a multi-parent crossover that creates offspring solutions centered around the parents; even more so towards the parent selected for mutation. Offspring solutions are generated by selecting $n_\mu \geq 3$ parents and computing their mean, $\overline{\mathbf{x}}(t)$. With equal probability, one of the n_μ parents is selected for mutation, denoted $\mathbf{x}_i$, and a direction vector is calculated as

$$\mathbf{d}_i(t) = \mathbf{x}_i(t) - \overline{\mathbf{x}}(t) \tag{4.14}$$

where $\overline{\mathbf{x}}(t)$ is the mean,

$$\overline{\mathbf{x}}(t) = \sum_{l=1}^{n_\mu} \mathbf{x}_l(t) \tag{4.15}$$

of the n_μ parents. For each of the remaining n_μ - 1 parents, perpendicular distances to the line $\mathbf{d}_i(t)$ are calculated, δ_l, and their average, $\overline{\delta}$, is taken. Offspring are then generated according to

$$\tilde{\mathbf{x}}_i(t) = \mathbf{x}_i(t) + N(0, \sigma_1^2)|\mathbf{d}_i(t)| + \sum_{l=1, l\neq i}^{n_\mu - 1} N(0, \sigma_2^2)\overline{\delta}\mathbf{e}_l(t) \tag{4.16}$$

where $\overline{\delta}\mathbf{e}_l(t)$ are the n_μ - 1 orthonormal basis vectors that space the subspace perpendicular to $\mathbf{d}_i(t)$, and σ_1 and σ_2 are deviations of two Gaussian distributions. Thus, offspring solutions are generated by mutating the selected parent based on the distance of the selected parent to the mean of all the parents and the distance that each other parent is from the direction vector, $\mathbf{d}_i(t)$.

Then, for each quantum particle, the new position is determined by

$$\mathbf{x}_i(t+1) = \tilde{\mathbf{x}}_i(t) \tag{4.17}$$

Thus, the resulting algorithm completely removes the atom metaphor, replacing it with a crossover operator instead. While the PCX QPSO removes the r_{cloud} and the probability distribution, *pd*, parameters of the original QPSO, it introduces two additional parameters: σ_1 and σ_2.

4.3.4 Alternative Sampling Methods for QPSO

Both self-adaptive and parent-centric crossover QPSOs use *nbest* position in a calculation of a new quantum particle position. For PCX QPSO, the randomly selected parent, and the parent corresponding to the *nbest* position were considered by Harrison, Ombuki-Berman, and Engelbrecht [46]. A comprehensive analysis of both approaches indicates that PCX QPSO with a *nbest* parent outperforms the randomly selected parent PCX version. Therefore, only the *nbest* approach was selected for this study and will be referred to as PCX_n for the rest of the paper. As for the self-adaptive QPSO, only the original strategy of the QPSO, where the quantum particle is sampled from a probability distribution centered at *nbest*, was considered by Pamparà and Engelbrecht [45]. It will be referred to as $QPSO_n$ for the rest of the paper.

Since the MGPSO introduced the bounded archive and the archive guide, $\hat{\mathbf{a}}_i$, this paper proposes two alternative sampling methods for both QPSO variants that also use $\hat{\mathbf{a}}_i$ in the calculation of a new quantum particle position. Specifically, a randomly selected archive guide and the tournament selected archive guide from the bounded archive is considered. For the randomly selected archive guide, each archive particle has equal probability of being selected. Self-adaptive and PCX QPSOs that use this strategy will be referred to as $QPSO_r$ and PCX_r, respectively, for the rest of the paper. Tournament selected archive guide uses the same procedure as the one explained in Section 3.2 used by neutral particles. The self-adaptive and PCX QPSOs that use this method will be referred to as $QPSO_t$ and PCX_t, respectively, for the rest of the paper.

For $QPSO_r$ and $QPSO_t$, the following change to the Equation 4.11 is made

$$\mathbf{x}_i(t+1) \sim pd(\hat{\mathbf{a}}_i, r_{cloud}) \tag{4.18}$$

where $\hat{\mathbf{a}}_i$ is the archive guide. Another change is made to the diversity calculation from Equation 4.12. This time, the quantum cloud radius is calculated using the particles from the bounded archive, rather than neutral or quantum particles from their respective swarms.

For PCX_r and PCX_t, the archive guide, $\hat{\mathbf{a}}_i$, is selected from $n_\mu \geq 3$ parents for mutation. The selected parent, denoted as $\mathbf{x}_i$, is used in the direction vector calculation the same way as in Equation 4.14. The n_μ parents are populated using particles from the bounded archive and their mean is calculated as in Equation 4.15.

4.4 Summary

This chapter proposed several changes to the original MGPSO to adapt it for dynamic multi-objective optimisation. Six archive management update approaches, eight balance coefficient initialization strategies, and six QPSO techniques were introduced into the MGPSO to allow efficient tracking of the changing true pareto-optimal front. To determine the most promising strategies when solving DMOPs, an in-depth statistical analysis is required.

Next chapter covers the experimental set-up on which the subsequent chapters build upon.

Chapter 5

Experimental Set-up

This chapter discusses the experimental set-up of the experiments conducted in this study. Section 5.1 covers the experiments on archive management update approaches, balance coefficient initialization strategies, and QPSO techniques. The DMOAs and their respective parameter settings are discussed in Section 5.2. The benchmark functions and performance measures used to evaluate the performance of the MGPSO and other DMOAs are covered in Sections 5.3 and 5.4 respectively. Section 5.5 provides an overview of the statistical methods and the ranking algorithm used to evaluate the performance of DMOAs. Lastly, the summary is given in Section 5.6.

5.1 Conducted Experiments

Three experiments are conducted in this study, that is

- **Archive Management Update Approaches:** The archive guide, selected from the bounded archive, controls the amount of influence that the archive guide has on the particle's velocity and thus the amount of exploitation of the already found POF. Whenever a change to the environment occurs, the particles in the archive need to be updated to allow efficient tracking of the POF. In this experiment, six archive management update approaches from Section 4.1, are evaluated against twenty-nine DMOPs across five n_t-τ_t combinations. The best performing approach is then used in a comparative study against other DMOAs.
- **Balance Coefficient Initialization Strategies:** The λ_i is the archive balance coefficient that balances the influence of the social and archive guides on the particle's velocity. In this experiment, nine archive balance coefficient initialization strategies, as defined in Section 4.2, are evaluated against twenty-nine

DMOPs across five n_t-τ_t combinations. The best performing strategy is then used in a comparative study against other DMOAs.

- **Self-adaptive QPSO and PCX QPSO Strategies:** In this experiment, six QPSO techniques (refer to Section 4.3) are evaluated against twenty-nine DMOPs across five n_t-τ_t combinations. These strategies convert a portion of neutral particles into quantum particles to help the MGPSO deal with the diversity loss. Both 10% and 50% proportion of quantum particles are analysed in their respective experiments and then compared with the MGPSO without any quantum particles. The goal is to determine which QPSO strategy works best and to see how the MGPSO behaves when the proportion of quantum particles is high and low.

The main goal of these experiments is to find the strategy that performs best across various DMOPs and environment types. The strategies that have the best overall performance in their respective experiments are then selected for the final comparative studies between other *state-of-the-art* DMOAs.

5.2 DMOO Algorithms and Parameter Settings

The best strategies from their respective experiments are compared with the baseline MGPSO as well as other popular DMOAs that represent different classes of meta-heuristics. The DMOAs selected for this study are: the dynamic version of NSGA-II (DNSGA-II) [2], SGEA [8], and DMOES [47]. To ensure the comparisons are conducted fairly, each DMOA has a population size set to 100 so that an equal number of fitness evaluations are performed by each algorithm. Performance measures from Section 2.3 are computed, right after the environment change has occurred, using 1000 and 900 true Pareto-optimal solutions uniformly generated for bi-objective and tri-objective DMOPs, respectively. No parameter tuning has been done, because such tuned parameters will be optimal for only the first environment. This decision is supported by the findings in [51] and [52]. Below is a short description of each DMOA and the parameter settings are given for each algorithm.

5.2.1 MGPSO

The adapted MGPSO algorithm described in this paper. All of the 100 particles are evenly distributed among swarms, i.e. 50-50 split for bi-objective and 34-33-33 split for tri-objective DMOPs. When the proportion of quantum particles is at 50%, there are 25-25 neutral particles and 25-25 quantum particles for bi-objective DMOPs, and 17-17-17 neutral particles and 17-16-16 quantum particles for tri-objective DMOPs. When the proportion of quantum particles is at 10%, there are 45-45 neutral particles and 5-5 quantum particles for bi-objective DMOPs, and 30-30-30 neutral particles and 4-3-3 quantum particles for tri-objective DMOPs. The cognitive, social and archive components are set to $c_1 = 0.1$, $c_2 = 0.02$, and $c_3 = 1.8$. Inertia weight is set to $\omega = 0.6$. These values were selected based on the preliminary investigations of c_1, c_2, c_3, and ω parameters when solving the problem set. The preliminary results indicated that the archive was the most influential component when approximating the POF, as it resulted in a more diverse and accurate POF when compared with MGPSO where c_3 was set to a lower value. On the other hand, large cognitive and social values were detrimental when solving the problem set and it was determined that a value in the range $[0.02, 0.1]$ for both c_1 and c_2 worked best. One explanation as to why a large influence of the archive component works well when solving DMOPs is that solutions found in the bounded archive, at time t, can be considered the best non-dominating solutions found so far by the MGPSO. Therefore, it makes sense to guide the rest of the particles towards trade-off solutions from the archive - it will result in the archive being populated more quickly when compared with the strategy where particles are guided primarily by the cognitive and social guides. The archive guide, $\hat{\mathbf{a}}_i$ is selected for each particle i using tournament selection with tournament size of 3. The MGPSO uses a local best (*l-best*) topology of size 3 with asynchronous *nbest* updates [53], where the *nbest* of the neighbourhood is updated right after the particle that belongs to the neighbourhood has been evaluated. As an environment change reaction strategy, neutral particle's *pbest* values are reset to the current particle's position, then re-evaluated, and quantum particles are re-initialized within the problem domain since they do not use *pbest* in the position update calculation [45]. The balance coefficient, λ_i, is randomly initialized from a uniform distribution in the range $(0, 1)$. Balance coefficient values are re-sampled right after the environment change has occurred. Better performance due to re-sampling during the search process is confirmed by a study of Erwin and Engelbrecht [48] for SMOPs. The iteration number of the local search strategies is set to 4.

5.2.2 DNSGA-II

This dynamic version of the popular NSGA-II algorithm by Deb et al. [43] is a representative of Pareto-dominance based multi-objective evolutionary algorithms (MOEAs). Deb, Rao, and Sindhya [2] adapted NSGA-II for DMOO by replacing some population members with either randomly created solutions or mutated solutions of existing solutions whenever environment change is detected. The mutated version has been selected for this study as it shows slightly better results than the random approach [2]. DNSGA-II does not use a bounded archive. The parameters of DNSGA-II are as follows: simulated binary crossover probability of 0.9, polynomial mutation probability of $1/n_x$ (where n_x is the number of decision variables), $\zeta = 30$, and distribution indices for crossover and mutation are 10 and 4, respectively. The population size is set to 100.

5.2.3 SGEA

Jiang and Yang [8] developed the steady-state and generational evolutionary algorithm (SGEA) for DMOO that combines the fast and steady tracking ability of steady-state algorithms and good diversity preservation of generational algorithms. If a change is detected, SGEA reuses a portion of the outdated solutions with good distribution and relocates a number of solutions close to the new POF based on the information collected from previous environments and the current environment. SGEA incorporates a bounded archive to store non-dominated solutions. The parameters of SGEA are as follows: simulated binary crossover probability of 1.0, polynomial mutation probability of $1/n_x$ (where n_x is the number of decision variables), $\zeta = 80$, and distribution indices for crossover and mutation are 20 and 20, respectively [8]. The population size is set to 100.

5.2.4 DMOES

More recently, Zhang et al. [47] proposed an evolution strategy based evolutionary algorithm, called DMOES, that has been shown to efficiently and effectively solve DMOPs. DMOES uses four self-adaptive precision-controllable mutation operators designed for individuals to explore and exploit the decision space. Simulated isotropic magnetic particle niching guides the individuals to keep a uniform distance and extent to approximate the entire POF automatically. Moreover, the non-dominated solution guided immigration facilitates the population convergence with two different strategies for the non-dominated solutions and the dominated solutions, respectively. DMOES does not use a bounded archive. The population size is set to 100 and precision-controllable mutation parameters p and q are set to 3 and 1, respectively [47]. The population size is set to 100.

5.3 Benchmark Functions

Based on the analysis of DMOPs in Section 2.2, twenty-nine benchmark functions were selected to evaluate the performance of the MGPSO between various archive management update approaches, balance coefficient initialization strategies, and QPSO techniques. The benchmark set includes: FDA1 [13], ZJZ [17], FDA2$_{Cam}$ and FDA3$_{Cam}$ [18], FDA4 and FDA5 [13], FDA5$_{iso}$ and FDA5$_{dec}$ [12], DIMP1 and DIMP2 [20], dMOP1-dMOP3 [9], dMOP3$_{mod}$ [23], dMOP2$_{iso}$ and dMOP2$_{dec}$ [12], HE1 and HE2 [21], HE7 and HE9 [12], F5-F7 [22], and DF4-DF9 [23].

The dynamic environment DMOP types, formally defined in Section 2.1.2, are summarized in Table 5.1. All of the chosen DMOPs belong only to the first three environment types and the reader is referred to the Section 2.2.7 where the summary of each DMOP is provided in Table 2.1.

Table 5.1 Dynamic environment types for DMOO problems

	POS	
POF	No Change	Change
No Change	Type IV	Type I
Change	Type III	Type II

For each benchmark function, the following severity of change (n_t) and frequency of change (τ_t) combinations were used: $n_t = 10$ and $\tau_t = 10$, $n_t = 10$ and $\tau_t = 25$, $n_t = 10$ and $\tau_t = 50$, $n_t = 1$ and $\tau_t = 10$, and $n_t = 20$ and $\tau_t = 10$. Both the spatial severity and temporal severity has been defined in Section 2.1.2.

Each DMOP is optimised over 1000 iterations and no pre-emptive termination of the algorithm was used by any DMOA considered in this study. Each n_t-τ_t combination, as well as the number of changes ($n_c = 1000/\tau_t$) that was used for each DMOP, is presented in Table 5.2.

Table 5.2 The n_t-τ_t combinations with regards to the spatial and temporal severity

Severity of Change	**Frequency of Change**	n_t	τ_t	n_c
Medium	Fast	10	10	100
Medium	Medium	10	25	40
Medium	Slow	10	50	20
Big	Fast	1	10	100
Small	Fast	20	10	100

5.4 Performance Measures

Based on the analysis of performance measures in Section 2.3, six performance measures were selected for this study, namely

- The number of non-dominated solutions (NS) in the found POF.
- Variational distance [17]. Low VD value indicates good performance with regards to how closely the found POF is to the true POF.
- Spacing of Schott [27]. Low S value indicates that the solutions are evenly spread along the found POF.
- Maximum spread [9]. High MS value (close to one) indicates good spread of solutions.
- The alternative accuracy measure, acc_{alt} [29], referred to as *acc* in this study. Low *acc* value indicates good performance.
- Stability [31] that quantifies the effect of changes in the environment on *acc* of the DMOA. Low *stab* value indicates good performance.

5.5 Evaluating the Performance

Section 5.5.1 provides definitions of the statistical tests that are used in the calculation of wins and losses between DMOAs. The ranking algorithm is covered in Section 5.5.2.

5.5.1 Statistical Tests

Mann-Whitney U Test

In statistics, the two-tailed Mann Whitney U test is a non-parametric test performed on exactly two samples. The median for both samples is calculated and compared to determine if there is a significant difference between them. The null and alternative hypothesises are

$$\begin{aligned} H_0&\colon \text{ Median } 1 = \text{Median } 2 \\ H_1&\colon \text{ Median } 1 \neq \text{Median } 2 \end{aligned} \tag{5.1}$$

If a 95% confidence is used (i.e. $\alpha = 0.05$) and the test fails to reject the null hypothesis (i.e. p-value $\geq \alpha$), it means that there is no statistically significant difference between the two algorithms. Otherwise, the algorithms differ from one another and the severity of the differences should be explored along with all of the contributing factors. If more than two algorithms are present, then a Mann-Whitney U test is not applicable; in such case, the Kruskal-Wallis test is a more appropriate alternative.

Kruskal-Wallis Test

This test is a nonparametric alternative to the ANOVA test. The procedure is performed to compare the medians between every sample and to determine whether samples differ from each other or not. The null and alternative hypothesises are

$$\begin{aligned} H_0&\colon \text{ Median } 1 = \text{Median } 2 = ... = \text{Median } k \\ H_1&\colon \text{ At least one pair Median } i \neq \text{ Median } j \end{aligned} \tag{5.2}$$

If the test fails to reject the null hypothesis, it means that the samples in all groups are equal (or similar to each other). Otherwise, at least one median of the group differs from a median in another group. It is important to note that the Kruskal-Wallis test provides no information as to which one of the samples differ from the rest.

5.5.2 Ranking Algorithm

Performance evaluation of the DMOAs is usually done by averaging the performance measures obtained at each time step, right before the change to the environment occurs. The overall rank is calculated for each algorithm, and the one with the best performance obtained against various DMOPs is ranked the highest. However, this approach does not take into account the tracking ability of the algorithm because the averaged metric looses that information. It does not say how well the algorithm adapts to the changes and might lead to incorrect conclusions. For the above reasons, Helbig and Engelbrecht [54] proposed a different approach to ranking DMOAs and the general calculation of wins and losses that is performed for each performance measure is presented in Algorithm 5.

Algorithm 5 Calculation of wins and losses [54]

```
for each DMOP do
    for each n_t-τ_t combination do
        for each pm do
            perform Kruskal-Wallis tests on pm
            if statistical significant difference then
                for each DMOA-pair do
                    perform Mann-Whitney U test on pm
                    if statistical significant difference then
                        assign wins and losses
                    end if
                end for
            end if
        end for
        calculate Diff for the pm
    end for
    calculate Diff for the n_t-τ_t combination
end for
calculate Diff for the DMOP
```

For each DMOP, n_t-τ_t combination and the performance measure, pm, the Kruskal-Wallis test is used to determine if there is a significant difference with regards to pm between DMOAs. If the test fails to reject the null hypothesis, this means that there is at least one DMOA for which results differ from the other DMOAs. If so, for each DMOA-pair, the two-tailed Mann-Whitney U test is performed to determine which algorithm performed better, and the ranking algorithm assigns the wins and losses accordingly. The *Diff* is the difference between the number of wins and the number of losses assigned: $Diff = \#wins - \#losses$. The DMOA with the highest *Diff* value is assigned a rank of 1, and the one with the lowest *Diff* value ranks the last.

This approach takes into account the tracking ability of the algorithm, because rather than assigning the wins and losses on the overall averaged pm value (averaged across all the environment changes), the pm values are compared between DMOAs at each time step t just before a change in the environment occurs. The pm values for each environment change are averaged over 30 independent runs to make the results statistically significant.

To ensure that a DMOA that tracks the changing POF very well for a DMOP does not lead to skewed results, the number of wins and losses are normalised as follows [54]:

$$\begin{aligned} \#wins_{norm} &= \frac{\#wins}{\#changes} \\ \#losses_{norm} &= \frac{\#losses}{\#changes} \end{aligned} \tag{5.3}$$

where $\#changes$ represents the number of changes that occurred during the entire run.

5.6 Summary

This chapter covered the experimental set-up on which the subsequent chapters build upon. Section 5.1 provided an overview of each experiment performed in this study. The DMOO algorithms and their respective parameter settings were discussed in Section 5.2. Section 5.3 covered twenty-nine benchmark functions as well as five severity of change and frequency of change combinations. Performance measures and the ranking algorithm used to evaluate the performance of DMOAs was discussed in Section 5.4 and 5.5, respectively.

Next chapter conducts detailed analysis of the experimental results for archive management update approaches.

Chapter 6

Archive Management Experiments

The purpose of this section is to go over all the experiments done in this study and to perform an extensive analysis of the results. Firstly, six archive management approaches for all DMOP types are compared in Section 6.1. Then, the results are broken down into each of the three environment types considered in this study. Section 6.2 discusses type I DMOPs, Section 6.3 covers type II DMOPs, and an analysis of the type III DMOPs is provided in Section 6.4. The goal of these experiments is to find the archive management approach that performs best across all of the twenty-nine benchmark functions and then use it in a comparative study with the other *state-of-the-art* DMOAs in Section 6.5. Performance measures used in the experiments are S, VD, MS, acc, $stab$, and NS as defined in Section 2.3. When the table with results says that $pm = all$, it means that the results of the wins and losses calculations for these six performance measures are combined in the calculation of ranks.

6.1 Results for all DMOP Types

This section compares the six archive management strategies over all the DMOP types, with the goal to identify the best archive management strategy to use in the later comparisons with state-of-the-art DMOAs. Table 6.1 displays the overall results by the various archive management approaches. It is evident that the hill climber with the decreasing step size ($\mathbf{h_d}$) outperformed the other approaches by a large margin. All the variants of the hill climbing algorithm performed better than the re-evaluation of non dominated solutions (**re**) approach and the archive clearing (**cl**) approach. While the evaluation algorithm takes into account the tracking ability of the algorithms, Table 6.1 contains only the overall results - which makes it hard to assess the performance of each approach with regards to the various frequencies and severities of change, as well as the six performance measures. Table 6.2 and Table 6.3 breaks down the results into overall wins and losses by the frequencies and severities of change, and by the performance measures, respectively.

Table 6.1 Overall wins and losses by the various archive management approaches for all performance measures and n_t-τ_t combinations

Results	**Archive Management Approaches**					
	cl	**re**	$\mathbf{h_2}$	$\mathbf{h_5}$	$\mathbf{h_{10}}$	$\mathbf{h_d}$
Wins	190.51	251.47	853.08	838.62	832.01	1168.79
Losses	1336.75	1096.16	455.3	439.84	566.42	240.01
Diff	-1146.24	-844.69	397.78	398.78	265.59	928.78
Rank	6	5	3	2	4	**1**

Table 6.2 Overall wins and losses for various frequencies and severities of change across all performance measures

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**Archive Management Approaches**					
				cl	**re**	$\mathbf{h_2}$	$\mathbf{h_5}$	$\mathbf{h_{10}}$	$\mathbf{h_d}$
10	10	all	Wins	50.65	70.92	279.27	273.27	229.23	391.69
10	10	all	Losses	432.05	358.57	141.27	128.51	178.94	55.69
10	10	all	Diff	-381.4	-287.65	138	144.76	50.29	336
10	10	all	Rank	6	5	3	2	4	**1**
10	25	all	Wins	24.54	22.3	94.13	91.97	80.11	136.38
10	25	all	Losses	145.25	127.07	50.05	46.45	60.95	19.66
10	25	all	Diff	-120.71	-104.77	44.08	45.52	19.16	116.72
10	25	all	Rank	6	5	3	2	4	**1**
10	50	all	Wins	11.52	10.59	38.74	38.72	33.61	60.77
10	50	all	Losses	59.93	53.83	23.96	20.55	26.91	8.77
10	50	all	Diff	-48.41	-43.24	14.78	18.17	6.7	52
10	50	all	Rank	6	5	3	2	4	**1**
1	10	all	Wins	54.8	76.44	129.61	175.47	300.49	203.38
1	10	all	Losses	270.4	209.88	148.45	110.26	93.2	108
1	10	all	Diff	-215.6	-133.44	-18.84	65.21	207.29	95.38
1	10	all	Rank	6	5	4	3	**1**	2
20	10	all	Wins	49	71.22	311.33	259.19	188.57	376.57
20	10	all	Losses	429.12	346.81	91.57	134.07	206.42	47.89
20	10	all	Diff	-380.12	-275.59	219.76	125.12	-17.85	328.68
20	10	all	Rank	6	5	2	3	4	**1**

Table 6.3 Overall wins and losses for various performance measures across all n_t-τ_t combinations

$\mathbf{n_t}$	τ_t	**PM**	**Results**	**Archive Management Approaches**					
				cl	**re**	$\mathbf{h_2}$	$\mathbf{h_5}$	$\mathbf{h_{10}}$	$\mathbf{h_d}$
all	all	*S*	Wins	40.24	53.09	161.34	147.62	140.92	224.07
all	all	*S*	Losses	248.51	184.71	83.64	86.93	112.92	50.57
all	all	*S*	Diff	-208.27	-131.62	77.7	60.69	28	173.5
all	all	*S*	Rank	6	5	2	3	4	**1**
all	all	*VD*	Wins	49.4	49.65	182.23	183.23	170.01	256.49
all	all	*VD*	Losses	278.8	238.09	102.87	94.27	132.1	44.88
all	all	*VD*	Diff	-229.4	-188.44	79.36	88.96	37.91	211.61
all	all	*VD*	Rank	6	5	3	2	4	**1**
all	all	*MS*	Wins	18.56	20.85	63.24	74.02	84.48	92.25
all	all	*MS*	Losses	120.8	105.3	41.32	28.27	32.25	25.46
all	all	*MS*	Diff	-102.24	-84.45	21.92	45.75	52.23	66.79
all	all	*MS*	Rank	6	5	4	3	2	**1**
all	all	*acc*	Wins	22.62	47.77	222.98	214.14	214.28	308.94
all	all	*acc*	Losses	346.2	287.54	109.96	109.15	137.36	40.52
all	all	*acc*	Diff	-323.58	-239.77	113.02	104.99	76.92	268.42
all	all	*acc*	Rank	6	5	2	3	4	**1**
all	all	*stab*	Wins	53.21	63.83	108.37	103.66	106.38	126.04
all	all	*stab*	Losses	151.22	115.92	71.1	74.68	83.54	65.03
all	all	*stab*	Diff	-98.01	-52.09	37.27	28.98	22.84	61.01
all	all	*stab*	Rank	6	5	2	3	4	**1**
all	all	*NS*	Wins	6.48	16.28	114.92	115.95	115.94	161
all	all	*NS*	Losses	191.22	164.6	46.41	46.54	68.25	13.55
all	all	*NS*	Diff	-184.74	-148.32	68.51	69.41	47.69	147.45
all	all	*NS*	Rank	6	5	3	2	4	**1**

With regards to Table 6.2, it can be concluded that while $\mathbf{h_5}$ and $\mathbf{h_d}$ performed better when $\mathbf{n_t} = 10$, both approaches were unable to adapt to severe changes to the environment ($\mathbf{n_t} = 1$) as well as $\mathbf{h_{10}}$. The $\mathbf{h_{10}}$ approach was the clear winner here, because of the big step size that allowed each decision variable to move a large distance during the search process. When small changes occurred in the environment ($\mathbf{n_t} = 20$), $\mathbf{h_2}$ and $\mathbf{h_d}$ outperformed the other approaches since each decision variable moved a small distance, making it unlikely to overshoot the feasible area of the search space. The ranks received by each approach is the same for various τ_t values when $\mathbf{n_t} = 10$, which shows that all of the approaches improve at the same rate as the frequency of changes increases. Overall, $\mathbf{h_d}$ was the most balanced approach because it moved each particle in the archive over a large distance in the beginning, then gradually slowed down to allow more effective exploitation of the feasible area. In short, the decreasing hill climber approach was capable of solving problems efficiently across different environment types.

Table 6.3 breaks down the results by each performance measure. Both the **cl** and **re** approaches were inferior to the hill climbing variants, with **re** ranking second last and **cl** ranking last. The reason behind these results is that while MGPSO has very strong exploitation capabilities, it struggled to explore the feasible areas of the search space efficiently. Dynamic environments are challenging because DMOAs have limited time to explore and exploit the decision space. It is very likely that when a change to the environment occurs, the new true POF will be different from the last found POF, and so the MGPSO needs to adapt quickly. That is precisely why the local search approach provided a steep increase in performance. A hill climber is known for finding better solutions quickly when the current solutions are bad, which is exactly the case when the environment changes. Going back to the results, it is evident that local search-based approaches were able to find new feasible regions of the search space and re-populated the bounded archive. The MGPSO then used one of the particles in the archive in the velocity update equation to move the other particles towards these better solutions. In short, a hill climber allowed the MGPSO to immediately exploit the feasible region rather than waste time looking for it.

6.2 Type I DMOP Results

A type I environment is the one where the POS changes over time, but the POF remains unchanged. Comparing the overall results from Table 6.4 to the results across all DMOP types in Table 6.1, it is clear that the hill climbing approaches consistently outperformed the **cl** and **re** approaches. It is evident that $\mathbf{h_d}$ was the winner, with $\mathbf{h_5}$ receiving a rank of 2 and $\mathbf{h_{10}}$ being slightly worse to $\mathbf{h_5}$. These results for the local search approaches were different from the results in Table 6.1, where $\mathbf{h_{10}}$ performed significantly worse compared to the $\mathbf{h_2}$ and $\mathbf{h_5}$ approaches. One of the reasons can be attributed to the fact that these type I benchmark functions change drastically in the decision space right after the environment change occurs and the hill climber needs to move large distances to find the new feasible areas. It is more clearly depicted in Table 6.5 where the severity of change had a big impact on which hill climbing variant performed best.

Table 6.4 Overall wins and losses by the various archive management approaches for type I DMOPs across all performance measures and n_t-τ_t combinations

Results	**Archive Management Approaches**					
	cl	**re**	$\mathbf{h_2}$	$\mathbf{h_5}$	$\mathbf{h_{10}}$	$\mathbf{h_d}$
Wins	19.07	25.23	140.54	157.5	171.78	191.63
Losses	235.01	221.77	85.28	55.86	73.25	34.58
Diff	-215.94	-196.54	55.26	101.64	98.53	157.05
Rank	6	5	4	2	3	**1**

Table 6.5 Overall wins and losses by various frequencies and severities of change for type I DMOPs across all performance measures

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**Archive Management Approaches**					
				cl	**re**	$\mathbf{h_2}$	$\mathbf{h_5}$	$\mathbf{h_{10}}$	$\mathbf{h_d}$
10	10	all	Wins	7.34	10.08	49.87	55.4	47.96	64.94
10	10	all	Losses	79.29	73.39	28.53	16.71	27.22	10.45
10	10	all	Diff	-71.95	-63.31	21.34	38.69	20.74	54.49
10	10	all	Rank	6	5	3	2	4	**1**
10	25	all	Wins	1.94	2.32	17.43	18.07	16.18	25.12
10	25	all	Losses	27.9	27.76	8.85	5.85	8.7	2
10	25	all	Diff	-25.96	-25.44	8.58	12.22	7.48	23.12
10	25	all	Rank	6	5	3	2	4	**1**
10	50	all	Wins	0.8	1.03	7.19	7.4	6.33	10.69
10	50	all	Losses	11.15	11.06	3.89	2.66	3.79	0.89
10	50	all	Diff	-10.35	-10.03	3.3	4.74	2.54	9.8
10	50	all	Rank	6	5	3	2	4	**1**
1	10	all	Wins	4.26	4.46	10.62	27	61.03	26.33
1	10	all	Losses	36.18	35.48	30.84	13.46	2.71	15.03
1	10	all	Diff	-31.92	-31.02	-20.22	13.54	58.32	11.3
1	10	all	Rank	6	5	4	2	**1**	3
20	10	all	Wins	4.73	7.34	55.43	49.63	40.28	64.55
20	10	all	Losses	80.49	74.08	13.17	17.18	30.83	6.21
20	10	all	Diff	-75.76	-66.74	42.26	32.45	9.45	58.34
20	10	all	Rank	6	5	2	3	4	**1**

With regards to Table 6.6, it can be concluded that $\mathbf{h_{10}}$ approach performed best for the MS and *stab* performance measures. The MS indicates how well the solutions are spread out and *stab* measures how well the DMOA recovers after the environment change. The $\mathbf{h_5}$ approach outperformed $\mathbf{h_{10}}$ and $\mathbf{h_2}$ for the VD and *acc* measures. The $\mathbf{h_d}$ outperformed the other approaches for the S, VD, *acc*, and NS measures.

The S metric measures how evenly the non-dominated solutions are distributed along the found POF. The VD measures the accuracy of the found POF against the true POF, and acc is computed by taking the absolute hypervolume difference at time t. It should be noted that when DMOA has a high NS value, these non-dominated solutions are not necessarily better in terms of accuracy. However, in this case $\mathbf{h_d}$ was not only more accurate, but also found more solutions. The **cl** and **re** approaches were vastly inferior, which is not surprising since local search approaches take more function evaluations.

Table 6.6 Overall wins and losses by various performance measures for type I DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	τ_t	**PM**	**Results**	**Archive Management Approaches**					
				cl	**re**	$\mathbf{h_2}$	$\mathbf{h_5}$	$\mathbf{h_{10}}$	$\mathbf{h_d}$
all	all	S	Wins	6.37	7.34	17.22	20.56	23.83	24.17
all	all	S	Losses	28.18	26.38	13.05	10.06	12.41	9.41
all	all	S	Diff	-21.81	-19.04	4.17	10.5	11.42	14.76
all	all	S	Rank	6	5	4	3	2	**1**
all	all	VD	Wins	1.16	3.22	37.75	39.82	40.19	53.78
all	all	VD	Losses	60.26	56.44	20.89	14.01	20.75	3.57
all	all	VD	Diff	-59.1	-53.22	16.86	25.81	19.44	50.21
all	all	VD	Rank	6	5	4	2	3	**1**
all	all	MS	Wins	0.41	0.28	12.61	18.38	22.61	17.3
all	all	MS	Losses	29.03	28.61	8.7	1.65	0.29	3.31
all	all	MS	Diff	-28.62	-28.33	3.91	16.73	22.32	13.99
all	all	MS	Rank	6	5	4	2	**1**	3
all	all	acc	Wins	0.87	2.73	32.82	36.15	36.88	46.33
all	all	acc	Losses	52.94	49.24	18.68	11.68	19.11	4.13
all	all	acc	Diff	-52.07	-46.51	14.14	24.47	17.77	42.2
all	all	acc	Rank	6	5	4	2	3	**1**
all	all	$stab$	Wins	9.65	10.08	14.25	12.69	14.8	14.23
all	all	$stab$	Losses	16.05	15.57	11.11	11.47	10.66	10.84
all	all	$stab$	Diff	-6.4	-5.49	3.14	1.22	4.14	3.39
all	all	$stab$	Rank	6	5	3	4	**1**	2
all	all	NS	Wins	0.61	1.58	25.89	29.9	33.47	35.82
all	all	NS	Losses	48.55	45.53	12.85	6.99	10.03	3.32
all	all	NS	Diff	-47.94	-43.95	13.04	22.91	23.44	32.5
all	all	NS	Rank	6	5	4	3	2	**1**

6.3 Type II DMOP Results

When both the POS and the POF change after an environment change occurs, the benchmark function is classified as a type II environment. As can be seen from Table 6.7, the $\mathbf{h_d}$ approach again outperformed the other approaches by a large margin. The $\mathbf{h_2}$ and $\mathbf{h_5}$ approaches were very close to one another when it came to the *Diff* of wins and losses. The local search approaches' *Diff* value is positive, which indicates that they had more statistically significant better results compared to the **cl** and **re** approaches, for which the *Diff* value is negative. When *Diff* is negative, it means that there were more losses than wins (i.e. more statistically significant worse results).

Table 6.7 Overall wins and losses by the various archive management approaches for type II DMOPs

Results	**Archive Management Approaches**					
	cl	**re**	$\mathbf{h_2}$	$\mathbf{h_5}$	$\mathbf{h_{10}}$	$\mathbf{h_d}$
Wins	106.34	162.52	593.41	585.82	558.24	836.75
Losses	867.86	740.98	318.2	319.55	433.85	162.64
Diff	-761.52	-578.46	275.21	266.27	124.39	674.11
Rank	6	5	2	3	4	**1**

Since the difference between $\mathbf{h_2}$ and $\mathbf{h_5}$ was very small, Table 6.8 is inspected for more insight. It is evident that $\mathbf{h_2}$ ranked higher than $\mathbf{h_5}$ and $\mathbf{h_{10}}$ when the changes were not severe and when the frequency of changes was fast. When the severity of changes was large or when there was more time to optimise a DMOP, then $\mathbf{h_5}$ and $\mathbf{h_{10}}$ approaches respectively outperformed $\mathbf{h_2}$. It should be noted that these results reflect the type of experiments conducted. For example, if there were more experiments where the severity is large and when changes happen at a rapid pace (i.e. $n_t = 1$ and $\tau_t = 10$), the overall results would most likely show that $\mathbf{h_{10}}$ was the best approach for managing the archive, as it is clearly more superior strategy when compared with $\mathbf{h_d}$. The **cl** approach ranked last and **re** ranked second last. The reason why **cl** is inferior to **re** is because clearing the archive results in a complete loss of information with regards to the previously found POF. If the POF did not change by much after the environment change, then clearing the archive is a wasteful procedure.

Table 6.8 Overall wins and losses by various frequencies and severities of change for type II DMOPs across all performance measures

n_t	τ_t	PM	Results	Archive Management Approaches					
				cl	**re**	h_2	h_5	h_{10}	h_d
10	10	all	Wins	25.6	44.44	200.53	191.98	155.62	288.69
10	10	all	Losses	288.53	250.18	99.49	97.33	136.78	34.55
10	10	all	Diff	-262.93	-205.74	101.04	94.65	18.84	254.14
10	10	all	Rank	6	5	2	3	4	**1**
10	25	all	Wins	13.22	14.99	67.5	66.07	55.71	99.01
10	25	all	Losses	97.16	86.83	36.41	35.12	46.97	14.01
10	25	all	Diff	-83.94	-71.84	31.09	30.95	8.74	85
10	25	all	Rank	6	5	2	3	4	**1**
10	50	all	Wins	6.45	7.47	27.8	28.32	24.21	44.07
10	50	all	Losses	41.1	36.59	18	15.57	20.58	6.48
10	50	all	Diff	-34.65	-29.12	9.8	12.75	3.63	37.59
10	50	all	Rank	6	5	3	2	4	**1**
1	10	all	Wins	34.47	47.91	71.34	115.04	199.15	129.61
1	10	all	Losses	157.94	129.4	99.43	68.61	67.95	74.19
1	10	all	Diff	-123.47	-81.49	-28.09	46.43	131.2	55.42
1	10	all	Rank	6	5	4	3	**1**	2
20	10	all	Wins	26.6	47.71	226.24	184.41	123.55	275.37
20	10	all	Losses	283.13	237.98	64.87	102.92	161.57	33.41
20	10	all	Diff	-256.53	-190.27	161.37	81.49	-38.02	241.96
20	10	all	Rank	6	5	2	3	4	**1**

It can be concluded from Table 6.9 that $\mathbf{h_d}$ ranked first for all of the performance measures. Therefore, it is the most accurate approach for managing the bounded archive. As for the **cl** and **re** approaches, the results were in line with the results for type I DMOPs. They were inferior to the local search, because the hill climber attempts to improve the quality of solutions, whereas the other two approaches simply delete or re-evaluate existing non-dominating solutions in the archive.

While the ranking algorithm is useful in evaluating archive management approaches, it is still useful to visually inspect the overall impact that the local search has on the MGPSO's ability to track the true POF. Figure 6.1 should give a general idea of how effective the $\mathbf{h_d}$ approach was compared to the **re** approach for the F7 DMOP. It is clear that $\mathbf{h_d}$ vastly outperformed the simpler **re** approach. Similar results were obtained for the other DMOPs considered in this study.

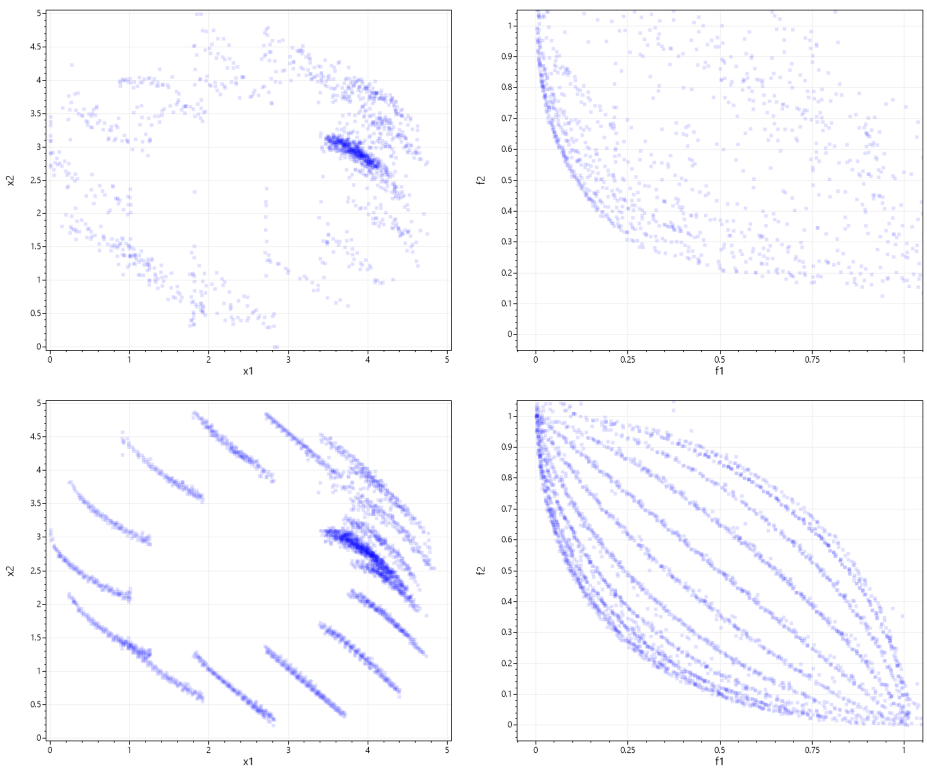
Table 6.9 Overall wins and losses by various performance measures for type II DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**Archive Management Approaches**					
				cl	**re**	$\mathbf{h_2}$	$\mathbf{h_5}$	$\mathbf{h_{10}}$	$\mathbf{h_d}$
all	all	*S*	Wins	15.81	27.07	114.02	107.42	95.79	159.63
all	all	*S*	Losses	165.66	134.48	54.54	54.55	79.9	30.61
all	all	*S*	Diff	-149.85	-107.41	59.48	52.87	15.89	129.02
all	all	*S*	Rank	6	5	2	3	4	**1**
all	all	*VD*	Wins	21.21	30.99	135.45	134.34	121.06	191.82
all	all	*VD*	Losses	197.57	170.31	70.11	68.77	99.56	28.55
all	all	*VD*	Diff	-176.36	-139.32	65.34	65.57	21.5	163.27
all	all	*VD*	Rank	6	5	3	2	4	**1**
all	all	*MS*	Wins	17.49	17.64	36.05	43.39	47.92	59.19
all	all	*MS*	Losses	61.14	57.39	29.99	23.37	29.17	20.62
all	all	*MS*	Diff	-43.65	-39.75	6.06	20.02	18.75	38.57
all	all	*MS*	Rank	6	5	4	2	3	**1**
all	all	*acc*	Wins	10.6	31.43	148.18	145.23	145.01	213.51
all	all	*acc*	Losses	220.52	189.27	77.5	79.16	100.54	26.97
all	all	*acc*	Diff	-209.92	-157.84	70.68	66.07	44.47	186.54
all	all	*acc*	Rank	6	5	2	3	4	**1**
all	all	*stab*	Wins	35.6	42.39	78.01	76.68	73.91	95.3
all	all	*stab*	Losses	98.05	83.7	53.05	54.83	66.53	45.73
all	all	*stab*	Diff	-62.45	-41.31	24.96	21.85	7.38	49.57
all	all	*stab*	Rank	6	5	2	3	4	**1**
all	all	*NS*	Wins	5.63	13	81.7	78.76	74.55	117.3
all	all	*NS*	Losses	124.92	105.83	33.01	38.87	58.15	10.16
all	all	*NS*	Diff	-119.29	-92.83	48.69	39.89	16.4	107.14
all	all	*NS*	Rank	6	5	2	3	4	**1**

6.4 Type III DMOP Results

This section focuses on type III DMOPs. A benchmark function belongs to a type III category when the POS remains unchanged, but the POF changes overtime. Table 6.10 indicates that, once gain, $\mathbf{h_d}$ was the superior approach, followed by $\mathbf{h_2}$ and then by $\mathbf{h_{10}}$. This is an interesting development because $\mathbf{h_5}$ was clearly inferior to the other hill climber variants, where for type I and type II problems it ranked higher.

In order to understand these findings better, the results are explored further by considering various frequencies of change and severities of change as in Table 6.11. When $n_t = 1$, it was $\mathbf{h_2}$ that received a rank of 1. However, for the other DMOP types, large spatial severity resulted in poor performance for $\mathbf{h_2}$ (see Table 6.5 and



(a) POS **(b)** POF

Figure 6.1 Obtained POS and POF for the F7 problem where $n_t = 10$ and $\tau_t = 25$. Top row shows the results obtained by the **re** approach whereas bottom row show the results obtained by the $\mathbf{h_d}$ approach.

Table 6.10 Overall wins and losses by the various archive management approaches for type III DMOPs across all performance measures and n_t-τ_t combinations

Results	**Archive Management Approaches**					
	cl	**re**	$\mathbf{h_2}$	$\mathbf{h_5}$	$\mathbf{h_{10}}$	$\mathbf{h_d}$
Wins	65.1	63.72	119.13	95.3	101.99	140.41
Losses	233.88	133.41	51.82	64.43	59.32	42.79
Diff	-168.78	-69.69	67.31	30.87	42.67	97.62
Rank	6	5	2	4	3	**1**

Table 6.8). The difference in ranks when compared to type I and type II DMOPs is due to the fact that type III DMOPs do not change in the decision space. In other words, decision variables did not move to a new feasible region far away from the old region when the change to the environment occurred. This means that MGPSO was able to focus more on exploitation of the already found solutions rather than exploring the search space for new, better solutions. A small step size contributed

Table 6.11 Overall wins and losses for various frequencies and severities of change for type III DMOPs across all performance measures

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**Archive Management Approaches**					
				cl	**re**	$\mathbf{h_2}$	$\mathbf{h_5}$	$\mathbf{h_{10}}$	$\mathbf{h_d}$
10	10	all	Wins	17.71	16.4	28.87	25.89	25.65	38.06
10	10	all	Losses	64.23	35	13.25	14.47	14.94	10.69
10	10	all	Diff	-46.52	-18.6	15.62	11.42	10.71	27.37
10	10	all	Rank	6	5	2	3	4	**1**
10	25	all	Wins	9.38	4.99	9.2	7.83	8.22	12.25
10	25	all	Losses	20.19	12.48	4.79	5.48	5.28	3.65
10	25	all	Diff	-10.81	-7.49	4.41	2.35	2.94	8.6
10	25	all	Rank	6	5	2	4	3	**1**
10	50	all	Wins	4.27	2.09	3.75	3	3.07	6.01
10	50	all	Losses	7.68	6.18	2.07	2.32	2.54	1.4
10	50	all	Diff	-3.41	-4.09	1.68	0.68	0.53	4.61
10	50	all	Rank	5	6	2	3	4	**1**
1	10	all	Wins	16.07	24.07	47.65	33.43	40.31	47.44
1	10	all	Losses	76.28	45	18.18	28.19	22.54	18.78
1	10	all	Diff	-60.21	-20.93	29.47	5.24	17.77	28.66
1	10	all	Rank	6	5	**1**	4	3	2
20	10	all	Wins	17.67	16.17	29.66	25.15	24.74	36.65
20	10	all	Losses	65.5	34.75	13.53	13.97	14.02	8.27
20	10	all	Diff	-47.83	-18.58	16.13	11.18	10.72	28.38
20	10	all	Rank	6	5	2	3	4	**1**

more to the exploitation and thus outperformed the other approaches.

When it comes to the performance measures (see Table 6.12), the results for the VD metric stand out since **re** and **cl** approaches outperformed the local search approaches. A direct comparison with the other performance measures clearly shows that **re** and **cl** approaches performed poorly on the other metrics. It is unclear at this point why this particular performance measure is deviating from the norm. For type I and type II DMOPs, the **re** approach typically ranked second last and **cl** ranked last. One way to explain this is that the found POF was very close to the true POF, but the solutions were not diverse enough. This scenario is precisely the reason why more than one performance measure should be used when evaluating the performance of DMOAs - to avoid misleading results.

Table 6.12 Overall wins and losses for various performance measures for type III DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	τ_t	**PM**	**Results**	**Archive Management Approaches**					
				cl	**re**	$\mathbf{h_2}$	$\mathbf{h_5}$	$\mathbf{h_{10}}$	$\mathbf{h_d}$
all	all	*S*	Wins	18.06	18.68	30.1	19.64	21.3	40.27
all	all	*S*	Losses	54.67	23.85	16.05	22.32	20.61	10.55
all	all	*S*	Diff	-36.61	-5.17	14.05	-2.68	0.69	29.72
all	all	*S*	Rank	6	5	2	4	3	**1**
all	all	*VD*	Wins	27.03	15.44	9.03	9.07	8.76	10.89
all	all	*VD*	Losses	20.97	11.34	11.87	11.49	11.79	12.76
all	all	*VD*	Diff	6.06	4.1	-2.84	-2.42	-3.03	-1.87
all	all	*VD*	Rank	**1**	2	5	4	6	3
all	all	*MS*	Wins	0.66	2.93	14.58	12.25	13.95	15.76
all	all	*MS*	Losses	30.63	19.3	2.63	3.25	2.79	1.53
all	all	*MS*	Diff	-29.97	-16.37	11.95	9	11.16	14.23
all	all	*MS*	Rank	6	5	2	4	3	**1**
all	all	*acc*	Wins	11.15	13.61	41.98	32.76	32.39	49.1
all	all	*acc*	Losses	72.74	49.03	13.78	18.31	17.71	9.42
all	all	*acc*	Diff	-61.59	-35.42	28.2	14.45	14.68	39.68
all	all	*acc*	Rank	6	5	2	4	3	**1**
all	all	*stab*	Wins	7.96	11.36	16.11	14.29	17.67	16.51
all	all	*stab*	Losses	37.12	16.65	6.94	8.38	6.35	8.46
all	all	*stab*	Diff	-29.16	-5.29	9.17	5.91	11.32	8.05
all	all	*stab*	Rank	6	5	2	4	**1**	3
all	all	*NS*	Wins	0.24	1.7	7.33	7.29	7.92	7.88
all	all	*NS*	Losses	17.75	13.24	0.55	0.68	0.07	0.07
all	all	*NS*	Diff	-17.51	-11.54	6.78	6.61	7.85	7.81
all	all	*NS*	Rank	6	5	3	4	**1**	2

6.5 Comparisons with the other DMOAs

Based on findings from the previous section on the archive management approaches, it can be concluded that $\mathbf{h_d}$ outperformed the other strategies. Therefore, the $\mathbf{h_d}$ approach is used in the final comparative study against baseline MGPSO (**re** approach), DMOES, SGEA, and DNSGA-II. Instead of using the **re** and $\mathbf{h_d}$ labels, the approaches are renamed to MGPSO and MGPSO$_{ls}$, respectively. Parameter settings used by each DMOA can be found in the Section 5.2.

Table 6.13 shows that MGPSO$_{ls}$ vastly outperformed the other algorithms. The rank was calculated by computing wins and losses over the twenty-nine DMOPs across all performance measures and n_t-τ_t combinations. Table 6.13 itself does not provide

Table 6.13 Overall wins and losses by the various DMOAs across all performance measures and n_t-τ_t combinations

Results	**DMOAs**				
	MGPSO	**MGPSO$_{ls}$**	**DMOES**	**DNSGA-II**	**SGEA**
Wins	968.35	1534.54	723.58	443.43	810.44
Losses	694.66	247.3	1200.33	1333.44	1004.61
Diff	273.69	1287.24	-476.75	-890.01	-194.17
Rank	2	**1**	4	5	3

Table 6.14 Overall wins and losses for various frequencies and severities of change across all performance measures

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**DMOAs**				
				MGPSO	**MGPSO$_{ls}$**	**DMOES**	**DNSGA-II**	**SGEA**
10	10	all	Wins	280.66	471.28	204.77	110.33	208.04
10	10	all	Losses	194.47	52.23	332.6	392.33	303.45
10	10	all	Diff	86.19	419.05	-127.83	-282	-95.41
10	10	all	Rank	2	**1**	4	5	3
10	25	all	Wins	98.5	169.85	93.46	46.77	91.72
10	25	all	Losses	84.97	28.56	120.7	154.38	111.69
10	25	all	Diff	13.53	141.29	-27.24	-107.61	-19.97
10	25	all	Rank	2	**1**	4	5	3
10	50	all	Wins	46.31	75.31	47.64	25.21	48.88
10	50	all	Losses	43.55	18.63	58.19	71.81	51.17
10	50	all	Diff	2.76	56.68	-10.55	-46.6	-2.29
10	50	all	Rank	2	**1**	4	5	3
1	10	all	Wins	282.88	359.67	145.55	152.75	257.91
1	10	all	Losses	166.68	93.19	384.35	323.1	231.44
1	10	all	Diff	116.2	266.48	-238.8	-170.35	26.47
1	10	all	Rank	2	**1**	5	4	3
20	10	all	Wins	260	458.43	232.16	108.37	203.89
20	10	all	Losses	204.99	54.69	304.49	391.82	306.86
20	10	all	Diff	55.01	403.74	-72.33	-283.45	-102.97
20	10	all	Rank	2	**1**	3	5	4

any information with regards to the performance measures or n_t-τ_t combinations. Therefore, Table 6.14 should be examined more closely to evaluate the performance of DMOAs with respect to n_t-τ_t combinations, or Table 6.15 for results with respect to S, VD, MS, acc, $stab$, and NS performance measures.

It is evident from the results in Table 6.14 that the MGPSO is outranked by the

MGPSO$_{ls}$, but managed to outperform the other DMOAs. The SGEA algorithm received a rank of 3 for most n_t-τ_t combinations, except when $n_t = 20$ and $\tau_t = 10$. The DMOES algorithm received a rank of 4, followed by DNSGA-II that ranked last for most of the n_t-τ_t combinations.

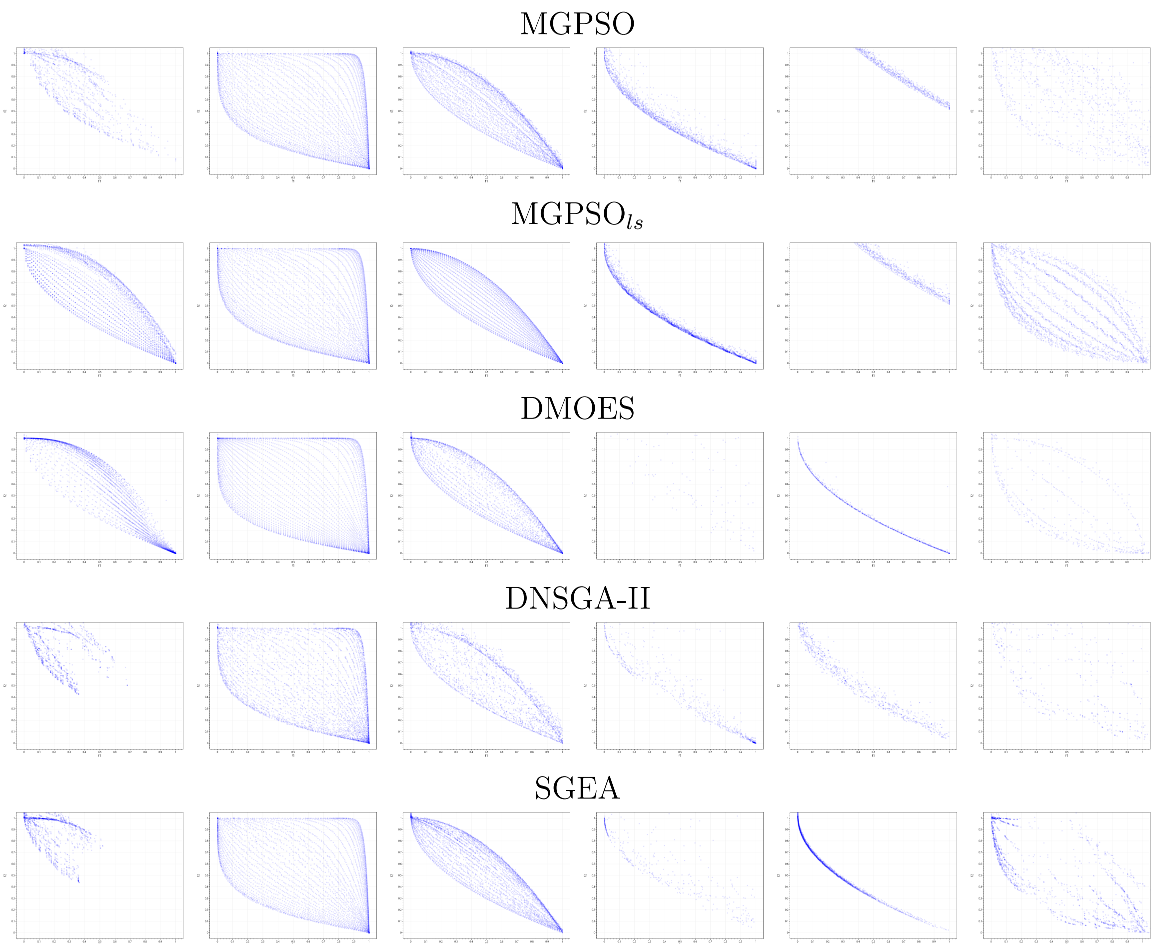
Table 6.15 Overall wins and losses for various performance measures across all n_t-τ_t combinations

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**DMOAs**				
				MGPSO	**MGPSO$_{ls}$**	**DMOES**	**DNSGA-II**	**SGEA**
all	all	*S*	Wins	211.95	294.03	82.04	52.94	180.57
all	all	*S*	Losses	97.52	38.74	244.97	286.56	153.74
all	all	*S*	Diff	114.43	255.29	-162.93	-233.62	26.83
all	all	*S*	Rank	2	**1**	4	5	3
all	all	*VD*	Wins	185.18	298.51	125.42	66.73	163.91
all	all	*VD*	Losses	134.83	42.32	205.82	278.36	178.42
all	all	*VD*	Diff	50.35	256.19	-80.4	-211.63	-14.51
all	all	*VD*	Rank	2	**1**	4	5	3
all	all	*MS*	Wins	184.35	263.92	90.08	126.97	89.58
all	all	*MS*	Losses	79.35	18.87	267.35	145.12	244.21
all	all	*MS*	Diff	105	245.05	-177.27	-18.15	-154.63
all	all	*MS*	Rank	2	**1**	5	3	4
all	all	*acc*	Wins	162.38	310.47	246.81	42.26	143.81
all	all	*acc*	Losses	179.77	56.1	128.73	325.12	216.01
all	all	*acc*	Diff	-17.39	254.37	118.08	-282.86	-72.2
all	all	*acc*	Rank	3	**1**	2	5	4
all	all	*stab*	Wins	112.7	160.94	144.41	76.29	80.64
all	all	*stab*	Losses	102.83	74.89	93.92	158.47	144.87
all	all	*stab*	Diff	9.87	86.05	50.49	-82.18	-64.23
all	all	*stab*	Rank	3	**1**	2	5	4
all	all	*NS*	Wins	111.79	206.67	34.82	78.24	151.93
all	all	*NS*	Losses	100.36	16.38	259.54	139.81	67.36
all	all	*NS*	Diff	11.43	190.29	-224.72	-61.57	84.57
all	all	*NS*	Rank	3	**1**	5	4	2

Table 6.15 provides similar results when it comes to the MGPSO$_{ls}$, where it clearly won against the other DMOAs across all performance measures. However, MGPSO this time received a rank of 2 for S, VD, and MS, while it ranked third for acc, $stab$, and NS. Good maximum spread and the spread of solutions are important, because the algorithm needs to have a diverse set of non-dominated solutions to choose from along the found POF. However, if the accuracy of the algorithm is not adequate,

then it does not matter much if the spread is good, because the DMOA needs to successfully approximate the true POF in the first place. DMOES received a rank of 2 for *acc*, and *stab*, which indicates that it was more accurate on some of the problems compared to MGPSO, SGEA, and DNSGA-II. The DNSGA-II algorithm ranked last on all metrics except for the MS and NS.

While the ranking algorithm is useful in evaluation of the various DMOAs, it still might be useful to visually inspect DMOAs' ability to track the true POF. Figure 6.2 depicts six different benchmark functions to visually explain the differences between each algorithm.



(a) ZJZ **(b)** $FDA2_{Cam}$ **(c)** dMOP2 **(d)** dMOP3 **(e)** DIMP2 **(f)** F5

Figure 6.2 Obtained POFs for some of the problems where $n_t = 10$ and $\tau_t = 50$ for DIMP2 and $n_t = 10$ and $\tau_t = 25$ for the other benchmark functions. Each row shows the results for a specific DMOA.

The POFs obtained by the DMOAs for ZJZ, FDA2$_{Cam}$, dMOP2, dMOP3, DIMP2 and F5 DMOPs are displayed in Figure 6.2. For the ZJZ benchmark function, MGPSO with the **re** archive management approach, SGEA, and DNSGA-II struggled with approximating the true POF. However, MGPSO$_{ls}$ and DMOES successfully approximated the POF, with MGPSO$_{ls}$ being slightly ahead when it comes to the spacing and diversity of the found solutions. The ZJZ problem is a modified version of the FDA1 problem, where unlike the POS of FDA1 that has line segments parallel to the coordinate axes, ZJZ has nonlinear linkages between the decision variables, thus making it much more difficult to solve [17]. This is a type II problem.

The FDA2$_{Cam}$ problem was an interesting case, because even though it appears that each DMOA was able to obtain good diversity of solutions, none of the algorithms were able to approximate the true POF of FDA2$_{Cam}$ at all times. When the POF began to change from convex to concave, all of the DMOAs started to loose track of the true POF. Still, DMOES performed slightly better on this problem since the spacing between solutions was more evenly spread and variational distance was low. This DMOP is a type III problem, where the POS remains static, but the POF changes over time.

The dMOP2 problem is classified as a type II problem, where both the POS and the POF change over time. MGPSO, MGPSO$_{ls}$, DMOES and SGEA performed quite well on this problem and while DNSGA-II was close to the true POF, it was not as accurate as the other DMOAs. It is evident that MGPSO$_{ls}$ outperformed the other algorithms in terms of the diversity of solutions, spacing between solutions, and overall accuracy. Good results can be attributed to the local search archive management approach that allowed MGPSO, right after the environment change, to begin to exploit already found good regions of the search space. Otherwise, it would need to spend more time exploring the search space for a feasible region.

The dMOP3 problem is classified as the type I problem, where the POS changes over time, but the POF remains static. This time, MGPSO and MGPSO$_{ls}$ were able to approximate the true POF, whereas DMOES, DNSGA-II, and SGEA struggled. DMOES did not manage to find any solution.

For the DIMP2 problem, the results were opposite to the ones from the dMOP3 problem. This time, DMOES, DNSGA-II, and SGEA were able to find good approximations, with DMOES being the clear winner. SGEA struggled with the maximum spread of solutions along the found POF, whereas DNSGA was not as accurate as DMOES and SGEA. MGPSO and MGPSO$_{ls}$ were unable to find the global POF optimum and were stuck in a local POF optimum instead. The maximum spread and spacing was quite good for both MGPSO and MGPSO_{ls}, but it does not matter much when the true POF is not found. It has been shown that the MGPSO implementation for SMOPs struggled with the ZDT4 static MOO problem [11]. The DIMP2 problem is a modified dynamic version of the ZDT4 problem which might explain why the results were poor.

The last comparison from Figure 6.2 was performed on the F5 problem, which is classified as a type II problem, where the POS and the POF changes with regards to time. Only MGPSO_{ls} was able to keep track of the true POF for this problem. The other DMOAs would require more iterations to be able to find the true POF, because they were unable to find and exploit the feasible region of the search space within the limited time-frame.

6.6 Summary

This chapter adapted the multi-guide particle swarm optimisation (MGPSO) algorithm for dynamic multi-objective optimisation (DMOO) by introducing various archive management strategies to allow MGPSO to track changing Pareto-optimal fronts. The effect that these various archive management approaches have on the performance of the MGPSO was examined in this chapter and it was determined that a hill climber with a decreasing step size worked best on a large variety of dynamic optimisation problems. The best performing archive approach was then used in a comparative study with a baseline MGPSO, DMOES, DNSGA-II, and SGEA. The extensive empirical analysis showed that MGPSO with a local search archive management strategy was highly competitive and oftentimes outperformed the other *state-of-the-art* dynamic optimisation algorithms.

Next chapter conducts detailed analysis of the experimental results for balance coefficient update strategies.

Chapter 7

Balance Coefficient Experiments

The purpose of this section is to go over all the experiments done in this study and to perform an extensive analysis of the results. Firstly, five balance coefficient update strategies for all DMOP types are compared in Section 7.1. Then in Section 7.2, the best performing strategy from the first experiment is compared with the linearly increasing and linearly decreasing strategies. The goal of these experiments is to find the balance coefficient update strategy that performs best across all of the twenty-nine benchmark functions and then use it in a comparative study with the other *state-of-the-art* DMOAs in Section 7.3. Performance measures used in the experiments are S, VD, MS, acc, $stab$, and NS as defined in Section 2.3. When the table with results says that $pm = all$, it means that the results of the wins and losses calculations for these six performance measures are combined in the calculation of ranks. The re-evaluation of non-dominating solutions archive management strategy, **re**, from the previous experiment is used in every MGPSO variant.

7.1 Results for Standard and Random Approaches

Table 7.1 displays the overall results by the various balance coefficient update strategies. It is evident that the standard strategies (**std** and $\mathbf{std}_{\tau_t}$) vastly outperformed the other approaches by a large margin. Out of these two standard approaches, it was the $\mathbf{std}_{\tau_t}$ strategy that performed best across all of the twenty-nine benchmark functions. The random update strategies were inferior due to being too stochastic, which will be further explored in the next sections. While the evaluation algorithm takes into account the tracking ability of the algorithms, Table 7.1 contains only the overall results - which makes it hard to assess the performance of each approach with regards to the various frequencies and severities of change, as well as the six performance measures. Table 7.2 and Table 7.3 breaks down the results into overall wins and losses by the frequencies and severities of change, and by the performance measures, respectively.

Table 7.1 Overall wins and losses by the various balance coefficient update strategies for all performance measures and n_t-τ_t combinations

Results	**Balance Coefficient Update Strategies**				
	std	$\mathbf{std}_{\tau_t}$	**r**	$\mathbf{r}_i$	$\mathbf{r}_{ij}$
Wins	598.51	670.05	185.55	301.67	184.8
Losses	239.71	161.54	503.86	388.31	647.16
Diff	358.8	508.51	-318.31	-86.64	-462.36
Rank	2	**1**	4	3	5

Table 7.2 Overall wins and losses for various frequencies and severities of change across all performance measures

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**Balance Coefficient Update Strategies**				
				std	$\mathbf{std}_{\tau_t}$	**r**	$\mathbf{r}_i$	$\mathbf{r}_{ij}$
10	10	all	Wins	164.51	175	39.22	88.95	52.56
10	10	all	Losses	60.81	42.66	149.01	96.21	171.55
10	10	all	Diff	103.7	132.34	-109.79	-7.26	-118.99
10	10	all	Rank	2	**1**	4	3	5
10	25	all	Wins	60.52	69.25	22.65	39.99	19.53
10	25	all	Losses	25.74	17.79	51.14	38.47	78.8
10	25	all	Diff	34.78	51.46	-28.49	1.52	-59.27
10	25	all	Rank	2	**1**	4	3	5
10	50	all	Wins	29.49	32.49	13.69	20.9	9.44
10	50	all	Losses	13.7	9.8	23.03	18.54	40.94
10	50	all	Diff	15.79	22.69	-9.34	2.36	-31.5
10	50	all	Rank	2	**1**	4	3	5
1	10	all	Wins	194.19	222.71	61.17	58.61	52.84
1	10	all	Losses	71.33	46.79	150.56	144.19	176.65
1	10	all	Diff	122.86	175.92	-89.39	-85.58	-123.81
1	10	all	Rank	2	**1**	4	3	5
20	10	all	Wins	149.8	170.6	48.82	93.22	50.43
20	10	all	Losses	68.13	44.5	130.12	90.9	179.22
20	10	all	Diff	81.67	126.1	-81.3	2.32	-128.79
20	10	all	Rank	2	**1**	4	3	5

The results from Table 7.2 indicate that for all environment types, $\mathbf{std}_{\tau_t}$ outperformed the other approaches, followed by **std** approach. The $\mathbf{r}_i$ strategy ranked third,

r ranked fourth, and $\mathbf{r}_{ij}$ ranked last on all environment types. This corresponds exactly with the results from Table 7.1. The results for the fast changing environment ($\tau_t = 10$) and for the environment with a big spatial severity ($n_t = 1$), showed that $\mathbf{std}_{\tau_t}$ is the preferable strategy compared to the **std** approach. However, this difference became less noticeable when the frequency of change became larger ($\tau_t \leq 25$), but this is due to the fact that the changes to the environment were not as frequent and each MGPSO variant had more time to optimise each objective. One explanation on why $\mathbf{std}_{\tau_t}$ is better, is that the **std** strategy might have randomly initialized the balance coefficient to a less desirable value, which inhibited the MGPSO's ability to approximate the true POF. Since $\mathbf{std}_{\tau_t}$ strategy re-initializes the λ_i parameter ev-

Table 7.3 Overall wins and losses for various performance measures across all n_t-τ_t combinations

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**Balance Coefficient Update Strategies**				
				std	$\mathbf{std}_{\tau_t}$	**r**	$\mathbf{r}_i$	$\mathbf{r}_{ij}$
all	all	*S*	Wins	68.73	79.82	33.24	40.7	23.48
all	all	*S*	Losses	39.86	24.42	47.22	43.8	90.67
all	all	*S*	Diff	28.87	55.4	-13.98	-3.1	-67.19
all	all	*S*	Rank	2	**1**	4	3	5
all	all	*VD*	Wins	151.18	176.11	37.01	73.81	52.13
all	all	*VD*	Losses	55.72	31.11	143.87	104.35	155.19
all	all	*VD*	Diff	95.46	145	-106.86	-30.54	-103.06
all	all	*VD*	Rank	2	**1**	5	3	4
all	all	*MS*	Wins	46.31	50.3	14.89	30	17.94
all	all	*MS*	Losses	22.39	14.36	45.43	25.01	52.25
all	all	*MS*	Diff	23.92	35.94	-30.54	4.99	-34.31
all	all	*MS*	Rank	2	**1**	4	3	5
all	all	*acc*	Wins	161.5	187.28	39.72	78.71	39.3
all	all	*acc*	Losses	54.39	33.21	139.1	104.61	175.2
all	all	*acc*	Diff	107.11	154.07	-99.38	-25.9	-135.9
all	all	*acc*	Rank	2	**1**	4	3	5
all	all	*stab*	Wins	50.21	55.01	30.71	41.01	44.11
all	all	*stab*	Losses	50.7	42.81	46.05	35.64	45.85
all	all	*stab*	Diff	-0.49	12.2	-15.34	5.37	-1.74
all	all	*stab*	Rank	3	**1**	5	2	4
all	all	*NS*	Wins	120.58	121.53	29.98	37.44	7.84
all	all	*NS*	Losses	16.65	15.63	82.19	74.9	128
all	all	*NS*	Diff	103.93	105.9	-52.21	-37.46	-120.16
all	all	*NS*	Rank	2	**1**	4	3	5

ery environment change, it overcomes the problem of a bad initial λ_i value. When it comes to $\mathbf{r}$, $\mathbf{r}_i$, and $\mathbf{r}_{ij}$ approaches, whose main idea is to re-initialize the λ_i parameter at every iteration, the results clearly indicate that these more stochastic strategies were vastly inferior to the standard approaches.

Table 7.3 breaks down the results by each performance measure. This time, the ranks were not assigned evenly as in Table 7.2, since $\mathbf{r}_i$ ranked second for the *stab* metric. The differences in ranks are also evident for random approaches. Still, it is clear that $\mathbf{std}_{\tau_t}$ outperformed the other approaches in every performance measure. The **std** approach ranked second for S, VD, MS, *acc*, and NS and received a rank of 3 for the *stab* performance measure. The random strategies were consistently outperformed by the standard strategies except for the *stab* metric. Out of the three random approaches, $\mathbf{r}_i$ was the more effective strategy, followed by $\mathbf{r}$ and then by $\mathbf{r}_{ij}$ strategy. Interestingly, re-sampling the balance coefficient, per particle, at every iteration vastly outperformed the other two random strategies.

7.2 Results for Linearly Increasing and Decreasing Strategies

Table 7.4 displays the overall results by the $\mathbf{std}_{\tau_t}$, linearly increasing (**li** and $\mathbf{li}_{\tau_t}$) and linearly decreasing (**ld** and $\mathbf{ld}_{\tau_t}$) balance coefficient update strategies. It is evident that $\mathbf{std}_{\tau_t}$ approach outperformed the other approaches by a large margin. Both $\mathbf{li}_{\tau_t}$ and $\mathbf{ld}_{\tau_t}$ did better compared to **li** and **ld** strategies. This implies that re-initializing the λ_i parameter, right after the change to the environment occurs, is preferable to the default strategy where the balance coefficient is set once at the beginning of the run. To learn more about each approach, Table 7.5 and Table 7.6 breaks down the results into overall wins and losses by the frequencies and severities of change, and by the performance measures, respectively.

Table 7.4 Overall wins and losses by the various balance coefficient update strategies for all performance measures and n_t-τ_t combinations

Results	**Balance Coefficient Update Strategies**				
	$\mathbf{std}_{\tau_t}$	**ld**	$\mathbf{ld}_{\tau_t}$	**li**	$\mathbf{li}_{\tau_t}$
Wins	922.98	503.85	634.07	574.9	750.05
Losses	333.49	1142.15	623.03	832.05	455.13
Diff	589.49	-638.3	11.04	-257.15	294.92
Rank	**1**	5	3	4	2

Table 7.5 Overall wins and losses for various frequencies and severities of change across all performance measures

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**Balance Coefficient Update Strategies**				
				$\mathbf{std}_{\tau_t}$	**ld**	$\mathbf{ld}_{\tau_t}$	**li**	$\mathbf{li}_{\tau_t}$
10	10	all	Wins	251.07	136.89	171.58	157.28	213.19
10	10	all	Losses	82.35	319.25	173.86	237.76	116.79
10	10	all	Diff	168.72	-182.36	-2.28	-80.48	96.4
10	10	all	Rank	**1**	5	3	4	2
10	25	all	Wins	104	48.73	79.56	64.86	99.66
10	25	all	Losses	35.59	144.52	74.13	99.06	43.51
10	25	all	Diff	68.41	-95.79	5.43	-34.2	56.15
10	25	all	Rank	**1**	5	3	4	2
10	50	all	Wins	52.76	22.84	38.47	35.86	49.68
10	50	all	Losses	19.06	74.34	38.56	44.7	22.95
10	50	all	Diff	33.7	-51.5	-0.09	-8.84	26.73
10	50	all	Rank	**1**	5	3	4	2
1	10	all	Wins	265.94	165.02	154.07	161.1	171.36
1	10	all	Losses	110.45	258.96	183.5	207.54	157.04
1	10	all	Diff	155.49	-93.94	-29.43	-46.44	14.32
1	10	all	Rank	**1**	5	3	4	2
20	10	all	Wins	249.21	130.37	190.39	155.8	216.16
20	10	all	Losses	86.04	345.08	152.98	242.99	114.84
20	10	all	Diff	163.17	-214.71	37.41	-87.19	101.32
20	10	all	Rank	**1**	5	3	4	2

With regards to Table 7.5, it can be concluded that ranks received by each strategy were evenly spread out across every n_t-τ_t combination. The linearly increasing and linearly decreasing strategies adapted for dynamic environments ($\mathbf{li}_{\tau_t}$ and $\mathbf{ld}_{\tau_t}$) were more preferable to their default implementations (**li** and **ld**). When decreasing strategy was picked (**ld**), the balance coefficient had very little influence on the velocity update equation during the first half of the run. This might work for static MOPs, but for dynamic MOPs, changes happen often and the archive guide needs to contribute to the movement of the particle so that MGPSO can effectively exploit the search space. As for the linearly increasing strategy (**li**), the balance coefficient had a large influence on the velocity update equation during the first half of the run, but then the results gradually deteriorated over time.

Overall, $\mathbf{std}_{\tau_t}$ strategy was the clear winner here, because MGPSO performs better when the λ_i parameter is re-initialized only once for each environment change. These results correlate with the previous one where the random strategies were explored. It

Table 7.6 Overall wins and losses for various performance measures across all n_t-τ_t combinations

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**Balance Coefficient Update Strategies**				
				$\mathbf{std}_{\tau_t}$	**ld**	$\mathbf{ld}_{\tau_t}$	**li**	$\mathbf{li}_{\tau_t}$
all	all	*S*	Wins	116.04	90.17	108.97	109.53	131.73
all	all	*S*	Losses	81.58	187.96	89.98	131.55	65.37
all	all	*S*	Diff	34.46	-97.79	18.99	-22.02	66.36
all	all	*S*	Rank	2	5	3	4	**1**
all	all	*VD*	Wins	215.83	77.09	130.22	113.26	185.96
all	all	*VD*	Losses	52.74	262.5	137.11	187.95	82.06
all	all	*VD*	Diff	163.09	-185.41	-6.89	-74.69	103.9
all	all	*VD*	Rank	**1**	5	3	4	2
all	all	*MS*	Wins	112.63	76.12	83.29	73.54	78.02
all	all	*MS*	Losses	32.04	163.39	72.52	96.64	59.01
all	all	*MS*	Diff	80.59	-87.27	10.77	-23.1	19.01
all	all	*MS*	Rank	**1**	5	3	4	2
all	all	*acc*	Wins	243.9	45.52	146.9	91.57	171.42
all	all	*acc*	Losses	34.91	264.57	118.66	192.96	88.21
all	all	*acc*	Diff	208.99	-219.05	28.24	-101.39	83.21
all	all	*acc*	Rank	**1**	5	3	4	2
all	all	*stab*	Wins	87.62	116.36	80.3	101.17	79.6
all	all	*stab*	Losses	87.96	110.8	88.71	95.47	82.11
all	all	*stab*	Diff	-0.34	5.56	-8.41	5.7	-2.51
all	all	*stab*	Rank	3	2	5	**1**	4
all	all	*NS*	Wins	146.96	98.59	84.39	85.83	103.32
all	all	*NS*	Losses	44.26	152.93	116.05	127.48	78.37
all	all	*NS*	Diff	102.7	-54.34	-31.66	-41.65	24.95
all	all	*NS*	Rank	**1**	5	3	4	2

was evident that changing the λ_i parameter every iteration had a negative impact on the performance of the algorithm.

Table 7.6 breaks down the results by each performance measure. The only performance measure where **ld** and **li** strategies outperformed their dynamic versions was for *stab* metric. Still, it can be concluded that re-initializing the λ_i parameter after the environment change is a better strategy as these approaches were better for the other performance measures. The $\mathbf{std}_{\tau_t}$ strategy did receive a rank of 1 for VD, MS, *acc*, and NS, a rank of 2 for S, and a rank of 3 for *stab* metric. In conclusion, $\mathbf{std}_{\tau_t}$ was the most accurate and diverse approach. These findings match the results from Table 7.4 and Table 7.5, where $\mathbf{std}_{\tau_t}$ outperformed the other strategies.

7.3 Comparisons with the other DMOAs

Based on the findings from the previous section on the balance coefficient update strategies, it can be concluded that $\mathbf{std}_{\tau_t}$ outperformed the other approaches. Therefore, $\mathbf{std}_{\tau_t}$ update strategy, referred to as MGPSO for the rest of the chapter, is used in the comparative study against the other *state-of-the-art* DMOAs. The DMOA's used in this study are DMOES, DNSGA-II, and SGEA. Refer to the Section 5.2 for more information about parameter settings used by each.

Inspecting Table 7.7, it is clear that MGPSO vastly outperformed the other algorithms. The rank was calculated by computing wins and losses over the twenty-nine DMOPs across all performance measures and n_t-τ_t combinations. Table 7.7 itself does not provide any information with regards to the performance measures or n_t-τ_t combinations, therefore, Table 7.8 should be examined to study the differences with respect to each n_t-τ_t combinations, or Table 7.9 for more information about the S, VD, MS, *acc*, *stab*, and NS performance measures.

Table 7.7 Overall wins and losses by the various DMOAs across all performance measures and n_t-τ_t combinations

Results	**DMOAs**			
	MGPSO	**DMOES**	**DNSGA-II**	**SGEA**
Wins	935.44	634.93	401.28	725.06
Losses	394.39	784.57	913.17	604.58
Diff	541.05	-149.64	-511.89	120.48
Rank	**1**	3	4	2

Examining the results from the Table 7.8, it is evident that MGPSO outperformed the other DMOAs across every environment type. The SGEA algorithm received a rank of 2 for most n_t-τ_t combinations, except when the spatial severity was low and the frequency of changes was fast ($n_t = 20$ and $\tau_t = 10$). The DMOES algorithm received a rank of 3 when $\tau_t = 10$, and ranked second when $\tau_t = 20$, followed by DNSGA-II that ranked last for most of the n_t-τ_t combinations.

Table 7.8 Overall wins and losses for various frequencies and severities of change across all performance measures

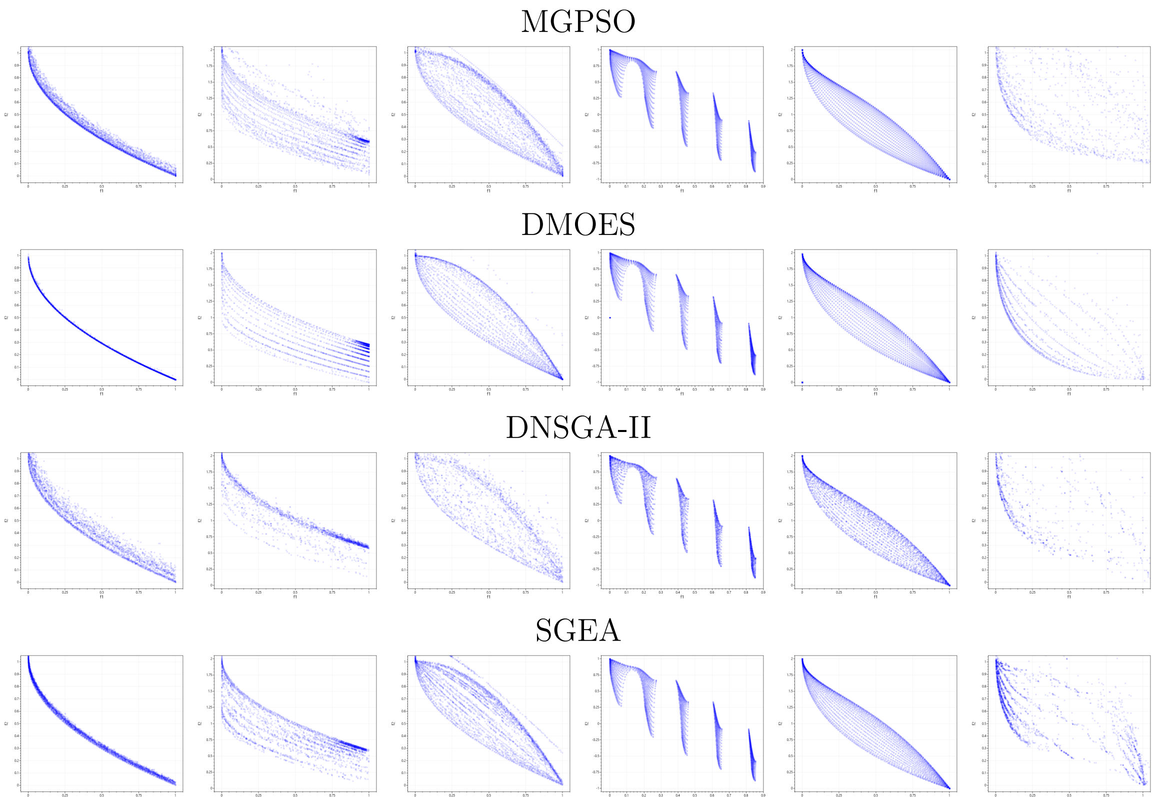
$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**DMOAs**			
				MGPSO	**DMOES**	**DNSGA-II**	**SGEA**
10	10	all	Wins	270.86	184.36	103.43	193.44
10	10	all	Losses	98.03	209.42	265.97	178.67
10	10	all	Diff	172.83	-25.06	-162.54	14.77
10	10	all	Rank	**1**	3	4	2
10	25	all	Wins	95.54	82.84	43.57	81.87
10	25	all	Losses	52.24	77.64	106.27	67.67
10	25	all	Diff	43.3	5.2	-62.7	14.2
10	25	all	Rank	**1**	3	4	2
10	50	all	Wins	44.54	40.06	22.46	42.74
10	50	all	Losses	28.98	38.8	49.87	32.15
10	50	all	Diff	15.56	1.26	-27.41	10.59
10	50	all	Rank	**1**	3	4	2
1	10	all	Wins	269.47	124.29	132.06	219.88
1	10	all	Losses	109.11	267.38	224.87	144.34
1	10	all	Diff	160.36	-143.09	-92.81	75.54
1	10	all	Rank	**1**	4	3	2
20	10	all	Wins	255.03	203.38	99.76	187.13
20	10	all	Losses	106.03	191.33	266.19	181.75
20	10	all	Diff	149	12.05	-166.43	5.38
20	10	all	Rank	**1**	2	4	3

Table 7.9 provides similar results when it comes to the MGPSO, where it clearly won against the other DMOAs for S, VD, and MS performance measures and received a rank of 2 for *acc*, *stab*, and NS metrics. The DNSGA-II ranked last on all metrics but NS and MS. The SGEA ranked 2nd for S, VD, and a rank of 1 for NS, but lost to DMOES in *acc* and *stab* performance measures. The DMOES ranked first for *acc* and *stab*, but was inferior in terms of spacing and diversity of solutions. It is clear that no DMOA was able to win across all the problems and environment types since each algorithm has losses assigned to them. While the ranking algorithm is useful in evaluation of the various DMOAs, it still might be useful to visually inspect DMOAs' ability to track the true POF. Figure 7.1 depicts six different benchmark functions to visually explain the differences between each algorithm.

Table 7.9 Overall wins and losses for various performance measures across all n_t-τ_t combinations

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**DMOAs**			
				MGPSO	**DMOES**	**DNSGA-II**	**SGEA**
all	all	*S*	Wins	205.35	74.97	49.25	161.01
all	all	*S*	Losses	45.22	164.43	198.32	82.61
all	all	*S*	Diff	160.13	-89.46	-149.07	78.4
all	all	*S*	Rank	**1**	3	4	2
all	all	*VD*	Wins	176.81	112.51	61.43	148.54
all	all	*VD*	Losses	73.34	134.61	190.05	101.29
all	all	*VD*	Diff	103.47	-22.1	-128.62	47.25
all	all	*VD*	Rank	**1**	3	4	2
all	all	*MS*	Wins	183.32	83.55	120.77	83.1
all	all	*MS*	Losses	46.64	178.56	87.6	157.94
all	all	*MS*	Diff	136.68	-95.01	33.17	-74.84
all	all	*MS*	Rank	**1**	4	2	3
all	all	*acc*	Wins	161.34	215.24	37.9	127.65
all	all	*acc*	Losses	105.46	64.17	234.55	137.95
all	all	*acc*	Diff	55.88	151.07	-196.65	-10.3
all	all	*acc*	Rank	2	**1**	4	3
all	all	*stab*	Wins	96.63	116.53	57.7	63.39
all	all	*stab*	Losses	70.06	59	109.88	95.31
all	all	*stab*	Diff	26.57	57.53	-52.18	-31.92
all	all	*stab*	Rank	2	**1**	4	3
all	all	*NS*	Wins	111.99	32.13	74.23	141.37
all	all	*NS*	Losses	53.67	183.8	92.77	29.48
all	all	*NS*	Diff	58.32	-151.67	-18.54	111.89
all	all	*NS*	Rank	2	4	3	**1**

The POFs obtained by the DMOAs for FDA1, FDA3$_{Cam}$, dMOP2$_{dec}$, HE2, HE9 and F7 DMOPs are displayed in Figure 7.1. For the FDA1 benchmark function, DMOES and SGEA were the most successful in approximating the true POF. The MGPSO and DNSGA-II algorithms were close, but would require more iterations to get the perfect results. The FDA1 DMOP is a type I problem, where the POS changes over time but the POF remains static. It was one of the first dynamic multi-objective optimisation problems ever created and it is not a particularly hard problem to solve. That is because the POS of FDA1 has line segments parallel to the coordinate axes, and once the DMOA finds the POF at any time t, it usually has no issue tracking the true POF when the environment changes.



(a) FDA1 **(b)** FDA3$_{Cam}$ **(c)** dMOP2$_{dec}$ **(d)** HE2 **(e)** HE9 **(f)** F7

Figure 7.1 Obtained POFs for some of the problems where $n_t = 10$ and $\tau_t = 25$. Each row shows the results for a specific DMOA.

The FDA3$_{Cam}$ modifies the original FDA3 problem by changing the solutions over time in both the decision and the objective space - making it a type II problem. From the Figure 7.1, it is evident that DNSGA-II was the only algorithm that was unable to find the true POF most of the time. Out of the other three DMOAs, DMOES did achieve the best spacing and accuracy. However, it needs to be noted that none of the DMOAs were able to obtain good set of solutions at all times, as indicated by thicker lines (on the right side) of the obtained POFs from the Figure 7.1. The POF at that particular time t was supposed to be spread out evenly along the convex line, not clumped up at one point.

The dMOP2$_{dec}$ problem is classified as a type II problem, where both the POS and the POF change over time. A DMOP with a deceptive POF is a multi-modal problem, since there exist more than one optima and the search space favours the deceptive optimum, which is a local POF and not the global POF [12]. Multi-modal problems are difficult to solve, since a DMOA can get stuck in a local POF. This can be clearly seen in Figure 7.1, where none of the DMOAs were able to find the global optima at all times, as indicated by a straight line slightly above the concave POF at the top.

The HE2 problem is classified as the type III problem, where the POS remains static but the POF changes over time. Unlike the other DMOPs, HE2 has a discontinuous POF with various disconnected but continuous subregions. This time, all of the DMOAs managed to find the true POF at all times, although MGPSO has the most equally spaced solutions.

The HE9 DMOP is also a type III problem, where the POF changes over time but the POS remains the same. What sets apart this problem from the rest is that the POS is defined by a complex function and that the POS is different for each decision variable. Still, each DMOA considered in this study did not have any problems approximating the true POF, as can be clearly seen in Figure 7.1.

The last comparison from Figure 7.1 was performed on the F7 function. This DMOP is classified as a type II problem, where the POS and the POF changes with regards to time. Even though MGPSO, DNSGA-II, DMOES and SGEA were able to obtain the general shape of the POF at certain times, they were never able to successfully change from convex to concave. The DMOES was able to keep track of the true POF at the beginning of the run, but later on would loose track of the POF.

7.4 Summary

This chapter conducted an extensive control parameter sensitivity analysis of the multi-guide particle swarm optimisation (MGPSO) algorithm for dynamic multi-objective optimisation (DMOO). Various balance coefficient initialization strategies were considered and the effect that these approaches had on the performance of the MGPSO was examined. It was determined that the original strategy, but re-initialized each time the environment change was detected, allowed for efficient tracking of the changing Pareto-optimal front. The best performing balance coefficient strategy was then used in a comparative study with DMOES, DNSGA-II, and SGEA. The extensive empirical analysis showed that MGPSO with the re-initializing balance coefficient strategy was highly competitive and oftentimes outperformed the other *state-of-the-art* dynamic optimisation algorithms.

Next chapter conducts detailed analysis of the experimental results for the QPSO strategies.

Chapter 8

Quantum Particle Swarm Optimisation Experiments

The goal of this section is to examine the experimental results for various QPSO strategies at two different quantum proportions. Firstly, six QPSO variants with 50% proportion of quantum particles are compared in Section 8.1. In Section 8.2, the same six QPSO strategies are analyzed, but with 10% proportion of quantum particles. The QPSO algorithms considered are self-adaptive QPSO and PCX QPSO. Each QPSO, as defined in Section 4.3, utilizes three different sampling methods. The goal of the first two experiments is to find the QPSO strategy that performs best across all of the twenty-nine benchmark functions and various n_t-τ_t combinations. In Section 8.3, the best QPSO strategies for both 50% quantum particles (MGPSO_{50}) and 10% quantum particles (MGPSO_{10}) are included in a comparative study with the other *state-of-the-art* DMOAs. The MGPSO without any quantum particles is also included. Performance measures used in the experiments are S, VD, MS, acc, $stab$, and NS as defined in Section 2.3. When the table with results says that $pm = all$, it means that the results of the wins and losses calculations for these six performance measures are combined in the calculation of ranks. Parameter λ_i is re-sampled at each environment change and the re-evaluation of non-dominating solutions archive management strategy is used.

8.1 Results for MGPSO with 50% Proportion of Quantum Particles

Table 8.1 displays the overall results by the various QPSO strategies used by MG-PSO, with 50% proportion of quantum particles. It is evident that strategies that use the randomly selected and tournament selected archive guide in the calculation of a new quantum particle position, outperformed the strategies that uses particle corresponding to the *nbest* position. The self-adaptive QPSO approaches ranked higher than the PCX QPSO approaches, with $\mathbf{PCX}_t$ trailing slightly behind the $\mathbf{QPSO}_n$. Overall, it was $\mathbf{QPSO}_t$ strategy that performed best across all of the twenty-nine benchmark functions. The results are broken down in Table 8.2 and Table 8.3 into overall wins and losses by the frequencies and severities of change, and by the performance measures, respectively.

Table 8.1 Overall wins and losses by the various QPSO strategies for all performance measures and n_t-τ_t combinations

Results	**QPSO Strategies**					
	$\mathbf{QPSO}_n$	$\mathbf{QPSO}_r$	$\mathbf{QPSO}_t$	$\mathbf{PCX}_n$	$\mathbf{PCX}_r$	$\mathbf{PCX}_t$
Wins	431.89	926.35	987.47	224.46	314.11	317.14
Losses	700.44	252.95	214.28	821.1	610.57	602.08
Diff	-268.55	673.4	773.19	-596.64	-296.46	-284.94
Rank	3	2	**1**	6	5	4

The results from Table 8.2 match the results from Table 8.1, where $\mathbf{QPSO}_t$ received a rank of 1 and $\mathbf{QPSO}_r$ received a rank of 2 across most environment types. When the frequency of changes was slow, $\mathbf{QPSO}_r$ outperformed $\mathbf{QPSO}_t$, albeit by a small *Diff* value. The $\mathbf{QPSO}_n$ received a rank of 3 only when changes to the environment occurred fast and when the severity of changes was large (i.e. $\mathbf{n_t} = 1$ and $\tau_\mathbf{t} = 10$), and when $\mathbf{n_t} = 10$ and $\tau_\mathbf{t} = 10$. For the other environment types, $\mathbf{QPSO}_n$ ranked behind the $\mathbf{PCX}_t$ or $\mathbf{PCX}_r$. The $\mathbf{PCX}_n$ ranked last for every n_t-τ_t combination. The reason why the archive based QPSO approaches outperformed the *nbest* QPSO approaches is that archive particles are always non-dominated trade-off solutions and can be considered the most optimal solutions across all of the particles at time t. It is more likely that the solution from the bounded archive is in the range of the most optimal regions of the search space, so when the quantum particles are sampled around the selected archive solution, they can more effectively exploit those regions.

Table 8.2 Overall wins and losses for various frequencies and severities of change across all performance measures

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**QPSO Strategies**					
				$\mathbf{QPSO}_n$	$\mathbf{QPSO}_r$	$\mathbf{QPSO}_t$	$\mathbf{PCX}_n$	$\mathbf{PCX}_r$	$\mathbf{PCX}_t$
10	10	all	Wins	111.76	266.07	297.06	48.86	79.94	83
10	10	all	Losses	196.08	64.66	45.46	238.15	174	168.34
10	10	all	Diff	-84.32	201.41	251.6	-189.29	-94.06	-85.34
10	10	all	Rank	3	2	**1**	6	5	4
10	25	all	Wins	36.82	110.02	111.4	20.41	30.25	30.8
10	25	all	Losses	76.36	22.07	20.45	86.82	67.13	66.87
10	25	all	Diff	-39.54	87.95	90.95	-66.41	-36.88	-36.07
10	25	all	Rank	5	2	**1**	6	4	3
10	50	all	Wins	17.19	51.05	50.84	12.79	15.74	14.98
10	50	all	Losses	39.15	11.52	11.4	38.47	31.77	30.28
10	50	all	Diff	-21.96	39.53	39.44	-25.68	-16.03	-15.3
10	50	all	Rank	5	**1**	2	6	4	3
1	10	all	Wins	165.82	240.12	246.8	90.66	114.51	111.84
1	10	all	Losses	196.11	95.38	91.44	247.09	169.39	170.34
1	10	all	Diff	-30.29	144.74	155.36	-156.43	-54.88	-58.5
1	10	all	Rank	3	2	**1**	6	4	5
20	10	all	Wins	100.3	259.09	281.37	51.74	73.67	76.52
20	10	all	Losses	192.74	59.32	45.53	210.57	168.28	166.25
20	10	all	Diff	-92.44	199.77	235.84	-158.83	-94.61	-89.73
20	10	all	Rank	4	2	**1**	6	5	3

Table 8.3 breaks down the results by each performance measure. This time, $\mathbf{QPSO}_t$ did win for S, MS, *acc*, *stab*, and NS metrics, but for the VD, it received a rank of 2. The VD metric measures the accuracy of the found POF by estimating how close it is to the true POF, and it was $\mathbf{QPSO}_r$ that received a rank of 1 for this metric. The $\mathbf{PCX}_t$ and $\mathbf{PCX}_r$ outperformed the $\mathbf{QPSO}_n$ for *stab* and VD performance measures, respectively. For the other metrics, PCX approaches were consistently outperformed by the self-adaptive QPSO approaches. The $\mathbf{PCX}_n$ ranked last for most metrics and is a clearly inferior strategy compared to the PCX approaches that use the archive guide to calculate the new position of a particle.

Table 8.3 Overall wins and losses for various performance measures across all n_t-τ_t combinations

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**QPSO Strategies**					
				QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
all	all	*S*	Wins	73.02	200.67	232.3	26.15	34.5	36.18
all	all	*S*	Losses	132.16	26.4	12.26	162.73	134.33	134.94
all	all	*S*	Diff	-59.14	174.27	220.04	-136.58	-99.83	-98.76
all	all	*S*	Rank	3	2	**1**	6	5	4
all	all	*VD*	Wins	72.17	206.23	184.08	58.78	86.9	86.13
all	all	*VD*	Losses	168.89	54.2	66.28	165.12	115.07	124.73
all	all	*VD*	Diff	-96.72	152.03	117.8	-106.34	-28.17	-38.6
all	all	*VD*	Rank	5	**1**	2	6	3	4
all	all	*MS*	Wins	64.53	53.88	84.2	18.61	29.93	26.85
all	all	*MS*	Losses	46.66	39.07	16.73	75.61	54.21	45.72
all	all	*MS*	Diff	17.87	14.81	67.47	-57	-24.28	-18.87
all	all	*MS*	Rank	2	3	**1**	6	5	4
all	all	*acc*	Wins	114.51	211.73	226.37	55.93	82.93	87.97
all	all	*acc*	Losses	160.09	72.52	62.83	202.72	142.81	138.47
all	all	*acc*	Diff	-45.58	139.21	163.54	-146.79	-59.88	-50.5
all	all	*acc*	Rank	3	2	**1**	6	5	4
all	all	*stab*	Wins	49.29	65.81	69.86	42.87	43.29	43.18
all	all	*stab*	Losses	73.5	41.29	39.53	63.47	48.39	48.12
all	all	*stab*	Diff	-24.21	24.52	30.33	-20.6	-5.1	-4.94
all	all	*stab*	Rank	6	2	**1**	5	4	3
all	all	*NS*	Wins	58.37	188.03	190.66	22.12	36.56	36.83
all	all	*NS*	Losses	119.14	19.47	16.65	151.45	115.76	110.1
all	all	*NS*	Diff	-60.77	168.56	174.01	-129.33	-79.2	-73.27
all	all	*NS*	Rank	3	2	**1**	6	5	4

8.2 Results for MGPSO with 10% Proportion of Quantum Particles

Table 8.4 displays the overall results by the same QPSO strategies from the last section, but with 10% proportion of quantum particles. The ranks received by each approach closely matches the results from Table 8.1, where **QPSO**$_t$ received a rank of 1, followed by **QPSO**$_r$, **QPSO**$_n$, **PCX**$_t$, **PCX**$_r$, and **PCX**$_n$. However, the *Diff* values were not as large as the ones from the previous experiment since more comparisons were statistically insignificant. Still, the results indicate that even when the proportion of quantum particles is small, the selection of an appropriate sampling method used by QPSO approaches is important. Table 8.5 and Table 8.6 breaks down the results into overall wins and losses by the frequencies and severities of change, and by the performance measures, respectively.

Table 8.4 Overall wins and losses by the various QPSO strategies for all performance measures and n_t-τ_t combinations

Results	QPSO Strategies					
	QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
Wins	246.1	266.48	314.93	140.12	150.51	162.65
Losses	251.37	146.71	139.23	307.37	217.67	218.44
Diff	-5.27	119.77	175.7	-167.25	-67.16	-55.79
Rank	3	2	**1**	6	5	4

Table 8.5 Overall Wins and losses for various frequencies and severities of change across all performance measures

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	QPSO Strategies					
				QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
10	10	all	Wins	68.68	65.95	86.93	29.87	33.34	39.61
10	10	all	Losses	66.36	34.6	30.79	73.22	61.27	58.14
10	10	all	Diff	2.32	31.35	56.14	-43.35	-27.93	-18.53
10	10	all	Rank	3	2	**1**	6	5	4
10	25	all	Wins	19.62	33.93	37.96	9.49	11.46	13.93
10	25	all	Losses	26.06	10.2	9.57	33.15	24.96	22.45
10	25	all	Diff	-6.44	23.73	28.39	-23.66	-13.5	-8.52
10	25	all	Rank	3	2	**1**	6	5	4
10	50	all	Wins	6.58	16.87	19.66	6.5	6.3	6.29
10	50	all	Losses	14.33	5.45	5.75	14.45	11.22	11
10	50	all	Diff	-7.75	11.42	13.91	-7.95	-4.92	-4.71
10	50	all	Rank	5	2	**1**	6	4	3
1	10	all	Wins	106.99	72.09	77.86	65.19	69.77	67.3
1	10	all	Losses	80.22	67.82	66.13	115.42	62.04	67.57
1	10	all	Diff	26.77	4.27	11.73	-50.23	7.73	-0.27
1	10	all	Rank	**1**	4	2	6	3	5
20	10	all	Wins	44.23	77.64	92.52	29.07	29.64	35.52
20	10	all	Losses	64.4	28.64	26.99	71.13	58.18	59.28
20	10	all	Diff	-20.17	49	65.53	-42.06	-28.54	-23.76
20	10	all	Rank	3	2	**1**	6	5	4

Table 8.6 Overall wins and losses for various performance measures across all n_t-τ_t combinations

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**QPSO Strategies**					
				QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
all	all	*S*	Wins	38.44	40.57	51.24	20.14	16.99	16.85
all	all	*S*	Losses	34.29	15.95	15.09	46.94	35.33	36.63
all	all	*S*	Diff	4.15	24.62	36.15	-26.8	-18.34	-19.78
all	all	*S*	Rank	3	2	**1**	6	4	5
all	all	*VD*	Wins	52.69	73.69	76.8	28.66	44.14	44.47
all	all	*VD*	Losses	68.08	31.85	37.19	81.82	49.61	51.9
all	all	*VD*	Diff	-15.39	41.84	39.61	-53.16	-5.47	-7.43
all	all	*VD*	Rank	5	**1**	2	6	3	4
all	all	*MS*	Wins	26.14	14.46	27.05	15	10.09	15.44
all	all	*MS*	Losses	15.64	19.96	13.16	24.64	18.19	16.59
all	all	*MS*	Diff	10.5	-5.5	13.89	-9.64	-8.1	-1.15
all	all	*MS*	Rank	2	4	**1**	6	5	3
all	all	*acc*	Wins	64.32	70.54	82.37	32.37	41.54	44.3
all	all	*acc*	Losses	66.44	42.24	37.04	76.48	56.86	56.38
all	all	*acc*	Diff	-2.12	28.3	45.33	-44.11	-15.32	-12.08
all	all	*acc*	Rank	3	2	**1**	6	5	4
all	all	*stab*	Wins	21.75	22.34	25.01	24.24	21.3	22.24
all	all	*stab*	Losses	36.89	19.01	18.8	25.75	18.77	17.66
all	all	*stab*	Diff	-15.14	3.33	6.21	-1.51	2.53	4.58
all	all	*stab*	Rank	6	3	**1**	5	4	2
all	all	*NS*	Wins	42.76	44.88	52.46	19.71	16.45	19.35
all	all	*NS*	Losses	30.03	17.7	17.95	51.74	38.91	39.28
all	all	*NS*	Diff	12.73	27.18	34.51	-32.03	-22.46	-19.93
all	all	*NS*	Rank	3	2	**1**	6	5	4

From Table 8.5, it can be concluded that when there are only 10% of quantum particles utilized, the **QPSO**$_t$ outperforms the other approaches on most environment types. However, when the severity of changes was large, it was **QPSO**$_n$ that received a rank of 1. One explanation is that the **QPSO**$_n$ approach is better at exploring the search space, because when $\mathbf{n_t} = 1$ and $\tau_\mathbf{t} = 10$, the optimal regions drastically change in location. Each of the self-adaptive QPSO approaches consistently outranked PCX QPSO approaches, except for **QPSO**$_r$ that lost to **PCX**$_r$ when $\mathbf{n_t} = 1$ and $\tau_\mathbf{t} = 10$. The **PCX**$_t$ received a rank of 4 on most environment types, followed by **PCX**$_r$, and then by **PCX**$_n$ which ranked last.

Table 8.6 breaks down the results by each performance measure. The results indicate that for all of the twenty-nine benchmark functions and five environment types, the **QPSO**$_t$ received a rank of 1 on most performance measures. For VD metric, it was **QPSO**$_r$ that won against the other strategies. Interestingly, **QPSO**$_n$ lost to PCX approaches on VD and $stab$, but did better on S, MS, acc, and NS performance measures. Since **QPSO**$_t$ received a rank of 1 for most metrics, it can be concluded that self-adaptive QPSO approach, sampled around tournament selected archive solution, was the most successful strategy when faced with a diversity loss. Consequently, it allowed for efficient tracking of the true POF and will be used in a final comparative study against the other DMOAs.

8.3 Comparisons with the other DMOAs

Based on the findings from the previous section on the QPSO strategies, it can be concluded that QPSO$_t$, for most DMOPs, was the most effective strategy. Therefore, QPSO$_t$ at 50% (MGPSO$_{50}$) and 10% (MGPSO$_{10}$) of quantum particles will be used in the comparative study against the other DMOAs. The DMOA's used in this study are DMOES, DNSGA-II, and SGEA. The baseline MGPSO without any quantum particles is also included, and will be referred to as MGPSO for the rest of the paper.

With regards to Table 8.7, it is clear that MGPSO$_{10}$ vastly outperformed the MGPSO$_{50}$ and the baseline MGPSO. The difference between the other DMOAs was even more significant. Table 8.7 itself does not provide any information with regards to the performance measures or individual n_t-τ_t combinations. Therefore, Table 8.8 should be examined more closely to evaluate the performance of DMOAs with respect to the n_t-τ_t combinations, or Table 8.9 for results with respect to S, VD, MS, acc, $stab$, and NS performance measures.

Table 8.7 Overall wins and losses by the various DMOAs across all performance measures and n_t-τ_t combinations

Results	**DMOAs**					
	MGPSO	**MGPSO$_{50}$**	**MGPSO$_{10}$**	**DMOES**	**DNSGA-II**	**SGEA**
Wins	1091.4	1164.97	1203.38	918.74	555.87	1019.47
Losses	599.77	627.71	453.57	1466.15	1613.14	1193.49
Diff	491.63	537.26	749.81	-547.41	-1057.27	-174.02
Rank	3	2	**1**	5	6	4

Table 8.8 Overall wins and losses for various frequencies and severities of change across all performance measures

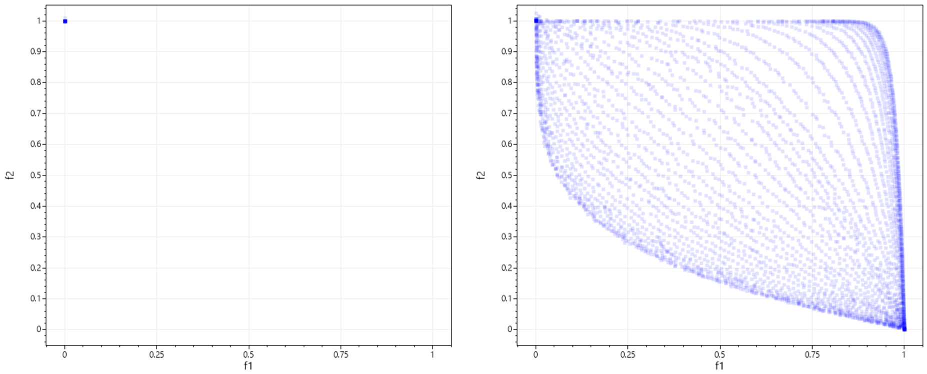
$\mathbf{n_t}$	$\tau_\mathbf{t}$	**Results**	**DMOAs**					
			MGPSO	$\mathbf{MGPSO_{50}}$	$\mathbf{MGPSO_{10}}$	**DMOES**	**DNSGA-II**	**SGEA**
10	10	Wins	309.19	328.69	343.51	266.07	139.56	266.06
10	10	Losses	153.49	168.29	113	395.73	471.2	351.37
10	10	Diff	155.7	160.4	230.51	-129.66	-331.64	-85.31
10	10	Rank	3	2	**1**	5	6	4
10	25	Wins	111.83	117.65	124.85	120.97	60.74	123.28
10	25	Losses	69.82	78.38	56.03	146.49	182.67	125.93
10	25	Diff	42.01	39.27	68.82	-25.52	-121.93	-2.65
10	25	Rank	2	3	**1**	5	6	4
10	50	Wins	52.82	54.21	58.64	61.2	34.3	65.29
10	50	Losses	37.52	42.61	32.64	70.55	83.78	59.36
10	50	Diff	15.3	11.6	26	-9.35	-49.48	5.93
10	50	Rank	2	3	**1**	5	6	4
1	10	Wins	324.83	358.71	350.97	170.46	184.45	301.32
1	10	Losses	179.25	161.18	129.85	500.36	412.29	307.81
1	10	Diff	145.58	197.53	221.12	-329.9	-227.84	-6.49
1	10	Rank	3	2	**1**	6	5	4
20	10	Wins	292.73	305.71	325.41	300.04	136.82	263.52
20	10	Losses	159.69	177.25	122.05	353.02	463.2	349.02
20	10	Diff	133.04	128.46	203.36	-52.98	-326.38	-85.5
20	10	Rank	2	3	**1**	4	6	5

It is evident from the results in Table 8.8 that both the MGPSO with quantum particles and the MGPSO without any quantum particles outperformed the other DMOAs. The MGPSO_{10} received a rank of 1 for all of the environment types, followed by the MGPSO_{50} and the baseline MGPSO. The MGPSO_{50} received a rank of 2 when $n_t = 10$ and $\tau_t = 10$, and when $n_t = 1$ and $\tau_t = 10$. The baseline MGPSO ranked second for the other environment types. Overall, the *Diff* values for both MGPSO and MGPSO_{50} were close, except when the severity and frequency of changes were high. These results are not surprising since quantum particles are effective way to explore the search space and this type of environment requires DMOA to exhibit strong exploratory capabilities to be successful. The SGEA consistently outperformed DMOES and DNSGA-II on all environment types except when $n_t = 20$ and $\tau_t = 10$. The DNSGA-II ranked last on most environment types, followed by DMOES.

Table 8.9 shows that MGPSO_{10} outperformed the other DMOAs for S and VD metrics. The MGPSO_{50} received a rank of 1 for the MS value, which shows that it had a good spread of solutions. The DMOES received a rank of 1 for the *acc* and *stab* metrics, and when it comes to the NS value, it was SGEA that outperformed the other DMOAs. However, a high NS value does not mean much when the accuracy of solutions is low. The results also show that it is better when a small proportion of neutral particles is converted into quantum particles, as taking away too many of the neutral particles deteriorates the performance. The positive effect of adding quantum particles to the MGPSO is visualized in Figure 8.1. The figure clearly shows that MGPSO without any quantum particles would sometimes converge at a single point and was then unable to approximate the POF. The self-adapting quantum particles were successful at exploring the nearby regions and helped MGPSO overcome the diversity loss.

Table 8.9 Overall wins and losses for various performance measures across all n_t-τ_t combinations

PM	**Results**	**DMOAs**					
		MGPSO	**MGPSO$_{50}$**	**MGPSO$_{10}$**	**DMOES**	**DNSGA-II**	**SGEA**
S	Wins	237.15	244.1	238.27	90.87	66.6	210.69
S	Losses	74.31	83.76	63.76	316.25	354.34	195.26
S	Diff	162.84	160.34	174.51	-225.38	-287.74	15.43
S	Rank	2	3	**1**	5	6	4
VD	Wins	227.34	203.31	238.36	159	83.91	219.9
VD	Losses	108.58	153.68	89.36	241.56	334.69	203.95
VD	Diff	118.76	49.63	149	-82.56	-250.78	15.95
VD	Rank	2	3	**1**	5	6	4
MS	Wins	180.6	228.84	219.65	116.87	136.43	97.5
MS	Losses	83.91	39.35	40.18	320.8	184.58	311.07
MS	Diff	96.69	189.49	179.47	-203.93	-48.15	-213.57
MS	Rank	3	**1**	2	5	4	6
acc	Wins	194.89	208.43	235.52	320.98	72.67	193.41
acc	Losses	157.92	178.83	116.45	142.94	381.72	248.04
acc	Diff	36.97	29.6	119.07	178.04	-309.05	-54.63
acc	Rank	3	4	2	**1**	6	5
$stab$	Wins	117.64	121.5	116.27	182.35	96.14	100.84
$stab$	Losses	90.87	91.17	84.81	114.35	186.37	167.17
$stab$	Diff	26.77	30.33	31.46	68	-90.23	-66.33
$stab$	Rank	4	3	2	**1**	6	5
NS	Wins	133.78	158.79	155.31	48.67	100.12	197.13
NS	Losses	84.18	80.92	59.01	330.25	171.44	68
NS	Diff	49.6	77.87	96.3	-281.58	-71.32	129.13
NS	Rank	4	3	2	6	5	**1**



(a) MGPSO **(b)** MGPSO_{10}

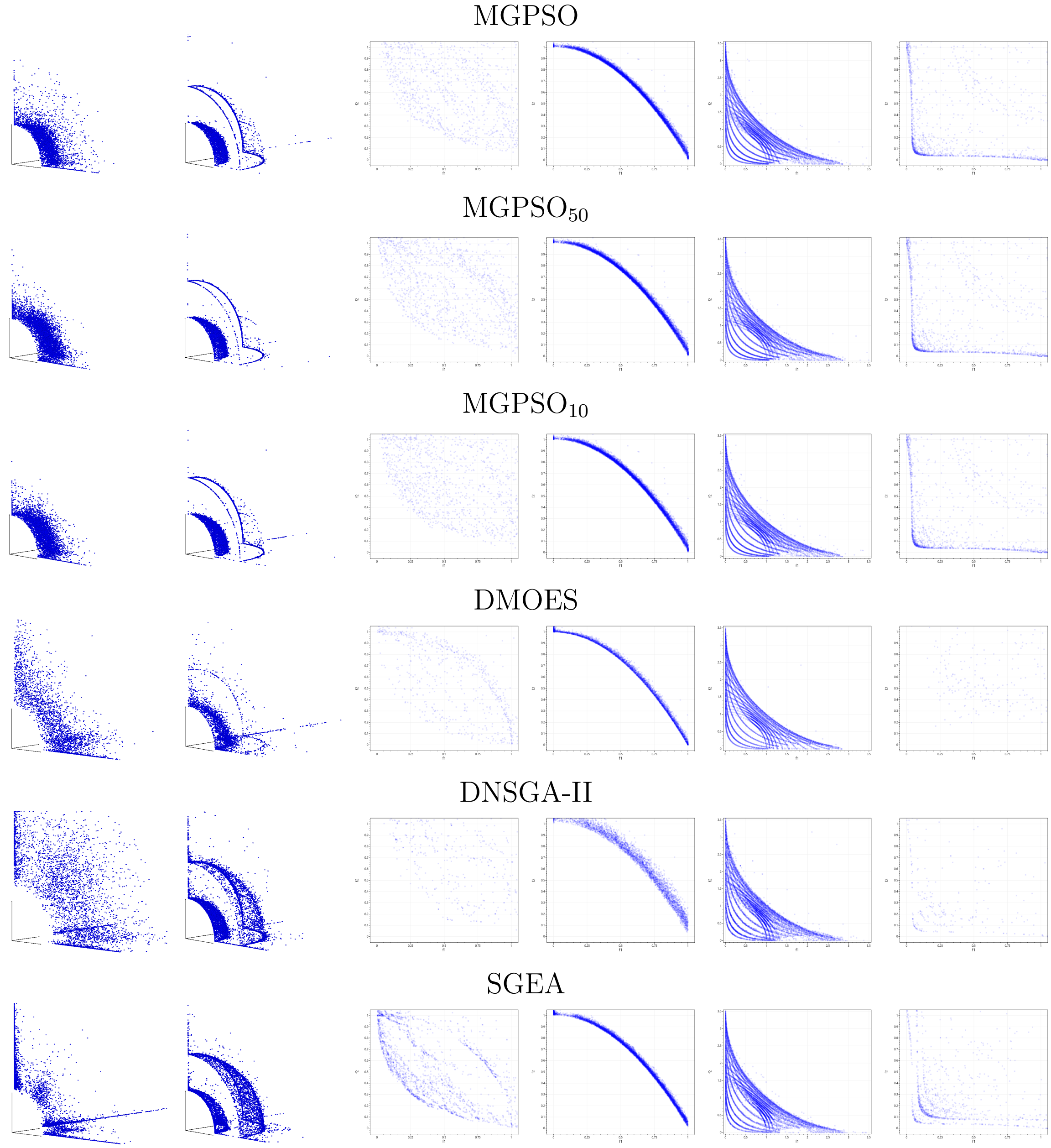
Figure 8.1 Obtained POFs for FDA2_{Cam} DMOP by MGPSO and MGPSO_{10}. Figure on the left depicts all of the MGPSO particles converged at a single point. The MGPSO with quantum particles was able to overcome the diversity loss.

While the ranking algorithm is useful in evaluating various DMOAs, it still is helpful to visually inspect DMOAs' ability to track the true POF. Figure 8.2 depicts POFs of six benchmark functions to visually explain the differences between each algorithm. These benchmark functions include: FDA4, FDA5_{dec}, F6, DIMP1, DF4 and DF6. The FDA4 and FDA5_{dec} DMOPs have three objectives where $n_t = 1$ and $\tau_t = 10$, while the rest have two objectives where $n_t = 1$ and $\tau_t = 10$.

For FDA4 DMOP, DMOES, DNSGA-II, and SGEA were unable to approximate the true POF as well as MGPSO, MGSPO_{50}, and MGSPO_{10}. The MGPSO with quantum particles were the most successful approaches since the particles were less scattered across the objective space. This is a type I DMOP.

The FDA5_{dec} modifies the original FDA5 problem by making it a DMOP with at least two optima, but the search space favours the deceptive one [12]. Similar to FDA4, FDA5_{dec} has a non-convex POF and is a tri-objective DMOP. This time, SGEA was able to successfully track the POF with a high rate of accuracy. The POF along the outer sphere for MGPSO, MGPSO_{50}, MGPSO_{10}, and DMOES was not populated with evenly spaced solutions.

Next comparison was performed on the F6 function. This DMOP is classified as a type II problem, where the POS and the POF changes with regards to time. None of the DMOAs were able to keep track of the true POF, as indicated by the scattered solutions in the objective space. This is a difficult problem to solve and typically requires more iterations to successfully track the POF as it changes from convex to concave.



(a) FDA4 **(b)** FDA5$_{dec}$ **(c)** F6 **(d)** DIMP1 **(e)** DF4 **(f)** DF6

Figure 8.2 Obtained POFs for some of the problems where $n_t = 1$ and $\tau_t = 10$ for FDA4 and FDA5$_{dec}$, and $n_t = 10$ and $\tau_t = 10$ for the rest of the problems. Each row shows the results for a specific DMOA.

Both DIMP1 and DF4 DMOPs were successfully optimised by all DMOAs considered in this study since all of the results from Figure 8.2 closely resemble the true POF for these benchmark functions. Only DNSGA-II was not as close to the true POF for DIMP1 DMOP as the other DMOAs. The DIMP1 is a type I problem, whereas DF4 is a type II problem.

The last comparison from Figure 8.2 was performed on the DF6 function. This is a type II problem, but unlike the other DMOPs, its POF contains knee regions and long tails that are known to be a challenging property [24], [25]. While MGPSO, MGPSO_{50}, MGPSO_{10}, and SGEA were able to somewhat approximate the true POF at one point, none of the approaches were accurate at all times.

8.4 Summary

This chapter conducted an extensive analysis of the quantum particle swarm optimisation (QPSO) strategies for the multi-guide particle swarm optimisation (MGPSO) algorithm used for dynamic multi-objective optimisation (DMOO). Two QPSO approaches with different proportion of quantum particles, as well as three sampling methods used by each were considered. The effect that these strategies had on the performance of the MGPSO was examined, and it was determined that self-adaptive quantum particles, with a normal distribution centered at a tournament selected archive guide, allowed for efficient tracking of the changing Pareto-optimal front. The PCX QPSO was outperformed by the self-adaptive QPSO across all of the performance measures.

The best performing QPSO strategies at both 50% and 10% proportion of quantum particles were then used in a comparative study against a baseline MGPSO, DMOES, DNSGA-II, and SGEA. The extensive empirical analysis showed that, across every environment type combination, the MGPSO with 10% of self-adaptive quantum particles received a rank of 1, followed by the MGPSO with 50% of self-adaptive quantum particles, and then by the baseline MGPSO that did not use any quantum particles. The results indicate that a small proportion of quantum particles used by the MGPSO is desirable. The SGEA received a rank of 4, followed by DMOES and then by DNSGA-II. In conclusion, the results showed that MGPSO with self-adaptive quantum particles was highly competitive and oftentimes outperformed the other *state-of-the-art* dynamic optimisation algorithms.

Next chapter concludes the thesis and the suggested future work is provided.

Chapter 9

Conclusion and Future Directions

This chapter summarises the major findings of the work done in this study, and provides a number of suggestions for future work that can be pursued as a result of this work. Section 9.1 presents the summary of findings and conclusions, followed by suggestions for future work in Section 9.2.

9.1 Summary of Conclusions

This study aimed to adapt the multi-guide particle swarm optimization (MGPSO) algorithm to solve dynamic multi-objective optimization (DMOO) problems. The MGPSO is a multi-objective particle swarm-based algorithm, originally used for static multi-objective optimisation problems, where each sub-swarm optimises one of the objectives.

Background on DMOO was provided. It included sections on Pareto-optimal set (POS) and Pareto-optimal front (POF), and the main goal when solving the dynamic multi-objective optimisation problems (DMOPs) was given. Moreover, various dynamic environment types, the current DMOPs, and the current performance measures used to evaluate the dynamic multi-objective optimisation algorithms (DMOAs) was described in detail. The test sets and performance measures presented were used throughout this study to benchmark the performance of various DMOAs and variations of the MGPSO algorithm.

An investigation of the alternative archive management update approaches, and the effect these approaches have on the exploration and exploitation of the feasible regions of the search space has led to development of the local search archive management strategy. It was determined that a hill climber with a decreasing step size performed better than the other hill climbing strategies on most environment type combinations. Only when the spatial severity was large and the frequency of changes was fast, the hill climber with a fixed step size of 10 outranked the other approaches. The archive clearing approach ranked last and the re-evaluation of non-dominated solutions approach ranked second last.

The best performing local search strategy was then used in a comparative study against a baseline MGPSO, DMOES, DNSGA-II, and SGEA. The baseline MGPSO used a re-evaluation approach to make the comparisons against the other DMOAs fair as the local search strategy takes more function evaluations than competing algorithms. The extensive empirical analysis showed that baseline MGPSO received a rank of 2 across all environment type combinations. The SGEA received a rank of 3, followed by DMOES and then by DNSGA-II. The MGPSO with a local search archive management strategy received a rank of 1 in every performance measure considered in this study. In conclusion, the extensive empirical analysis showed that baseline MGPSO and MGPSO with a local search performed on-par, or exceeded competing algorithms in terms of performance.

The next step involved investigation of the current archive balance coefficient initialization strategies. The analysis has led to development of three variants of the balance coefficient parameter that take into account the dynamic nature of the problems. An extensive control parameter sensitivity analysis of the nine approaches was conducted and the effect that these strategies had on the performance of the MGPSO was examined. It was determined that the original strategy, but re-initialized each time the environment change was detected, allowed for efficient tracking of the changing POF and it outperformed the other strategies. The best performing balance coefficient strategy was then used in a comparative study with the other DMOAs and the results have shown that MGPSO exceeded competing algorithms across every environment type.

The last step involved an extensive analysis of the quantum particle swarm optimisation (QPSO) strategies. The self-adaptive QPSO and parent centric crossover QPSO were considered, and sampling methods used by each were defined. The investigation of the bounded archive used by the MGPSO has led to development of two alternative sampling methods, where the archive guide is used in the calculation of the new particle's position. In total, six QPSO variants were considered in the experimental analyses, where MGPSO at both 50% and 10% proportion of quantum particles was examined. The effect that these strategies had on the performance of the MGPSO was explored, and it was determined that self-adaptive QPSO, with a normal distribution centered at a tournament selected archive guide, allowed for efficient tracking of the changing POF. The best performing QPSO strategies from both experiments were then used in a comparative study with a baseline MGPSO as well as the other DMOAs. The results indicate that MGPSO with 10% self-adaptive particles outperformed the other algorithms.

In conclusion, the lessons learned from the three experiments conducted in this study were combined into the final version of the MGPSO. The improvements gained from the local-search archive management strategy, combined with re-initialization approach to the archive balance coefficient as well as the addition of 10% of the self-adaptive quantum particles, has demonstrated that the MGPSO is capable of solving DMOPs. Overall, twenty-nine benchmark functions and six performance measures were used to evaluate DMOAs included in this study. The experiments were run against five different environment types where both the frequency of change and the severity of change parameters controlled how often and how big the changes were during the optimisation process. The overwhelming evidence presented in this study indicate that the MGPSO is highly competitive and performs on-par, or exceeds the performance of the competing algorithms.

9.2 Future Work

Throughout this study, several new ideas for future research have been identified. A summary of each of these ideas is given below.

Dynamic Many-objective Optimisation

While the adapted MGPSO is clearly capable of solving dynamic multi-objective optimisation problems, the effectiveness of the MGPSO for dynamic many-objective optimisation is still unknown. Therefore, a scalability of the adapted MGPSO to dynamic many-objective optimisation problems should be considered. The key difference between many-objective optimisation and multi-objective optimisation is in the number of objectives being optimised. Multi-objective optimisation considers only two or three objectives whereas many-objective optimisation involves more than three objectives. This change will require an alternative to the Pareto-dominance approach that was extensively used in this study, since Pareto-based DMOAs are known to suffer from the deterioration of searchability on many-objective problems.

Empirical Study of Different Topologies

In this study, the local best topology of size 3 was used by the MGPSO as it showed a significant improvement over the global best topology. However, a more comprehensive empirical analysis of different topologies and the effect they have on the performance of the MGPSO for DMOPs should be considered. Neighbourhood topologies such as the local best allows for a slower information exchange and a larger coverage of the search space by particles when compared to a fully connected topology.

Effectiveness of MGPSO on Constrained DMOO Real-world Problems

Only unconstrained dynamic multi-objective optimisation problems were considered in this study. These type of DMOPs only have one boundary constraint where the values of the decision variables beyond the defined bounds are not allowed. However, the effectiveness of the MGPSO on constrained DMOO real-world problems should be investigated. These type of problems often involve more than one constraint, and are generally more difficult to optimise. Both discrete and real optimisation problems should be considered.

Bibliography

[1] Frederick D. Atiah and Mardé Helbig. “Effects of Decision Models on Dynamic Multi-objective Optimization Algorithms for Financial Markets”. In: *Proceedings of the IEEE Congress on Evolutionary Computation*. IEEE, 2019, pp. 762–770. DOI: https://doi.org/10.1109/CEC.2019.8790275.

[2] Kalyanmoy Deb, N. Udaya Bhaskara Rao, and Karthik Sindhya. “Dynamic Multi-objective Optimization and Decision-Making Using Modified NSGA-II: A Case Study on Hydro-thermal Power Scheduling”. In: *Proceedings of the Evolutionary Multi-Criterion Optimization, 4th International Conference*. Vol. 4403. Lecture Notes in Computer Science. Springer, 2006, pp. 803–817. DOI: https://doi.org/10.1007/978-3-540-70928-2_60.

[3] Dujuan Wang, Feng Liu, and Yaochu Jin. “A multi-objective evolutionary algorithm guided by directed search for dynamic scheduling”. In: *Comput. Oper. Res.* 79 (2017), pp. 279–290. DOI: https://doi.org/10.1016/j.cor.2016.04.024.

[4] Goshgar Ismayilov and Haluk R. Topcuoglu. “Neural network based multi-objective evolutionary algorithm for dynamic workflow scheduling in cloud computing”. In: *Future Gener. Comput. Syst.* 102 (2020), pp. 307–322. DOI: https://doi.org/10.1016/j.future.2019.08.012.

[5] Hitoshi Kanoh and Kenta Hara. “Hybrid genetic algorithm for dynamic multi-objective route planning with predicted traffic in a real-world road network”. In: *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, 2008, pp. 657–664. DOI: https://doi.org/10.1145/1389095.1389226.

[6] Xuanli Wu, Zheming Ma, and Ye Wang. “Joint User Grouping and Resource Allocation for Multi-User Dual Layer Beamforming in LTE-A”. In: *IEEE Commun. Lett.* 19.10 (2015), pp. 1822–1825. DOI: https://doi.org/10.1109/LCOMM.2015.2458861.

[7] Jinliang Ding, Cuie Yang, Qiong Xiao, Tianyou Chai, and Yaochu Jin. “Dynamic Evolutionary Multiobjective Optimization for Raw Ore Allocation in Mineral Processing”. In: *IEEE Trans. Emerg. Top. Comput. Intell.* 3.1 (2019), pp. 36–48. DOI: https://doi.org/10.1109/TETCI.2018.2812897.

[8] Shouyong Jiang and Shengxiang Yang. “A Steady-State and Generational Evolutionary Algorithm for Dynamic Multiobjective Optimization”. In: *IEEE Transactions on Evolutionary Computation* 21.1 (2017), pp. 65–82. DOI: https://doi.org/10.1109/TEVC.2016.2574621.

[9] Chi K. Goh and Kay C. Tan. “A Competitive-Cooperative Coevolutionary Paradigm for Dynamic Multiobjective Optimization”. In: *IEEE Transactions on Evolutionary Computation* 13.1 (2009), pp. 103–127. DOI: https://doi.org/10.1109/TEVC.2008.920671.

[10] Mardé Greeff and Andries P. Engelbrecht. “Solving dynamic multi-objective problems with vector evaluated particle swarm optimisation”. In: *Proceedings of the IEEE Congress on Evolutionary Computation*. IEEE, 2008, pp. 2917–2924. DOI: https://doi.org/10.1109/CEC.2008.4631190.

[11] Christiaan Scheepers, Andries P. Engelbrecht, and Christopher W. Cleghorn. “Multi-guide particle swarm optimization for multi-objective optimization: empirical and stability analysis”. In: *Swarm Intelligence* 13.3-4 (2019), pp. 245–276. DOI: https://doi.org/10.1007/s11721-019-00171-0.

[12] Mardé Helbig and Andries P. Engelbrecht. “Benchmarks for dynamic multi-objective optimisation”. In: *Proceedings of the IEEE Symposium on Computational Intelligence in Dynamic and Uncertain Environments*. IEEE, 2013, pp. 84–91. DOI: https://doi.org/10.1109/CIDUE.2013.6595776.

[13] Marco Farina, Kalyanmoy Deb, and Paolo Amato. “Dynamic multiobjective optimization problems: test cases, approximations, and applications”. In: *IEEE Transactions on Evolutionary Computation* 8.5 (2004), pp. 425–442. DOI: https://doi.org/10.1109/TEVC.2004.831456.

[14] Jürgen Branke, Erdem Salihoglu, and Sima Uyar. “Towards an analysis of dynamic environments”. In: *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, 2005, pp. 1433–1440. DOI: https://doi.org/10.1145/1068009.1068237.

[15] Eckart Zitzler, Kalyanmoy Deb, and Lothar Thiele. "Comparison of Multiobjective Evolutionary Algorithms: Empirical Results". In: *Evoutionary Computation* 8.2 (2000), pp. 173–195. DOI: https://doi.org/10.1162/106365600568202.

[16] Kalyanmoy Deb, Lothar Thiele, Marco Laumanns, and Eckart Zitzler. "Scalable multi-objective optimization test problems". In: *Proceedings of the Congress on Evolutionary Computation*. Vol. 1. 2002, pp. 825–830. DOI: https://doi.org/10.1109/CEC.2002.1007032.

[17] Aimin Zhou, Yaochu Jin, Qingfu Zhang, Bernhard Sendhoff, and Edward P. K. Tsang. "Prediction-Based Population Re-initialization for Evolutionary Dynamic Multi-objective Optimization". In: *Proceedings of the Evolutionary Multi-Criterion Optimization, 4th International Conference*. Vol. 4403. Lecture Notes in Computer Science. Springer, 2006, pp. 832–846. DOI: https://doi.org/10.1007/978-3-540-70928-2_62.

[18] Mario Cámara, Julio Ortega, and Francisco de Toro. "Approaching Dynamic Multi-Objective Optimization Problems by Using Parallel Evolutionary Algorithms". In: *Advances in Multi-Objective Nature Inspired Computing*. Vol. 272. Studies in Computational Intelligence. Springer, 2010, pp. 63–86. DOI: https://doi.org/10.1007/978-3-642-11218-8_4.

[19] Carlos Coello, Clarisse Dhaenens, and Laetitia Jourdan. *Advances in Multi-Objective Nature Inspired Computing*. Vol. 272. Jan. 2009. DOI: https://doi.org/10.1007/978-3-642-11218-8.

[20] Wee T. Koo, Chi K. Goh, and Kay C. Tan. "A predictive gradient strategy for multiobjective evolutionary algorithms in a fast changing environment". In: *Memetic Computing* 2.2 (2010), pp. 87–110. DOI: https://doi.org/10.1007/s12293-009-0026-7.

[21] Mardé Helbig and Andries P. Engelbrecht. "Analyses of guide update approaches for vector evaluated particle swarm optimisation on dynamic multi-objective optimisation problems". In: *Proceedings of the IEEE Congress on Evolutionary Computation*. IEEE, 2012, pp. 1–8. DOI: https://doi.org/10.1109/CEC.2012.6252882.

[22] Aimin Zhou, Yaochu Jin, and Qingfu Zhang. "A Population Prediction Strategy for Evolutionary Dynamic Multiobjective Optimization". In: *IEEE Transactions on Cybernetics* 44.1 (2014), pp. 40–53. DOI: https://doi.org/10.1109/TCYB.2013.2245892.

[23] Shouyong Jiang, Shengxiang Yang, Xin Yao, Kay C. Tan, Marcus Kaiser, and Natalio Krasnogor. *Benchmark Problems for CEC2018 Competition on Dynamic Multiobjective Optimisation*. Tech. rep. Newcastle University, School of Computing, 2017. URL: http://homepages.cs.ncl.ac.uk/shouyong.jiang/cec2018/CEC2018_Tech_Rep_DMOP.pdf.

[24] Shouyong Jiang and Shengxiang Yang. "An Improved Multiobjective Optimization Evolutionary Algorithm Based on Decomposition for Complex Pareto Fronts". In: *IEEE Transactions on Cybernetics* 46.2 (2016), pp. 421–437. DOI: https://doi.org/10.1109/TCYB.2015.2403131.

[25] Yutao Qi, Xiaoliang Ma, Fang Liu, Licheng Jiao, Jianyong Sun, and Jianshe Wu. "MOEA/D with Adaptive Weight Adjustment". In: *Evolutionary Computation* 22.2 (2014), pp. 231–264. DOI: https://doi.org/10.1162/EVCO_a_00109.

[26] Mardé Helbig and Andries P. Engelbrecht. "Performance measures for dynamic multi-objective optimisation algorithms". In: *Information Sciiences* 250 (2013), pp. 61–81. DOI: https://doi.org/10.1016/j.ins.2013.06.051.

[27] Jason Schott. "Fault tolerant design using single and multicriteria genetic algorithm optimization". MA thesis. Massachusetts Institute of Technology, 1995. URL: https://dspace.mit.edu/handle/1721.1/11582.

[28] Eckart Zitzler. "Evolutionary algorithms for multiobjective optimization: methods and applications". PhD thesis. University of Zurich, Zürich, Switzerland, 1999. URL: http://d-nb.info/95814172X.

[29] Mario Cámara, Julio Ortega, and Francisco de Toro. "A single front genetic algorithm for parallel multi-objective optimization in dynamic environments". In: *Neurocomputing* 72.16-18 (2009), pp. 3570–3579. DOI: https://doi.org/10.1016/j.neucom.2008.12.041.

[30] Karsten Weicker. "Performance Measures for Dynamic Environments". In: *Proceedings of the Parallel Problem Solving from Nature*. Vol. 2439. Lecture Notes in Computer Science. Springer, 2002, pp. 64–76. DOI: https://doi.org/10.1007/3-540-45712-7_7.

[31] Mario Cámara, Julio Ortega, and Francisco de Toro. "Parallel Processing for Multi-objective Optimization in Dynamic Environments". In: *Proceedings of the IEEE International Parallel and Distributed Processing Symposium*. 2007, pp. 1–8. DOI: https://doi.org/10.1109/IPDPS.2007.370433.

[32] James Kennedy and Russell Eberhart. “Particle swarm optimization”. In: *Proceedings of the International Conference on Neural Networks*. Vol. 4. 1995, 1942–1948 vol.4.

[33] Yu-Hui Shi and Russell Eberhart. “A modified particle swarm optimizer”. In: *Proceedings of the IEEE International Conference on Evolutionary Computation*. 1998, pp. 69–73. DOI: https://doi.org/10.1109/ICEC.1998.699146.

[34] Wei Chu, Xiaogang Gao, and Soroosh Sorooshian. “Handling boundary constraints for particle swarm optimization in high-dimensional search space”. In: *Inf. Sci.* 181.20 (2011), pp. 4569–4581. DOI: https://doi.org/10.1016/j.ins.2010.11.030.

[35] Andries P. Engelbrecht. *Fundamentals of Computational Swarm Intelligence*. Wiley, 2005. ISBN: 978-0-470-09191-3.

[36] Anthony Carlisle and Gerry Dozier. “An off-the-shelf pso”. In: *Proceedings of the Workshop on Particle Swarm Optimization*. 2001.

[37] James Kennedy. “Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance”. In: *Proceedings of the Congress on Evolutionary Computation*. Vol. 3. 1999, pp. 1931–1938. DOI: https://doi.org/10.1109/CEC.1999.785509.

[38] James Kennedy and Rui Mendes. “Population structure and particle swarm performance”. In: *Proceedings of the Congress on Evolutionary Computation*. Vol. 2. 2002, pp. 1671–1676. DOI: https://doi.org/10.1109/CEC.2002.1004493.

[39] Rui Mendes, James Kennedy, and José Neves. “Watch thy neighbor or how the swarm can learn from its environment”. In: *Proceedings of the Swarm Intelligence Symposium*. IEEE, 2003, pp. 88–94. DOI: https://doi.org/10.1109/SIS.2003.1202252.

[40] Andries P. Engelbrecht, B. S. Masiye, and Gary Pamparà. “Niching ability of basic particle swarm optimization algorithms”. In: *Proceedings of the Swarm Intelligence Symposium*. IEEE, 2005, pp. 397–400. DOI: https://doi.org/10.1109/SIS.2005.1501650.

[41] Dongshu Wang, Dapei Tan, and Lei Liu. “Particle swarm optimization algorithm: an overview”. In: *Soft Computing* 22.2 (2018), pp. 387–408. DOI: https://doi.org/10.1007/s00500-016-2474-6.

[42] Elre T. Oldewage, Andries P. Engelbrecht, and Christopher W. Cleghorn. "Degrees of stochasticity in particle swarm optimization". In: *Swarm Intelligence* 13.3-4 (2019), pp. 193–215. DOI: https://doi.org/10.1007/s11721-019-00168-9.

[43] Kalyanmoy Deb, Samir Agrawal, Amrit Pratap, and T. Meyarivan. "A fast and elitist multiobjective genetic algorithm: NSGA-II". In: *IEEE Transactions on Evolutionary Computation* 6.2 (2002), pp. 182–197. DOI: https://doi.org/10.1109/4235.996017.

[44] Christiaan Scheepers. "Multi-guided particle swarm optimization: A multi-objective particle swarm optimizer". PhD thesis. University of Pretoria, 2017. URL: https://repository.up.ac.za/handle/2263/64041.

[45] Gary Pamparà and Andries P. Engelbrecht. "Self-adaptive Quantum Particle Swarm Optimization for Dynamic Environments". In: *Proceedings of the Swarm Intelligence, 11th International Conference*. Vol. 11172. Lecture Notes in Computer Science. Springer, 2018, pp. 163–175. DOI: https://doi.org/10.1007/978-3-030-00533-7_13.

[46] Kyle R. Harrison, Beatrice M. Ombuki-Berman, and Andries P. Engelbrecht. "A radius-free quantum particle swarm optimization technique for dynamic optimization problems". In: *Proceedings of the IEEE Congress on Evolutionary Computation*. IEEE, 2016, pp. 578–585. DOI: https://doi.org/10.1109/CEC.2016.7743845.

[47] Kai Zhang, Chaonan Shen, Xiaoming Liu, and Gary G. Yen. "Multiobjective Evolution Strategy for Dynamic Multiobjective Optimization". In: *IEEE Transactions on Evolutionary Computation* 24.5 (2020), pp. 974–988. DOI: https://doi.org/10.1109/TEVC.2020.2985323.

[48] Kyle Erwin and Andries P. Engelbrecht. "Control parameter sensitivity analysis of the multi-guide particle swarm optimization algorithm". In: *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, 2019, pp. 22–29. DOI: https://doi.org/10.1145/3321707.3321739.

[49] Tim Blackwell and Jürgen Branke. "Multi-swarm Optimization in Dynamic Environments". In: *Proceedings of the Applications of Evolutionary Computing*. Vol. 3005. Lecture Notes in Computer Science. Springer, 2004, pp. 489–500. DOI: https://doi.org/10.1007/978-3-540-24653-4_50.

[50] Kalyanmoy Deb, Dhiraj Joshi, and Ashish Anand. "Real-coded evolutionary algorithms with parent-centric recombination". In: *Proceedings of the Congress on Evolutionary Computation*. Vol. 1. 2002, 61–66 vol.1. DOI: https://doi.org/10.1109/CEC.2002.1006210.

[51] Barend J. Leonard and Andries P. Engelbrecht. "On the optimality of particle swarm parameters in dynamic environments". In: *Proceedings of the IEEE Congress on Evolutionary Computation*. IEEE, 2013, pp. 1564–1569. DOI: https://doi.org/10.1109/CEC.2013.6557748.

[52] Kyle R. Harrison, Andries P. Engelbrecht, and Beatrice M. Ombuki-Berman. "Optimal parameter regions and the time-dependence of control parameter values for the particle swarm optimization algorithm". In: *Swarm and Evolutionary Computation* 41 (2018), pp. 20–35. DOI: https://doi.org/10.1016/j.swevo.2018.01.006.

[53] Andries P. Engelbrecht. "Particle Swarm Optimization: Iteration Strategies Revisited". In: *Proceedings of the BRICS Congress on Computational Intelligence & 11th Brazilian Congress on Computational Intelligence*. Sept. 2013, pp. 119–123. DOI: https://doi.org/10.1109/BRICS-CCI-CBIC.2013.30.

[54] Mardé Helbig and Andries P. Engelbrecht. "Analysing the performance of dynamic multi-objective optimisation algorithms". In: *Proceedings of the IEEE Congress on Evolutionary Computation*. IEEE, 2013, pp. 1531–1539. DOI: https://doi.org/10.1109/CEC.2013.6557744.

Appendix A

Additional Results

A.1 Archive Management Experiments Broken Down by DMOP Type

This appendix lists experimental results for Type I, Type II, and Type III DMOPs by the baseline MGPSO, MGPSO with local search archive management strategy, DMOES, DNSGA-II, and SGEA.

Table A.1 Overall wins and losses for type I DMOPs across all performance measures and n_t-τ_t combinations

Results	**DMOAs**				
	MGPSO	**MGPSO$_{ls}$**	**DMOES**	**DNSGA-II**	**SGEA**
Wins	197.11	310.18	150.39	95.6	186.17
Losses	147.26	61.53	254.8	281.97	193.89
Diff	49.85	248.65	-104.41	-186.37	-7.72
Rank	2	**1**	4	5	3

Table A.2 Overall wins and losses by various frequencies and severities of change for type I DMOPs across all performance measures

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**DMOAs**				
				MGPSO	**MGPSO$_{ls}$**	**DMOES**	**DNSGA-II**	**SGEA**
10	10	all	Wins	60.8	95.3	43.7	22.3	48.24
10	10	all	Losses	40.56	14.03	71.12	85.8	58.83
10	10	all	Diff	20.24	81.27	-27.42	-63.5	-10.59
10	10	all	Rank	2	**1**	4	5	3
10	25	all	Wins	20.3	35.72	18.59	10.5	21.13
10	25	all	Losses	18.49	7.44	26.83	32.39	21.09
10	25	all	Diff	1.81	28.28	-8.24	-21.89	0.04
10	25	all	Rank	2	**1**	4	5	3
10	50	all	Wins	10.11	15.51	9.56	5.3	11.43
10	50	all	Losses	8.95	4.68	12.62	15.78	9.88
10	50	all	Diff	1.16	10.83	-3.06	-10.48	1.55
10	50	all	Rank	3	**1**	4	5	2
1	10	all	Wins	53.72	68.82	29.49	35.74	60.77
1	10	all	Losses	38.96	21.33	81.26	65.12	41.87
1	10	all	Diff	14.76	47.49	-51.77	-29.38	18.9
1	10	all	Rank	3	**1**	5	4	2
20	10	all	Wins	52.18	94.83	49.05	21.76	44.6
20	10	all	Losses	40.3	14.05	62.97	82.88	62.22
20	10	all	Diff	11.88	80.78	-13.92	-61.12	-17.62
20	10	all	Rank	2	**1**	3	5	4

Table A.3 Overall wins and losses by various performance measures for type I DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**DMOAs**				
				MGPSO	**MGPSO$_{ls}$**	**DMOES**	**DNSGA-II**	**SGEA**
all	all	*S*	Wins	44.8	53.02	9.66	10.91	43.37
all	all	*S*	Losses	15.04	9.2	58.12	56.51	22.89
all	all	*S*	Diff	29.76	43.82	-48.46	-45.6	20.48
all	all	*S*	Rank	2	**1**	5	4	3
all	all	*VD*	Wins	38.4	64.75	24.9	15.98	39.94
all	all	*VD*	Losses	31.81	9.25	45.66	61.46	35.79
all	all	*VD*	Diff	6.59	55.5	-20.76	-45.48	4.15
all	all	*VD*	Rank	2	**1**	4	5	3
all	all	*MS*	Wins	34	56.76	25.17	24.35	14.86
all	all	*MS*	Losses	21.69	5.51	45.14	34.95	47.85
all	all	*MS*	Diff	12.31	51.25	-19.97	-10.6	-32.99
all	all	*MS*	Rank	2	**1**	4	3	5
all	all	*acc*	Wins	27.46	51.89	54.01	8.23	33.21
all	all	*acc*	Losses	35.83	16.48	23.08	61.12	38.29
all	all	*acc*	Diff	-8.37	35.41	30.93	-52.89	-5.08
all	all	*acc*	Rank	4	**1**	2	5	3
all	all	*stab*	Wins	24.48	28.89	28.14	18.35	11.91
all	all	*stab*	Losses	16.73	16.27	20.94	26.11	31.72
all	all	*stab*	Diff	7.75	12.62	7.2	-7.76	-19.81
all	all	*stab*	Rank	2	**1**	3	4	5
all	all	*NS*	Wins	27.97	54.87	8.51	17.78	42.88
all	all	*NS*	Losses	26.16	4.82	61.86	41.82	17.35
all	all	*NS*	Diff	1.81	50.05	-53.35	-24.04	25.53
all	all	*NS*	Rank	3	**1**	5	4	2

Table A.4 Overall wins and losses for type II DMOPs across all performance measures and n_t-τ_t combinations

Results	**DMOAs**				
	MGPSO	**MGPSO$_{ls}$**	**DMOES**	**DNSGA-II**	**SGEA**
Wins	561.4	948.71	439.77	264.1	508.05
Losses	459.81	156.82	716.95	807.76	580.69
Diff	101.59	791.89	-277.18	-543.66	-72.64
Rank	2	**1**	4	5	3

Table A.5 Overall wins and losses by various frequencies and severities of change for type II DMOPs across all performance measures

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**DMOAs**				
				MGPSO	**MGPSO$_{ls}$**	**DMOES**	**DNSGA-II**	**SGEA**
10	10	all	Wins	160.56	298.02	122.1	66.84	127.95
10	10	all	Losses	130.07	31.52	199.5	235.72	178.66
10	10	all	Diff	30.49	266.5	-77.4	-168.88	-50.71
10	10	all	Rank	2	**1**	4	5	3
10	25	all	Wins	55.67	104.33	60.6	26.89	56.88
10	25	all	Losses	57.14	17.83	69.17	94.93	65.3
10	25	all	Diff	-1.47	86.5	-8.57	-68.04	-8.42
10	25	all	Rank	2	**1**	4	5	3
10	50	all	Wins	25.1	45.43	30.36	15.1	30.32
10	50	all	Losses	29.54	11.93	33.34	42.6	28.9
10	50	all	Diff	-4.44	33.5	-2.98	-27.5	1.42
10	50	all	Rank	4	**1**	3	5	2
1	10	all	Wins	168.27	212.97	81.53	90.76	165.85
1	10	all	Losses	102.95	61.52	234.21	193.74	126.96
1	10	all	Diff	65.32	151.45	-152.68	-102.98	38.89
1	10	all	Rank	2	**1**	5	4	3
20	10	all	Wins	151.8	287.96	145.18	64.51	127.05
20	10	all	Losses	140.11	34.02	180.73	240.77	180.87
20	10	all	Diff	11.69	253.94	-35.55	-176.26	-53.82
20	10	all	Rank	2	**1**	3	5	4

Table A.6 Overall wins and losses by various performance measures for type II DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**DMOAs**				
				MGPSO	**MGPSO$_{ls}$**	**DMOES**	**DNSGA-II**	**SGEA**
all	all	*S*	Wins	110.05	175.99	48.73	31.8	111.58
all	all	*S*	Losses	72.53	25.6	136.34	163.07	80.61
all	all	*S*	Diff	37.52	150.39	-87.61	-131.27	30.97
all	all	*S*	Rank	2	**1**	4	5	3
all	all	*VD*	Wins	96.56	188.21	88.51	31.95	108.25
all	all	*VD*	Losses	100.39	24.87	113.15	179.03	96.04
all	all	*VD*	Diff	-3.83	163.34	-24.64	-147.08	12.21
all	all	*VD*	Rank	3	**1**	4	5	2
all	all	*MS*	Wins	122.43	155.74	50.61	72.73	48.08
all	all	*MS*	Losses	37.99	12.21	161.4	89.16	148.83
all	all	*MS*	Diff	84.44	143.53	-110.79	-16.43	-100.75
all	all	*MS*	Rank	2	**1**	5	3	4
all	all	*acc*	Wins	92.25	189.42	142.5	24.87	85.29
all	all	*acc*	Losses	113.78	32.49	73.79	191.47	122.8
all	all	*acc*	Diff	-21.53	156.93	68.71	-166.6	-37.51
all	all	*acc*	Rank	3	**1**	2	5	4
all	all	*stab*	Wins	61.83	100.36	85.14	52.7	55.52
all	all	*stab*	Losses	70.78	50.37	60.43	89.07	84.9
all	all	*stab*	Diff	-8.95	49.99	24.71	-36.37	-29.38
all	all	*stab*	Rank	3	**1**	2	5	4
all	all	*NS*	Wins	78.28	138.99	24.28	50.05	99.33
all	all	*NS*	Losses	64.34	11.28	171.84	95.96	47.51
all	all	*NS*	Diff	13.94	127.71	-147.56	-45.91	51.82
all	all	*NS*	Rank	3	**1**	5	4	2

Table A.7 Overall wins and losses for type III DMOPs across all performance measures and n_t-τ_t combinations

Results	**DMOAs**				
	MGPSO	**MGPSO$_{ls}$**	**DMOES**	**DNSGA-II**	**SGEA**
Wins	209.84	275.65	133.42	83.73	116.22
Losses	87.59	28.95	228.58	243.71	230.03
Diff	122.25	246.7	-95.16	-159.98	-113.81
Rank	2	**1**	3	5	4

Table A.8 Overall wins and losses by various frequencies and severities of change for type III DMOPs across all performance measures

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**DMOAs**				
				MGPSO	**MGPSO$_{ls}$**	**DMOES**	**DNSGA-II**	**SGEA**
10	10	all	Wins	59.3	77.96	38.97	21.19	31.85
10	10	all	Losses	23.84	6.68	61.98	70.81	65.96
10	10	all	Diff	35.46	71.28	-23.01	-49.62	-34.11
10	10	all	Rank	2	**1**	3	5	4
10	25	all	Wins	22.53	29.8	14.27	9.38	13.71
10	25	all	Losses	9.34	3.29	24.7	27.06	25.3
10	25	all	Diff	13.19	26.51	-10.43	-17.68	-11.59
10	25	all	Rank	2	**1**	3	5	4
10	50	all	Wins	11.1	14.37	7.72	4.81	7.13
10	50	all	Losses	5.06	2.02	12.23	13.43	12.39
10	50	all	Diff	6.04	12.35	-4.51	-8.62	-5.26
10	50	all	Rank	2	**1**	3	5	4
1	10	all	Wins	60.89	77.88	34.53	26.25	31.29
1	10	all	Losses	24.77	10.34	68.88	64.24	62.61
1	10	all	Diff	36.12	67.54	-34.35	-37.99	-31.32
1	10	all	Rank	2	**1**	4	5	3
20	10	all	Wins	56.02	75.64	37.93	22.1	32.24
20	10	all	Losses	24.58	6.62	60.79	68.17	63.77
20	10	all	Diff	31.44	69.02	-22.86	-46.07	-31.53
20	10	all	Rank	2	**1**	3	5	4

Table A.9 Overall wins and losses by various performance measures for type III DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**DMOAs**				
				MGPSO	**MGPSO$_{ls}$**	**DMOES**	**DNSGA-II**	**SGEA**
all	all	*S*	Wins	57.1	65.02	23.65	10.23	25.62
all	all	*S*	Losses	9.95	3.94	50.51	66.98	50.24
all	all	*S*	Diff	47.15	61.08	-26.86	-56.75	-24.62
all	all	*S*	Rank	2	**1**	4	5	3
all	all	*VD*	Wins	50.22	45.55	12.01	18.8	15.72
all	all	*VD*	Losses	2.63	8.2	47.01	37.87	46.59
all	all	*VD*	Diff	47.59	37.35	-35	-19.07	-30.87
all	all	*VD*	Rank	**1**	2	5	3	4
all	all	*MS*	Wins	27.92	51.42	14.3	29.89	26.64
all	all	*MS*	Losses	19.67	1.15	60.81	21.01	47.53
all	all	*MS*	Diff	8.25	50.27	-46.51	8.88	-20.89
all	all	*MS*	Rank	3	**1**	5	2	4
all	all	*acc*	Wins	42.67	69.16	50.3	9.16	25.31
all	all	*acc*	Losses	30.16	7.13	31.86	72.53	54.92
all	all	*acc*	Diff	12.51	62.03	18.44	-63.37	-29.61
all	all	*acc*	Rank	3	**1**	2	5	4
all	all	*stab*	Wins	26.39	31.69	31.13	5.24	13.21
all	all	*stab*	Losses	15.32	8.25	12.55	43.29	28.25
all	all	*stab*	Diff	11.07	23.44	18.58	-38.05	-15.04
all	all	*stab*	Rank	3	**1**	2	5	4
all	all	*NS*	Wins	5.54	12.81	2.03	10.41	9.72
all	all	*NS*	Losses	9.86	0.28	25.84	2.03	2.5
all	all	*NS*	Diff	-4.32	12.53	-23.81	8.38	7.22
all	all	*NS*	Rank	4	**1**	5	2	3

A.2 Balance Coefficient Experiments Broken Down by DMOP Type

A.2.1 Standard and Random Approaches

This appendix lists experimental results for Type I, Type II, and Type III DMOPs by the standard and random balance coefficient initialization strategies.

Table A.10 Overall wins and losses for type I DMOPs across all performance measures and n_t-τ_t combinations

Results	Balance Coefficient Strategies				
	std	**std**$_{\tau_t}$	**r**	**r**$_i$	**r**$_{ij}$
Wins	104.37	143.86	34.35	65.8	49.32
Losses	66.7	34.04	113.61	81.17	102.18
Diff	37.67	109.82	-79.26	-15.37	-52.86
Rank	2	**1**	5	3	4

Table A.11 Overall wins and losses by various frequencies and severities of change for type I DMOPs across all performance measures

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**Balance Coefficient Strategies**				
				std	**std**$_{\tau_t}$	**r**	**r**$_i$	**r**$_{ij}$
10	10	all	Wins	28.07	37.74	4.75	20.28	16.61
10	10	all	Losses	19.4	9.35	36.36	18.79	23.55
10	10	all	Diff	8.67	28.39	-31.61	1.49	-6.94
10	10	all	Rank	2	**1**	5	3	4
10	25	all	Wins	8.41	12.92	3.15	8.45	3.9
10	25	all	Losses	6.14	3.07	9.74	6.23	11.65
10	25	all	Diff	2.27	9.85	-6.59	2.22	-7.75
10	25	all	Rank	2	**1**	4	3	5
10	50	all	Wins	4.77	6.11	1.62	4.51	1.7
10	50	all	Losses	3.06	1.34	4.65	2.76	6.9
10	50	all	Diff	1.71	4.77	-3.03	1.75	-5.2
10	50	all	Rank	3	**1**	4	2	5
1	10	all	Wins	41.76	56.28	18.42	13.09	13.93
1	10	all	Losses	19.11	11.81	37.34	39.03	36.19
1	10	all	Diff	22.65	44.47	-18.92	-25.94	-22.26
1	10	all	Rank	2	**1**	3	5	4
20	10	all	Wins	21.36	30.81	6.41	19.47	13.18
20	10	all	Losses	18.99	8.47	25.52	14.36	23.89
20	10	all	Diff	2.37	22.34	-19.11	5.11	-10.71
20	10	all	Rank	3	**1**	5	2	4

Table A.12 Overall wins and losses by various performance measures for type I DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**Balance Coefficient Strategies**				
				std	$\mathbf{std}_{\tau_t}$	**r**	$\mathbf{r}_i$	$\mathbf{r}_{ij}$
all	all	*S*	Wins	11.09	16.58	4.74	8.15	4.47
all	all	*S*	Losses	7.62	3.44	11.07	8.67	14.23
all	all	*S*	Diff	3.47	13.14	-6.33	-0.52	-9.76
all	all	*S*	Rank	2	**1**	4	3	5
all	all	*VD*	Wins	33.8	47.84	6.38	21.8	14.45
all	all	*VD*	Losses	19.61	6.7	40.39	25.16	32.41
all	all	*VD*	Diff	14.19	41.14	-34.01	-3.36	-17.96
all	all	*VD*	Rank	2	**1**	5	3	4
all	all	*MS*	Wins	5.23	7.11	3.42	3.94	5.36
all	all	*MS*	Losses	5.33	4.79	6.76	4.23	3.95
all	all	*MS*	Diff	-0.1	2.32	-3.34	-0.29	1.41
all	all	*MS*	Rank	3	**1**	5	4	2
all	all	*acc*	Wins	28.04	38.64	6.04	15.81	12.27
all	all	*acc*	Losses	15.65	7.44	30.78	21.83	25.1
all	all	*acc*	Diff	12.39	31.2	-24.74	-6.02	-12.83
all	all	*acc*	Rank	2	**1**	5	3	4
all	all	*stab*	Wins	6.45	9.51	6.42	9.2	9.44
all	all	*stab*	Losses	11.94	8.18	9.75	5.38	5.77
all	all	*stab*	Diff	-5.49	1.33	-3.33	3.82	3.67
all	all	*stab*	Rank	5	3	4	**1**	2
all	all	*NS*	Wins	19.76	24.18	7.35	6.9	3.33
all	all	*NS*	Losses	6.55	3.49	14.86	15.9	20.72
all	all	*NS*	Diff	13.21	20.69	-7.51	-9	-17.39
all	all	*NS*	Rank	2	**1**	3	4	5

Table A.13 Overall wins and losses for type II DMOPs across all performance measures and n_t-τ_t combinations

Results	**Balance Coefficient Strategies**				
	std	$\mathbf{std}_{\tau_t}$	**r**	$\mathbf{r}_i$	$\mathbf{r}_{ij}$
Wins	445.14	472.17	121.96	197.55	100.76
Losses	124.57	95.32	352.52	277.11	488.06
Diff	320.57	376.85	-230.56	-79.56	-387.3
Rank	2	**1**	4	3	5

Table A.14 Overall wins and losses by various frequencies and severities of change for type II DMOPs across all performance measures

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**Balance Coefficient Strategies**				
				std	$\mathbf{std}_{\tau_t}$	**r**	$\mathbf{r}_i$	$\mathbf{r}_{ij}$
10	10	all	Wins	124.16	121.32	26.84	58.99	26.89
10	10	all	Losses	27.87	26.48	102.5	68.87	132.48
10	10	all	Diff	96.29	94.84	-75.66	-9.88	-105.59
10	10	all	Rank	**1**	2	4	3	5
10	25	all	Wins	46.67	50.29	15.25	26.39	10.88
10	25	all	Losses	12.88	9.54	37.2	28.5	61.36
10	25	all	Diff	33.79	40.75	-21.95	-2.11	-50.48
10	25	all	Rank	2	**1**	4	3	5
10	50	all	Wins	22.09	23.7	9.29	13.37	5.17
10	50	all	Losses	6.6	4.75	16.63	14.03	31.61
10	50	all	Diff	15.49	18.95	-7.34	-0.66	-26.44
10	50	all	Rank	2	**1**	4	3	5
1	10	all	Wins	138.83	153.02	36.8	36.74	31.89
1	10	all	Losses	43.28	29.97	102.67	98.29	123.07
1	10	all	Diff	95.55	123.05	-65.87	-61.55	-91.18
1	10	all	Rank	2	**1**	4	3	5
20	10	all	Wins	113.39	123.84	33.78	62.06	25.93
20	10	all	Losses	33.94	24.58	93.52	67.42	139.54
20	10	all	Diff	79.45	99.26	-59.74	-5.36	-113.61
20	10	all	Rank	2	**1**	4	3	5

Table A.15 Overall wins and losses by various performance measures for type II DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**Balance Coefficient Strategies**				
				std	**std**$_{\tau_t}$	**r**	**r**$_i$	**r**$_{ij}$
all	all	*S*	Wins	57.45	61.87	19.13	24.07	7.64
all	all	*S*	Losses	14.71	9.53	35.87	34.13	75.92
all	all	*S*	Diff	42.74	52.34	-16.74	-10.06	-68.28
all	all	*S*	Rank	2	**1**	4	3	5
all	all	*VD*	Wins	115.2	124.18	24.29	47.46	27.63
all	all	*VD*	Losses	26.02	17.17	99.68	77.82	118.07
all	all	*VD*	Diff	89.18	107.01	-75.39	-30.36	-90.44
all	all	*VD*	Rank	2	**1**	4	3	5
all	all	*MS*	Wins	29.92	32.87	10.15	20.94	11.01
all	all	*MS*	Losses	13.68	8.44	31.02	14.46	37.29
all	all	*MS*	Diff	16.24	24.43	-20.87	6.48	-26.28
all	all	*MS*	Rank	2	**1**	4	3	5
all	all	*acc*	Wins	117.04	129.53	26.55	51	20.18
all	all	*acc*	Losses	26.46	18.01	95.36	74.54	129.93
all	all	*acc*	Diff	90.58	111.52	-68.81	-23.54	-109.75
all	all	*acc*	Rank	2	**1**	4	3	5
all	all	*stab*	Wins	35.49	37.06	20.92	26.96	29.8
all	all	*stab*	Losses	33.62	30.14	30.45	24.26	31.76
all	all	*stab*	Diff	1.87	6.92	-9.53	2.7	-1.96
all	all	*stab*	Rank	3	**1**	5	2	4
all	all	*NS*	Wins	90.04	86.66	20.92	27.12	4.5
all	all	*NS*	Losses	10.08	12.03	60.14	51.9	95.09
all	all	*NS*	Diff	79.96	74.63	-39.22	-24.78	-90.59
all	all	*NS*	Rank	**1**	2	4	3	5

Table A.16 Overall wins and losses for type III DMOPs across all performance measures and n_t-τ_t combinations

Results	**Balance Coefficient Strategies**				
	std	**std**$_{\tau_t}$	**r**	**r**$_i$	**r**$_{ij}$
Wins	49	54.02	29.24	38.32	34.72
Losses	48.44	32.18	37.73	30.03	56.92
Diff	0.56	21.84	-8.49	8.29	-22.2
Rank	3	**1**	4	2	5

Table A.17 Overall wins and losses by various frequencies and severities of change for type III DMOPs across all performance measures

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**Balance Coefficient Strategies**				
				std	$\mathbf{std}_{\tau_t}$	**r**	$\mathbf{r}_i$	$\mathbf{r}_{ij}$
10	10	all	Wins	12.28	15.94	7.63	9.68	9.06
10	10	all	Losses	13.54	6.83	10.15	8.55	15.52
10	10	all	Diff	-1.26	9.11	-2.52	1.13	-6.46
10	10	all	Rank	3	**1**	4	2	5
10	25	all	Wins	5.44	6.04	4.25	5.15	4.75
10	25	all	Losses	6.72	5.18	4.2	3.74	5.79
10	25	all	Diff	-1.28	0.86	0.05	1.41	-1.04
10	25	all	Rank	5	2	3	**1**	4
10	50	all	Wins	2.63	2.68	2.78	3.02	2.57
10	50	all	Losses	4.04	3.71	1.75	1.75	2.43
10	50	all	Diff	-1.41	-1.03	1.03	1.27	0.14
10	50	all	Rank	5	4	2	**1**	3
1	10	all	Wins	13.6	13.41	5.95	8.78	7.02
1	10	all	Losses	8.94	5.01	10.55	6.87	17.39
1	10	all	Diff	4.66	8.4	-4.6	1.91	-10.37
1	10	all	Rank	2	**1**	4	3	5
20	10	all	Wins	15.05	15.95	8.63	11.69	11.32
20	10	all	Losses	15.2	11.45	11.08	9.12	15.79
20	10	all	Diff	-0.15	4.5	-2.45	2.57	-4.47
20	10	all	Rank	3	**1**	4	2	5

Table A.18 Overall wins and losses by various performance measures for type III DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**Balance Coefficient Strategies**				
				std	$\mathbf{std}_{\tau_t}$	**r**	$\mathbf{r}_i$	$\mathbf{r}_{ij}$
all	all	*S*	Wins	0.19	1.37	9.37	8.48	11.37
all	all	*S*	Losses	17.53	11.45	0.28	1	0.52
all	all	*S*	Diff	-17.34	-10.08	9.09	7.48	10.85
all	all	*S*	Rank	5	4	2	3	**1**
all	all	*VD*	Wins	2.18	4.09	6.34	4.55	10.05
all	all	*VD*	Losses	10.09	7.24	3.8	1.37	4.71
all	all	*VD*	Diff	-7.91	-3.15	2.54	3.18	5.34
all	all	*VD*	Rank	5	4	3	2	**1**
all	all	*MS*	Wins	11.16	10.32	1.32	5.12	1.57
all	all	*MS*	Losses	3.38	1.13	7.65	6.32	11.01
all	all	*MS*	Diff	7.78	9.19	-6.33	-1.2	-9.44
all	all	*MS*	Rank	2	**1**	4	3	5
all	all	*acc*	Wins	16.42	19.11	7.13	11.9	6.85
all	all	*acc*	Losses	12.28	7.76	12.96	8.24	20.17
all	all	*acc*	Diff	4.14	11.35	-5.83	3.66	-13.32
all	all	*acc*	Rank	2	**1**	4	3	5
all	all	*stab*	Wins	8.27	8.44	3.37	4.85	4.87
all	all	*stab*	Losses	5.14	4.49	5.85	6	8.32
all	all	*stab*	Diff	3.13	3.95	-2.48	-1.15	-3.45
all	all	*stab*	Rank	2	**1**	4	3	5
all	all	*NS*	Wins	10.78	10.69	1.71	3.42	0.01
all	all	*NS*	Losses	0.02	0.11	7.19	7.1	12.19
all	all	*NS*	Diff	10.76	10.58	-5.48	-3.68	-12.18
all	all	*NS*	Rank	**1**	2	4	3	5

A.2.2 Standard and Linearly Increasing/Decreasing Results

This appendix lists experimental results for Type I, Type II, and Type III DMOPs by the std_{τ_t}, linearly increasing and linearly decreasing coefficient initialization strategies.

Table A.19 Overall wins and losses for type I DMOPs across all performance measures and n_t-τ_t combinations

Results	Balance Coefficient Strategies				
	$\mathbf{std}_{\tau_t}$	**ld**	$\mathbf{ld}_{\tau_t}$	**li**	$\mathbf{li}_{\tau_t}$
Wins	214.87	130.77	121.32	122.41	196.95
Losses	80.55	237.53	175.6	201.24	91.4
Diff	134.32	-106.76	-54.28	-78.83	105.55
Rank	**1**	5	3	4	2

Table A.20 Overall wins and losses by various frequencies and severities of change for type I DMOPs across all performance measures

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	Balance Coefficient Strategies				
				$\mathbf{std}_{\tau_t}$	**ld**	$\mathbf{ld}_{\tau_t}$	**li**	$\mathbf{li}_{\tau_t}$
10	10	all	Wins	59.33	36.63	33.33	32.24	61.62
10	10	all	Losses	19.61	71.28	52.84	59.84	19.58
10	10	all	Diff	39.72	-34.65	-19.51	-27.6	42.04
10	10	all	Rank	2	5	3	4	**1**
10	25	all	Wins	23.57	12.29	15.57	13.19	25.09
10	25	all	Losses	8.62	28.75	19.69	24.86	7.79
10	25	all	Diff	14.95	-16.46	-4.12	-11.67	17.3
10	25	all	Rank	2	5	3	4	**1**
10	50	all	Wins	13.26	5.16	7.75	6.91	12.35
10	50	all	Losses	3.95	16.26	9.79	11.36	4.07
10	50	all	Diff	9.31	-11.1	-2.04	-4.45	8.28
10	50	all	Rank	**1**	5	3	4	2
1	10	all	Wins	63.34	46.25	26.77	36.71	38.12
1	10	all	Losses	28.11	43.7	52.26	43.87	43.25
1	10	all	Diff	35.23	2.55	-25.49	-7.16	-5.13
1	10	all	Rank	**1**	2	5	4	3
20	10	all	Wins	55.37	30.44	37.9	33.36	59.77
20	10	all	Losses	20.26	77.54	41.02	61.31	16.71
20	10	all	Diff	35.11	-47.1	-3.12	-27.95	43.06
20	10	all	Rank	2	5	3	4	**1**

Table A.21 Overall wins and losses by various performance measures for type I DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**Balance Coefficient Strategies**				
				std$_{\tau_t}$	**ld**	**ld**$_{\tau_t}$	**li**	**li**$_{\tau_t}$
all	all	*S*	Wins	28.58	23.39	17.86	24.98	35.11
all	all	*S*	Losses	18.02	39.1	29.93	29.23	13.64
all	all	*S*	Diff	10.56	-15.71	-12.07	-4.25	21.47
all	all	*S*	Rank	2	5	4	3	**1**
all	all	*VD*	Wins	62.11	17.8	27.79	20.55	52.1
all	all	*VD*	Losses	9.17	58.89	42.81	53.54	15.94
all	all	*VD*	Diff	52.94	-41.09	-15.02	-32.99	36.16
all	all	*VD*	Rank	**1**	5	3	4	2
all	all	*MS*	Wins	27.14	16.95	20.97	14.62	21.27
all	all	*MS*	Losses	7.05	40.45	12.91	28.07	12.47
all	all	*MS*	Diff	20.09	-23.5	8.06	-13.45	8.8
all	all	*MS*	Rank	**1**	5	3	4	2
all	all	*acc*	Wins	50.03	8.71	23.87	17.45	41.74
all	all	*acc*	Losses	8.91	49.99	32.39	38.36	12.15
all	all	*acc*	Diff	41.12	-41.28	-8.52	-20.91	29.59
all	all	*acc*	Rank	**1**	5	3	4	2
all	all	*stab*	Wins	14.03	30.89	13.4	23.12	15.33
all	all	*stab*	Losses	21.82	14.53	23.3	17.81	19.31
all	all	*stab*	Diff	-7.79	16.36	-9.9	5.31	-3.98
all	all	*stab*	Rank	4	**1**	5	2	3
all	all	*NS*	Wins	32.98	33.03	17.43	21.69	31.4
all	all	*NS*	Losses	15.58	34.57	34.26	34.23	17.89
all	all	*NS*	Diff	17.4	-1.54	-16.83	-12.54	13.51
all	all	*NS*	Rank	**1**	3	5	4	2

Table A.22 Overall wins and losses for type II DMOPs across all performance measures and n_t-τ_t combinations

Results	**Balance Coefficient Strategies**				
	$\mathbf{std}_{\tau_t}$	**ld**	$\mathbf{ld}_{\tau_t}$	**li**	$\mathbf{li}_{\tau_t}$
Wins	601.17	298.31	403.16	359.89	462.06
Losses	196.14	720.95	381.35	530.57	295.58
Diff	405.03	-422.64	21.81	-170.68	166.48
Rank	**1**	5	3	4	2

Table A.23 Overall wins and losses by various frequencies and severities of change for type II DMOPs across all performance measures

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**Balance Coefficient Strategies**				
				$\mathbf{std}_{\tau_t}$	**ld**	$\mathbf{ld}_{\tau_t}$	**li**	$\mathbf{li}_{\tau_t}$
10	10	all	Wins	163.79	85.07	109.43	101.03	129.17
10	10	all	Losses	49.11	199.64	106.87	152.92	79.95
10	10	all	Diff	114.68	-114.57	2.56	-51.89	49.22
10	10	all	Rank	**1**	5	3	4	2
10	25	all	Wins	67.57	25.78	49.6	40.05	62.3
10	25	all	Losses	18.95	93.76	44.34	62.16	26.09
10	25	all	Diff	48.62	-67.98	5.26	-22.11	36.21
10	25	all	Rank	**1**	5	3	4	2
10	50	all	Wins	33.37	10.99	23.76	22.58	30.7
10	50	all	Losses	9.63	48.08	22.8	27.59	13.3
10	50	all	Diff	23.74	-37.09	0.96	-5.01	17.4
10	50	all	Rank	**1**	5	3	4	2
1	10	all	Wins	172.15	100.78	98.87	98.84	108.53
1	10	all	Losses	70.19	162.4	113.86	134.02	98.7
1	10	all	Diff	101.96	-61.62	-14.99	-35.18	9.83
1	10	all	Rank	**1**	5	3	4	2
20	10	all	Wins	164.29	75.69	121.5	97.39	131.36
20	10	all	Losses	48.26	217.07	93.48	153.88	77.54
20	10	all	Diff	116.03	-141.38	28.02	-56.49	53.82
20	10	all	Rank	**1**	5	3	4	2

Table A.24 Overall wins and losses by various performance measures for type II DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**Balance Coefficient Strategies**				
				$\mathbf{std}_{\tau_t}$	**ld**	$\mathbf{ld}_{\tau_t}$	**li**	$\mathbf{li}_{\tau_t}$
all	all	*S*	Wins	74.46	53.48	70.04	63.65	71.1
all	all	*S*	Losses	44.67	111.02	46.76	84.05	46.23
all	all	*S*	Diff	29.79	-57.54	23.28	-20.4	24.87
all	all	*S*	Rank	**1**	5	3	4	2
all	all	*VD*	Wins	144.38	42.31	89.97	67.59	114.43
all	all	*VD*	Losses	26.75	172.68	79.71	123.7	55.84
all	all	*VD*	Diff	117.63	-130.37	10.26	-56.11	58.59
all	all	*VD*	Rank	**1**	5	3	4	2
all	all	*MS*	Wins	61.58	47.49	39.48	47.23	44.64
all	all	*MS*	Losses	20.81	86.68	51.79	48.09	33.05
all	all	*MS*	Diff	40.77	-39.19	-12.31	-0.86	11.59
all	all	*MS*	Rank	**1**	5	4	3	2
all	all	*acc*	Wins	164.28	25.92	95.25	58.27	110.75
all	all	*acc*	Losses	19.65	174.12	74.23	127.28	59.19
all	all	*acc*	Diff	144.63	-148.2	21.02	-69.01	51.56
all	all	*acc*	Rank	**1**	5	3	4	2
all	all	*stab*	Wins	58.08	71.42	53.19	64.32	53.44
all	all	*stab*	Losses	56.55	74.49	53.63	64.79	50.99
all	all	*stab*	Diff	1.53	-3.07	-0.44	-0.47	2.45
all	all	*stab*	Rank	2	5	3	4	**1**
all	all	*NS*	Wins	98.39	57.69	55.23	58.83	67.7
all	all	*NS*	Losses	27.71	101.96	75.23	82.66	50.28
all	all	*NS*	Diff	70.68	-44.27	-20	-23.83	17.42
all	all	*NS*	Rank	**1**	5	3	4	2

Table A.25 Overall wins and losses for type III DMOPs across all performance measures and n_t-τ_t combinations

Results	**Balance Coefficient Strategies**				
	$\mathbf{std}_{\tau_t}$	**ld**	$\mathbf{ld}_{\tau_t}$	**li**	$\mathbf{li}_{\tau_t}$
Wins	106.94	74.77	109.59	92.6	91.04
Losses	56.8	183.67	66.08	100.24	68.15
Diff	50.14	-108.9	43.51	-7.64	22.89
Rank	**1**	5	2	4	3

Table A.26 Overall wins and losses by various frequencies and severities of change for type III DMOPs across all performance measures

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**Balance Coefficient Strategies**				
				$\mathbf{std}_{\tau_t}$	**ld**	$\mathbf{ld}_{\tau_t}$	**li**	$\mathbf{li}_{\tau_t}$
10	10	all	Wins	27.95	15.19	28.82	24.01	22.4
10	10	all	Losses	13.63	48.33	14.15	25	17.26
10	10	all	Diff	14.32	-33.14	14.67	-0.99	5.14
10	10	all	Rank	2	5	**1**	4	3
10	25	all	Wins	12.86	10.66	14.39	11.62	12.27
10	25	all	Losses	8.02	22.01	10.1	12.04	9.63
10	25	all	Diff	4.84	-11.35	4.29	-0.42	2.64
10	25	all	Rank	**1**	5	2	4	3
10	50	all	Wins	6.13	6.69	6.96	6.37	6.63
10	50	all	Losses	5.48	10	5.97	5.75	5.58
10	50	all	Diff	0.65	-3.31	0.99	0.62	1.05
10	50	all	Rank	3	5	2	4	**1**
1	10	all	Wins	30.45	17.99	28.43	25.55	24.71
1	10	all	Losses	12.15	52.86	17.38	29.65	15.09
1	10	all	Diff	18.3	-34.87	11.05	-4.1	9.62
1	10	all	Rank	**1**	5	2	4	3
20	10	all	Wins	29.55	24.24	30.99	25.05	25.03
20	10	all	Losses	17.52	50.47	18.48	27.8	20.59
20	10	all	Diff	12.03	-26.23	12.51	-2.75	4.44
20	10	all	Rank	2	5	**1**	4	3

Table A.27 Overall wins and losses by various performance measures for type III DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**Balance Coefficient Strategies**				
				std$_{\tau_t}$	**ld**	**ld**$_{\tau_t}$	**li**	**li**$_{\tau_t}$
all	all	*S*	Wins	13	13.3	21.07	20.9	25.52
all	all	*S*	Losses	18.89	37.84	13.29	18.27	5.5
all	all	*S*	Diff	-5.89	-24.54	7.78	2.63	20.02
all	all	*S*	Rank	4	5	2	3	**1**
all	all	*VD*	Wins	9.34	16.98	12.46	25.12	19.43
all	all	*VD*	Losses	16.82	30.93	14.59	10.71	10.28
all	all	*VD*	Diff	-7.48	-13.95	-2.13	14.41	9.15
all	all	*VD*	Rank	4	5	3	**1**	2
all	all	*MS*	Wins	23.91	11.68	22.84	11.69	12.11
all	all	*MS*	Losses	4.18	36.26	7.82	20.48	13.49
all	all	*MS*	Diff	19.73	-24.58	15.02	-8.79	-1.38
all	all	*MS*	Rank	**1**	5	2	4	3
all	all	*acc*	Wins	29.59	10.89	27.78	15.85	18.93
all	all	*acc*	Losses	6.35	40.46	12.04	27.32	16.87
all	all	*acc*	Diff	23.24	-29.57	15.74	-11.47	2.06
all	all	*acc*	Rank	**1**	5	2	4	3
all	all	*stab*	Wins	15.51	14.05	13.71	13.73	10.83
all	all	*stab*	Losses	9.59	21.78	11.78	12.87	11.81
all	all	*stab*	Diff	5.92	-7.73	1.93	0.86	-0.98
all	all	*stab*	Rank	**1**	5	2	3	4
all	all	*NS*	Wins	15.59	7.87	11.73	5.31	4.22
all	all	*NS*	Losses	0.97	16.4	6.56	10.59	10.2
all	all	*NS*	Diff	14.62	-8.53	5.17	-5.28	-5.98
all	all	*NS*	Rank	**1**	5	2	3	4

A.2.3 Comparisons with the other DMOAs

This appendix lists experimental results for Type I, Type II, and Type III DMOPs by the MGPSO with std_{τ_t} balance coefficient update strategy, DMOES, DNSGA-II, and SGEA.

Table A.28 Overall wins and losses for type I DMOPs across all performance measures and n_t-τ_t combinations

Results	**DMOAs**			
	MGPSO	**DMOES**	**DNSGA-II**	**SGEA**
Wins	192.71	125.69	86.34	163
Losses	89.03	169.8	191.31	117.6
Diff	103.68	-44.11	-104.97	45.4
Rank	**1**	3	4	2

Table A.29 Overall wins and losses by various frequencies and severities of change for type I DMOPs across all performance measures

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**DMOAs**			
				MGPSO	**DMOES**	**DNSGA-II**	**SGEA**
10	10	all	Wins	58.89	37.4	20.78	43.62
10	10	all	Losses	21.13	45.9	58.12	35.54
10	10	all	Diff	37.76	-8.5	-37.34	8.08
10	10	all	Rank	**1**	3	4	2
10	25	all	Wins	19.94	16.56	10.17	18.03
10	25	all	Losses	12.32	17.62	21.6	13.16
10	25	all	Diff	7.62	-1.06	-11.43	4.87
10	25	all	Rank	**1**	3	4	2
10	50	all	Wins	9.9	7.71	4.81	9.65
10	50	all	Losses	6.19	9.15	10.54	6.19
10	50	all	Diff	3.71	-1.44	-5.73	3.46
10	50	all	Rank	**1**	3	4	2
1	10	all	Wins	52.74	22.85	30.25	51.87
1	10	all	Losses	28.85	56.9	46.5	25.46
1	10	all	Diff	23.89	-34.05	-16.25	26.41
1	10	all	Rank	2	4	3	**1**
20	10	all	Wins	51.24	41.17	20.33	39.83
20	10	all	Losses	20.54	40.23	54.55	37.25
20	10	all	Diff	30.7	0.94	-34.22	2.58
20	10	all	Rank	**1**	3	4	2

Table A.30 Overall wins and losses by various performance measures for type I DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**DMOAs**			
				MGPSO	**DMOES**	**DNSGA-II**	**SGEA**
all	all	*S*	Wins	43.42	8.21	10.41	38.1
all	all	*S*	Losses	8.23	41.33	38.71	11.87
all	all	*S*	Diff	35.19	-33.12	-28.3	26.23
all	all	*S*	Rank	**1**	4	3	2
all	all	*VD*	Wins	38.19	22.98	14.58	34.85
all	all	*VD*	Losses	16.61	29.38	42.74	21.87
all	all	*VD*	Diff	21.58	-6.4	-28.16	12.98
all	all	*VD*	Rank	**1**	3	4	2
all	all	*MS*	Wins	34.19	22.9	22.25	13.46
all	all	*MS*	Losses	14.27	27.96	19.56	31.01
all	all	*MS*	Diff	19.92	-5.06	2.69	-17.55
all	all	*MS*	Rank	**1**	3	2	4
all	all	*acc*	Wins	27.38	42.87	7.86	28.27
all	all	*acc*	Losses	23.58	14.1	44.34	24.36
all	all	*acc*	Diff	3.8	28.77	-36.48	3.91
all	all	*acc*	Rank	3	**1**	4	2
all	all	*stab*	Wins	21.33	21.18	14.68	8.39
all	all	*stab*	Losses	12.37	13.5	18.24	21.47
all	all	*stab*	Diff	8.96	7.68	-3.56	-13.08
all	all	*stab*	Rank	**1**	2	3	4
all	all	*NS*	Wins	28.2	7.55	16.56	39.93
all	all	*NS*	Losses	13.97	43.53	27.72	7.02
all	all	*NS*	Diff	14.23	-35.98	-11.16	32.91
all	all	*NS*	Rank	2	4	3	**1**

Table A.31 Overall wins and losses for type II DMOPs across all performance measures and n_t-τ_t combinations

Results	DMOAs			
	MGPSO	**DMOES**	**DNSGA-II**	**SGEA**
Wins	544.01	384.69	238.63	453.98
Losses	256	470.23	552.36	342.72
Diff	288.01	-85.54	-313.73	111.26
Rank	**1**	3	4	2

Table A.32 Overall wins and losses by various frequencies and severities of change for type II DMOPs across all performance measures

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**DMOAs**			
				MGPSO	**DMOES**	**DNSGA-II**	**SGEA**
10	10	all	Wins	156.19	110.98	62.31	119.61
10	10	all	Losses	64.17	124.1	158.76	102.06
10	10	all	Diff	92.02	-13.12	-96.45	17.55
10	10	all	Rank	**1**	3	4	2
10	25	all	Wins	53.96	52.15	25.24	51.5
10	25	all	Losses	34.06	44.94	65.3	38.55
10	25	all	Diff	19.9	7.21	-40.06	12.95
10	25	all	Rank	**1**	3	4	2
10	50	all	Wins	24.03	25.5	13.32	26.61
10	50	all	Losses	19.67	22.03	29.76	18
10	50	all	Diff	4.36	3.47	-16.44	8.61
10	50	all	Rank	2	3	4	**1**
1	10	all	Wins	159.7	69.83	77.63	138.81
1	10	all	Losses	66.18	165.68	134.5	79.61
1	10	all	Diff	93.52	-95.85	-56.87	59.2
1	10	all	Rank	**1**	4	3	2
20	10	all	Wins	150.13	126.23	60.13	117.45
20	10	all	Losses	71.92	113.48	164.04	104.5
20	10	all	Diff	78.21	12.75	-103.91	12.95
20	10	all	Rank	**1**	3	4	2

Table A.33 Overall wins and losses by various performance measures for type II DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**DMOAs**			
				MGPSO	**DMOES**	**DNSGA-II**	**SGEA**
all	all	*S*	Wins	107.33	42.29	30.61	98.12
all	all	*S*	Losses	34.79	92.84	110.47	40.25
all	all	*S*	Diff	72.54	-50.55	-79.86	57.87
all	all	*S*	Rank	**1**	3	4	2
all	all	*VD*	Wins	93.22	77.43	29.61	99.95
all	all	*VD*	Losses	55.76	72.27	123.33	48.85
all	all	*VD*	Diff	37.46	5.16	-93.72	51.1
all	all	*VD*	Rank	2	3	4	**1**
all	all	*MS*	Wins	121.32	47.25	69.65	45.53
all	all	*MS*	Losses	20.24	109.27	55.61	98.63
all	all	*MS*	Diff	101.08	-62.02	14.04	-53.1
all	all	*MS*	Rank	**1**	4	2	3
all	all	*acc*	Wins	92.03	125.53	22.21	75.42
all	all	*acc*	Losses	64.14	35.66	137.34	78.05
all	all	*acc*	Diff	27.89	89.87	-115.13	-2.63
all	all	*acc*	Rank	2	**1**	4	3
all	all	*stab*	Wins	51.83	69.14	39.16	43.25
all	all	*stab*	Losses	47.37	38.6	61.57	55.84
all	all	*stab*	Diff	4.46	30.54	-22.41	-12.59
all	all	*stab*	Rank	2	**1**	4	3
all	all	*NS*	Wins	78.28	23.05	47.39	91.71
all	all	*NS*	Losses	33.7	121.59	64.04	21.1
all	all	*NS*	Diff	44.58	-98.54	-16.65	70.61
all	all	*NS*	Rank	2	4	3	**1**

Table A.34 Overall wins and losses for type III DMOPs across all performance measures and n_t-τ_t combinations

Results	**DMOAs**			
	MGPSO	**DMOES**	**DNSGA-II**	**SGEA**
Wins	198.72	124.55	76.31	108.08
Losses	49.36	144.54	169.5	144.26
Diff	149.36	-19.99	-93.19	-36.18
Rank	**1**	2	4	3

Table A.35 Overall wins and losses by various frequencies and severities of change for type III DMOPs across all performance measures

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**DMOAs**			
				MGPSO	**DMOES**	**DNSGA-II**	**SGEA**
10	10	all	Wins	55.78	35.98	20.34	30.21
10	10	all	Losses	12.73	39.42	49.09	41.07
10	10	all	Diff	43.05	-3.44	-28.75	-10.86
10	10	all	Rank	**1**	2	4	3
10	25	all	Wins	21.64	14.13	8.16	12.34
10	25	all	Losses	5.86	15.08	19.37	15.96
10	25	all	Diff	15.78	-0.95	-11.21	-3.62
10	25	all	Rank	**1**	2	4	3
10	50	all	Wins	10.61	6.85	4.33	6.48
10	50	all	Losses	3.12	7.62	9.57	7.96
10	50	all	Diff	7.49	-0.77	-5.24	-1.48
10	50	all	Rank	**1**	2	4	3
1	10	all	Wins	57.03	31.61	24.18	29.2
1	10	all	Losses	14.08	44.8	43.87	39.27
1	10	all	Diff	42.95	-13.19	-19.69	-10.07
1	10	all	Rank	**1**	3	4	2
20	10	all	Wins	53.66	35.98	19.3	29.85
20	10	all	Losses	13.57	37.62	47.6	40
20	10	all	Diff	40.09	-1.64	-28.3	-10.15
20	10	all	Rank	**1**	2	4	3

Table A.36 Overall wins and losses by various performance measures for type III DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**DMOAs**			
				MGPSO	**DMOES**	**DNSGA-II**	**SGEA**
all	all	*S*	Wins	54.6	24.47	8.23	24.79
all	all	*S*	Losses	2.2	30.26	49.14	30.49
all	all	*S*	Diff	52.4	-5.79	-40.91	-5.7
all	all	*S*	Rank	**1**	3	4	2
all	all	*VD*	Wins	45.4	12.1	17.24	13.74
all	all	*VD*	Losses	0.97	32.96	23.98	30.57
all	all	*VD*	Diff	44.43	-20.86	-6.74	-16.83
all	all	*VD*	Rank	**1**	4	2	3
all	all	*MS*	Wins	27.81	13.4	28.87	24.11
all	all	*MS*	Losses	12.13	41.33	12.43	28.3
all	all	*MS*	Diff	15.68	-27.93	16.44	-4.19
all	all	*MS*	Rank	2	4	**1**	3
all	all	*acc*	Wins	41.93	46.84	7.83	23.96
all	all	*acc*	Losses	17.74	14.41	52.87	35.54
all	all	*acc*	Diff	24.19	32.43	-45.04	-11.58
all	all	*acc*	Rank	2	**1**	4	3
all	all	*stab*	Wins	23.47	26.21	3.86	11.75
all	all	*stab*	Losses	10.32	6.9	30.07	18
all	all	*stab*	Diff	13.15	19.31	-26.21	-6.25
all	all	*stab*	Rank	2	**1**	4	3
all	all	*NS*	Wins	5.51	1.53	10.28	9.73
all	all	*NS*	Losses	6	18.68	1.01	1.36
all	all	*NS*	Diff	-0.49	-17.15	9.27	8.37
all	all	*NS*	Rank	3	4	**1**	2

A.3 QPSO Experiments Broken Down by DMOP Type

A.3.1 Results for MGPSO with 50% Proportion of Quantum Particles

This appendix lists experimental results for Type I, Type II, and Type III DMOPs by the MGPSO with self-adaptive quantum particles and PCX quantum particles at 50% quantum proportion.

Table A.37 Overall wins and losses for type I DMOPs across all performance measures and n_t-τ_t combinations

Results	**QPSO Strategies**					
	QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
Wins	83.55	232.33	253.99	25.68	56.93	55.58
Losses	154.81	44.37	34.24	195.37	140.12	139.15
Diff	-71.26	187.96	219.75	-169.69	-83.19	-83.57
Rank	3	2	**1**	6	4	5

Table A.38 Overall wins and losses by various frequencies and severities of change for type I DMOPs across all performance measures

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**QPSO Strategies**					
				QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
10	10	all	Wins	19.6	72.35	82.31	4.13	11.79	11.41
10	10	all	Losses	44.6	8.91	4.14	57.39	44.07	42.48
10	10	all	Diff	-25	63.44	78.17	-53.26	-32.28	-31.07
10	10	all	Rank	3	2	**1**	6	5	4
10	25	all	Wins	6.67	28.97	30.8	1.63	3.77	4.22
10	25	all	Losses	16.85	3.22	1.61	21.74	16.17	16.47
10	25	all	Diff	-10.18	25.75	29.19	-20.11	-12.4	-12.25
10	25	all	Rank	3	2	**1**	6	5	4
10	50	all	Wins	4.64	12.67	13.54	1.57	2.85	2.54
10	50	all	Losses	8.81	2.07	1.31	9.82	8.61	7.19
10	50	all	Diff	-4.17	10.6	12.23	-8.25	-5.76	-4.65
10	50	all	Rank	3	2	**1**	6	5	4
1	10	all	Wins	29.3	46.5	49.02	14.01	27.72	27.21
1	10	all	Losses	40.93	21.55	22.6	52.43	28.33	27.92
1	10	all	Diff	-11.63	24.95	26.42	-38.42	-0.61	-0.71
1	10	all	Rank	5	2	**1**	6	3	4
20	10	all	Wins	23.34	71.84	78.32	4.34	10.8	10.2
20	10	all	Losses	43.62	8.62	4.58	53.99	42.94	45.09
20	10	all	Diff	-20.28	63.22	73.74	-49.65	-32.14	-34.89
20	10	all	Rank	3	2	**1**	6	4	5

Table A.39 Overall wins and losses by various performance measures for type I DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**QPSO Strategies**					
				QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
all	all	*S*	Wins	4.2	49.88	60.04	3.06	7.85	8.05
all	all	*S*	Losses	37.95	5.64	1.37	35.23	25.58	27.31
all	all	*S*	Diff	-33.75	44.24	58.67	-32.17	-17.73	-19.26
all	all	*S*	Rank	6	2	**1**	5	3	4
all	all	*VD*	Wins	20.75	57.29	59.54	5.66	15.67	14.44
all	all	*VD*	Losses	37.33	9.79	10.03	48.74	33.55	33.91
all	all	*VD*	Diff	-16.58	47.5	49.51	-43.08	-17.88	-19.47
all	all	*VD*	Rank	3	2	**1**	6	4	5
all	all	*MS*	Wins	8.9	6.58	10.86	4.08	3.63	4.17
all	all	*MS*	Losses	4.89	6.98	2.83	8.15	8.46	6.91
all	all	*MS*	Diff	4.01	-0.4	8.03	-4.07	-4.83	-2.74
all	all	*MS*	Rank	2	3	**1**	5	6	4
all	all	*acc*	Wins	26.44	43.57	47.28	4.95	15.03	14.74
all	all	*acc*	Losses	26.84	13.64	11.87	45.44	27.49	26.73
all	all	*acc*	Diff	-0.4	29.93	35.41	-40.49	-12.46	-11.99
all	all	*acc*	Rank	3	2	**1**	6	5	4
all	all	*stab*	Wins	6.5	13.42	14.15	5.84	7.37	6.9
all	all	*stab*	Losses	15.86	5.25	5.2	11.18	8.35	8.34
all	all	*stab*	Diff	-9.36	8.17	8.95	-5.34	-0.98	-1.44
all	all	*stab*	Rank	6	2	**1**	5	3	4
all	all	*NS*	Wins	16.76	61.59	62.12	2.09	7.38	7.28
all	all	*NS*	Losses	31.94	3.07	2.94	46.63	36.69	35.95
all	all	*NS*	Diff	-15.18	58.52	59.18	-44.54	-29.31	-28.67
all	all	*NS*	Rank	3	2	**1**	6	5	4

Table A.40 Overall wins and losses for type II DMOPs across all performance measures and n_t-τ_t combinations

Results	**QPSO Strategies**					
	QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
Wins	237.32	609.19	630.73	159.2	215.93	228.64
Losses	496.25	157.9	153.48	530.08	370	373.3
Diff	-258.93	451.29	477.25	-370.88	-154.07	-144.66
Rank	5	2	**1**	6	4	3

Table A.41 Overall wins and losses by various frequencies and severities of change for type II DMOPs across all performance measures

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**QPSO Strategies**					
				QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
10	10	all	Wins	62.68	172.46	186.12	33.65	58.03	63.78
10	10	all	Losses	137.83	39.75	35.79	156.96	103.31	103.08
10	10	all	Diff	-75.15	132.71	150.33	-123.31	-45.28	-39.3
10	10	all	Rank	5	2	**1**	6	4	3
10	25	all	Wins	20.69	73.08	70.89	15.73	22.47	22.95
10	25	all	Losses	55.09	14.95	16.39	55.33	42.06	41.99
10	25	all	Diff	-34.4	58.13	54.5	-39.6	-19.59	-19.04
10	25	all	Rank	5	**1**	2	6	4	3
10	50	all	Wins	9.06	34.46	33.42	9.8	11	10.79
10	50	all	Losses	28.12	8.28	8.6	24.14	19.56	19.83
10	50	all	Diff	-19.06	26.18	24.82	-14.34	-8.56	-9.04
10	50	all	Rank	6	**1**	2	5	3	4
1	10	all	Wins	93.65	159.93	159.95	60.99	70.63	71.93
1	10	all	Losses	138.03	56.66	56.41	157.03	102.98	105.97
1	10	all	Diff	-44.38	103.27	103.54	-96.04	-32.35	-34.04
1	10	all	Rank	5	2	**1**	6	3	4
20	10	all	Wins	51.24	169.26	180.35	39.03	53.8	59.19
20	10	all	Losses	137.18	38.26	36.29	136.62	102.09	102.43
20	10	all	Diff	-85.94	131	144.06	-97.59	-48.29	-43.24
20	10	all	Rank	5	2	**1**	6	4	3

Table A.42 Overall wins and losses by various performance measures for Type II DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**QPSO Strategies**					
				QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
all	all	*S*	Wins	34.62	119.9	136.47	20.65	22.94	24.7
all	all	*S*	Losses	87.93	14.93	9.07	94.09	76.46	76.8
all	all	*S*	Diff	-53.31	104.97	127.4	-73.44	-53.52	-52.1
all	all	*S*	Rank	4	2	**1**	6	5	3
all	all	*VD*	Wins	43.96	133.13	114.66	43.09	64.24	64.47
all	all	*VD*	Losses	120.46	38.88	48.16	110.09	70.52	75.44
all	all	*VD*	Diff	-76.5	94.25	66.5	-67	-6.28	-10.97
all	all	*VD*	Rank	6	**1**	2	5	3	4
all	all	*MS*	Wins	32.96	45.25	66.18	10.05	16.77	18.9
all	all	*MS*	Losses	35.8	19.28	10.55	57.44	34.68	32.36
all	all	*MS*	Diff	-2.84	25.97	55.63	-47.39	-17.91	-13.46
all	all	*MS*	Rank	3	2	**1**	6	5	4
all	all	*acc*	Wins	57.26	142.08	141.96	37.55	58.24	62.66
all	all	*acc*	Losses	119.29	44.83	42.7	126.26	83.06	83.61
all	all	*acc*	Diff	-62.03	97.25	99.26	-88.71	-24.82	-20.95
all	all	*acc*	Rank	5	2	**1**	6	4	3
all	all	*stab*	Wins	35.01	46.45	48.34	31.87	29.61	30.71
all	all	*stab*	Losses	51.07	30.05	29.58	44.48	33.24	33.57
all	all	*stab*	Diff	-16.06	16.4	18.76	-12.61	-3.63	-2.86
all	all	*stab*	Rank	6	2	**1**	5	4	3
all	all	*NS*	Wins	33.51	122.38	123.12	15.99	24.13	27.2
all	all	*NS*	Losses	81.7	9.93	13.42	97.72	72.04	71.52
all	all	*NS*	Diff	-48.19	112.45	109.7	-81.73	-47.91	-44.32
all	all	*NS*	Rank	5	**1**	2	6	4	3

Table A.43 Overall wins and losses for type III DMOPs across all performance measures and n_t-τ_t combinations

Results	**QPSO Strategies**					
	QPSO$_n$	**QPSO$_r$**	**QPSO$_t$**	**PCX$_n$**	**PCX$_r$**	**PCX$_t$**
Wins	111.02	84.83	102.75	39.58	41.25	32.92
Losses	49.38	50.68	26.56	95.65	100.45	89.63
Diff	61.64	34.15	76.19	-56.07	-59.2	-56.71
Rank	2	3	**1**	4	6	5

Table A.44 Overall wins and losses by various frequencies and severities of change for type III DMOPs across all performance measures

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**QPSO Strategies**					
				QPSO$_n$	**QPSO$_r$**	**QPSO$_t$**	**PCX$_n$**	**PCX$_r$**	**PCX$_t$**
10	10	all	Wins	29.48	21.26	28.63	11.08	10.12	7.81
10	10	all	Losses	13.65	16	5.53	23.8	26.62	22.78
10	10	all	Diff	15.83	5.26	23.1	-12.72	-16.5	-14.97
10	10	all	Rank	2	3	**1**	4	6	5
10	25	all	Wins	9.46	7.97	9.71	3.05	4.01	3.63
10	25	all	Losses	4.42	3.9	2.45	9.75	8.9	8.41
10	25	all	Diff	5.04	4.07	7.26	-6.7	-4.89	-4.78
10	25	all	Rank	2	3	**1**	6	5	4
10	50	all	Wins	3.49	3.92	3.88	1.42	1.89	1.65
10	50	all	Losses	2.22	1.17	1.49	4.51	3.6	3.26
10	50	all	Diff	1.27	2.75	2.39	-3.09	-1.71	-1.61
10	50	all	Rank	3	**1**	2	6	5	4
1	10	all	Wins	42.87	33.69	37.83	15.66	16.16	12.7
1	10	all	Losses	17.15	17.17	12.43	37.63	38.08	36.45
1	10	all	Diff	25.72	16.52	25.4	-21.97	-21.92	-23.75
1	10	all	Rank	**1**	3	2	5	4	6
20	10	all	Wins	25.72	17.99	22.7	8.37	9.07	7.13
20	10	all	Losses	11.94	12.44	4.66	19.96	23.25	18.73
20	10	all	Diff	13.78	5.55	18.04	-11.59	-14.18	-11.6
20	10	all	Rank	2	3	**1**	4	6	5

Table A.45 Overall wins and losses by various performance measures for type III DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**QPSO Strategies**					
				QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
all	all	*S*	Wins	34.2	30.89	35.79	2.44	3.71	3.43
all	all	*S*	Losses	6.28	5.83	1.82	33.41	32.29	30.83
all	all	*S*	Diff	27.92	25.06	33.97	-30.97	-28.58	-27.4
all	all	*S*	Rank	2	3	**1**	6	5	4
all	all	*VD*	Wins	7.46	15.81	9.88	10.03	6.99	7.22
all	all	*VD*	Losses	11.1	5.53	8.09	6.29	11	15.38
all	all	*VD*	Diff	-3.64	10.28	1.79	3.74	-4.01	-8.16
all	all	*VD*	Rank	4	**1**	3	2	5	6
all	all	*MS*	Wins	22.67	2.05	7.16	4.48	9.53	3.78
all	all	*MS*	Losses	5.97	12.81	3.35	10.02	11.07	6.45
all	all	*MS*	Diff	16.7	-10.76	3.81	-5.54	-1.54	-2.67
all	all	*MS*	Rank	**1**	6	2	5	3	4
all	all	*acc*	Wins	30.81	26.08	37.13	13.43	9.66	10.57
all	all	*acc*	Losses	13.96	14.05	8.26	31.02	32.26	28.13
all	all	*acc*	Diff	16.85	12.03	28.87	-17.59	-22.6	-17.56
all	all	*acc*	Rank	2	3	**1**	5	6	4
all	all	*stab*	Wins	7.78	5.94	7.37	5.16	6.31	5.57
all	all	*stab*	Losses	6.57	5.99	4.75	7.81	6.8	6.21
all	all	*stab*	Diff	1.21	-0.05	2.62	-2.65	-0.49	-0.64
all	all	*stab*	Rank	2	3	**1**	6	4	5
all	all	*NS*	Wins	8.1	4.06	5.42	4.04	5.05	2.35
all	all	*NS*	Losses	5.5	6.47	0.29	7.1	7.03	2.63
all	all	*NS*	Diff	2.6	-2.41	5.13	-3.06	-1.98	-0.28
all	all	*NS*	Rank	2	5	**1**	6	4	3

A.3.2 Results for MGPSO with 10% Proportion of Quantum Particles

This appendix lists experimental results for Type I, Type II, and Type III DMOPs by the MGPSO with self-adaptive quantum particles and PCX quantum particles at 10% quantum proportion.

Table A.46 Overall wins and losses for type I DMOPs across all performance measures and n_t-τ_t combinations

Results	**QPSO Strategies**					
	QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
Wins	66.37	51.59	71.25	28.81	26.7	27.64
Losses	42.85	34.08	34.01	61.02	48.62	51.78
Diff	23.52	17.51	37.24	-32.21	-21.92	-24.14
Rank	2	3	**1**	6	4	5

Table A.47 Overall wins and losses by various frequencies and severities of change for type I DMOPs across all performance measures

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**QPSO Strategies**					
				QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
10	10	all	Wins	15.37	11.88	19.73	2.51	2.83	3.27
10	10	all	Losses	9.09	4.43	3.39	11.61	14.64	12.43
10	10	all	Diff	6.28	7.45	16.34	-9.1	-11.81	-9.16
10	10	all	Rank	3	2	**1**	4	6	5
10	25	all	Wins	5.02	7.76	8.95	0.38	0.91	2.53
10	25	all	Losses	3.86	0.9	0.59	8.9	6.88	4.42
10	25	all	Diff	1.16	6.86	8.36	-8.52	-5.97	-1.89
10	25	all	Rank	3	2	**1**	6	5	4
10	50	all	Wins	0.9	3.28	4	0.24	0.58	0.55
10	50	all	Losses	2.04	0.31	0.28	3	1.74	2.18
10	50	all	Diff	-1.14	2.97	3.72	-2.76	-1.16	-1.63
10	50	all	Rank	3	2	**1**	6	4	5
1	10	all	Wins	31.95	9.6	12.52	23.02	18.67	18.46
1	10	all	Losses	16.59	25.62	27.2	17.65	12.07	15.09
1	10	all	Diff	15.36	-16.02	-14.68	5.37	6.6	3.37
1	10	all	Rank	**1**	6	5	3	2	4
20	10	all	Wins	13.13	19.07	26.05	2.66	3.71	2.83
20	10	all	Losses	11.27	2.82	2.55	19.86	13.29	17.66
20	10	all	Diff	1.86	16.25	23.5	-17.2	-9.58	-14.83
20	10	all	Rank	3	2	**1**	6	4	5

Table A.48 Overall wins and losses by various performance measures for type I DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**QPSO Strategies**					
				QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
all	all	*S*	Wins	4.41	6.17	8.48	6.72	4.25	4.79
all	all	*S*	Losses	9.78	4.67	5.77	5.7	4.38	4.52
all	all	*S*	Diff	-5.37	1.5	2.71	1.02	-0.13	0.27
all	all	*S*	Rank	6	2	**1**	3	5	4
all	all	*VD*	Wins	18.57	15.85	20.41	5.8	6.3	6.29
all	all	*VD*	Losses	9.61	8.28	8.83	17.8	13.65	15.05
all	all	*VD*	Diff	8.96	7.57	11.58	-12	-7.35	-8.76
all	all	*VD*	Rank	2	3	**1**	6	4	5
all	all	*MS*	Wins	8.98	4.46	6.88	2.29	2.74	3.14
all	all	*MS*	Losses	2.09	3.44	2.12	7.29	6.92	6.63
all	all	*MS*	Diff	6.89	1.02	4.76	-5	-4.18	-3.49
all	all	*MS*	Rank	**1**	3	2	6	5	4
all	all	*acc*	Wins	16.47	9.68	15.15	5.52	6.36	6.09
all	all	*acc*	Losses	7.95	9.44	8.96	13.09	9.38	10.45
all	all	*acc*	Diff	8.52	0.24	6.19	-7.57	-3.02	-4.36
all	all	*acc*	Rank	**1**	3	2	6	4	5
all	all	*stab*	Wins	3.49	2.94	2.98	4.1	3.42	3.27
all	all	*stab*	Losses	7.03	2.53	2.85	3.29	2.16	2.34
all	all	*stab*	Diff	-3.54	0.41	0.13	0.81	1.26	0.93
all	all	*stab*	Rank	6	4	5	3	**1**	2
all	all	*NS*	Wins	14.45	12.49	17.35	4.38	3.63	4.06
all	all	*NS*	Losses	6.39	5.72	5.48	13.85	12.13	12.79
all	all	*NS*	Diff	8.06	6.77	11.87	-9.47	-8.5	-8.73
all	all	*NS*	Rank	2	3	**1**	6	4	5

Table A.49 Overall wins and losses for type II DMOPs across all performance measures and n_t-τ_t combinations

Results	**QPSO Strategies**					
	QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
Wins	160.72	200.88	226.96	92.78	111.54	116.51
Losses	195.93	92.81	89.42	228.3	152.84	150.09
Diff	-35.21	108.07	137.54	-135.52	-41.3	-33.58
Rank	4	2	**1**	6	5	3

Table A.50 Overall wins and losses by various frequencies and severities of change for type II DMOPs across all performance measures

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**QPSO Strategies**					
				QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
10	10	all	Wins	47.24	50.4	62.9	21.41	28.87	30.16
10	10	all	Losses	52.96	25.06	23.3	57.1	41.83	40.73
10	10	all	Diff	-5.72	25.34	39.6	-35.69	-12.96	-10.57
10	10	all	Rank	3	2	**1**	6	5	4
10	25	all	Wins	12.18	24.55	27.09	8.14	9.93	10.67
10	25	all	Losses	21.13	8.48	8.21	22.07	16.54	16.13
10	25	all	Diff	-8.95	16.07	18.88	-13.93	-6.61	-5.46
10	25	all	Rank	5	2	**1**	6	4	3
10	50	all	Wins	5.24	12.99	14.65	5.15	5.13	5.06
10	50	all	Losses	11.78	3.95	4.7	10.76	8.89	8.14
10	50	all	Diff	-6.54	9.04	9.95	-5.61	-3.76	-3.08
10	50	all	Rank	6	2	**1**	5	4	3
1	10	all	Wins	69.3	56.74	59.19	36.37	43.95	42.89
1	10	all	Losses	59.51	34.78	33.13	90.08	44.39	46.55
1	10	all	Diff	9.79	21.96	26.06	-53.71	-0.44	-3.66
1	10	all	Rank	3	2	**1**	6	4	5
20	10	all	Wins	26.76	56.2	63.13	21.71	23.66	27.73
20	10	all	Losses	50.55	20.54	20.08	48.29	41.19	38.54
20	10	all	Diff	-23.79	35.66	43.05	-26.58	-17.53	-10.81
20	10	all	Rank	5	2	**1**	6	4	3

Table A.51 Overall wins and losses by various performance measures for type II DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**QPSO Strategies**					
				QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
all	all	*S*	Wins	27.44	29.48	36.83	12.82	11.22	11.07
all	all	*S*	Losses	23.43	9.9	8.71	33.97	25.69	27.16
all	all	*S*	Diff	4.01	19.58	28.12	-21.15	-14.47	-16.09
all	all	*S*	Rank	3	2	**1**	6	4	5
all	all	*VD*	Wins	32.68	53.13	51.78	21.64	36.22	37.11
all	all	*VD*	Losses	56.52	22.77	27.62	59.77	33.18	32.7
all	all	*VD*	Diff	-23.84	30.36	24.16	-38.13	3.04	4.41
all	all	*VD*	Rank	5	**1**	2	6	4	3
all	all	*MS*	Wins	14.08	9.69	19.69	5.7	5.21	6.16
all	all	*MS*	Losses	11.29	8.71	4.86	16.54	10.24	8.89
all	all	*MS*	Diff	2.79	0.98	14.83	-10.84	-5.03	-2.73
all	all	*MS*	Rank	2	3	**1**	6	5	4
all	all	*acc*	Wins	41.24	57.91	62.93	21.77	31.2	33.56
all	all	*acc*	Losses	54.48	27.68	24.46	59.64	41.71	40.64
all	all	*acc*	Diff	-13.24	30.23	38.47	-37.87	-10.51	-7.08
all	all	*acc*	Rank	5	2	**1**	6	4	3
all	all	*stab*	Wins	16.97	18.31	20.65	17.65	16.16	16.5
all	all	*stab*	Losses	27.99	14.03	13.59	21.06	15.31	14.26
all	all	*stab*	Diff	-11.02	4.28	7.06	-3.41	0.85	2.24
all	all	*stab*	Rank	6	2	**1**	5	4	3
all	all	*NS*	Wins	28.31	32.36	35.08	13.2	11.53	12.11
all	all	*NS*	Losses	22.22	9.72	10.18	37.32	26.71	26.44
all	all	*NS*	Diff	6.09	22.64	24.9	-24.12	-15.18	-14.33
all	all	*NS*	Rank	3	2	**1**	6	5	4

Table A.52 Overall wins and losses for type III DMOPs across all performance measures and n_t-τ_t combinations

Results	**QPSO Strategies**					
	QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
Wins	19.01	14.01	16.72	18.53	12.27	18.5
Losses	12.59	19.82	15.8	18.05	16.21	16.57
Diff	6.42	-5.81	0.92	0.48	-3.94	1.93
Rank	**1**	6	3	4	5	2

Table A.53 Overall wins and losses by various frequencies and severities of change for type III DMOPs across all performance measures

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**QPSO Strategies**					
				QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
10	10	all	Wins	6.07	3.67	4.3	5.95	1.64	6.18
10	10	all	Losses	4.31	5.11	4.1	4.51	4.8	4.98
10	10	all	Diff	1.76	-1.44	0.2	1.44	-3.16	1.2
10	10	all	Rank	**1**	5	4	2	6	3
10	25	all	Wins	2.42	1.62	1.92	0.97	0.62	0.73
10	25	all	Losses	1.07	0.82	0.77	2.18	1.54	1.9
10	25	all	Diff	1.35	0.8	1.15	-1.21	-0.92	-1.17
10	25	all	Rank	**1**	3	2	6	4	5
10	50	all	Wins	0.44	0.6	1.01	1.11	0.59	0.68
10	50	all	Losses	0.51	1.19	0.77	0.69	0.59	0.68
10	50	all	Diff	-0.07	-0.59	0.24	0.42	0	0
10	50	all	Rank	5	6	2	**1**	4	4
1	10	all	Wins	5.74	5.75	6.15	5.8	7.15	5.95
1	10	all	Losses	4.12	7.42	5.8	7.69	5.58	5.93
1	10	all	Diff	1.62	-1.67	0.35	-1.89	1.57	0.02
1	10	all	Rank	**1**	5	3	6	2	4
20	10	all	Wins	4.34	2.37	3.34	4.7	2.27	4.96
20	10	all	Losses	2.58	5.28	4.36	2.98	3.7	3.08
20	10	all	Diff	1.76	-2.91	-1.02	1.72	-1.43	1.88
20	10	all	Rank	2	6	4	3	5	**1**

Table A.54 Overall wins and losses by various performance measures for type III DMOPs across all n_t-τ_t combinations

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**QPSO Strategies**					
				QPSO$_n$	**QPSO**$_r$	**QPSO**$_t$	**PCX**$_n$	**PCX**$_r$	**PCX**$_t$
all	all	*S*	Wins	6.59	4.92	5.93	0.6	1.52	0.99
all	all	*S*	Losses	1.08	1.38	0.61	7.27	5.26	4.95
all	all	*S*	Diff	5.51	3.54	5.32	-6.67	-3.74	-3.96
all	all	*S*	Rank	**1**	3	2	6	4	5
all	all	*VD*	Wins	1.44	4.71	4.61	1.22	1.62	1.07
all	all	*VD*	Losses	1.95	0.8	0.74	4.25	2.78	4.15
all	all	*VD*	Diff	-0.51	3.91	3.87	-3.03	-1.16	-3.08
all	all	*VD*	Rank	3	**1**	2	5	4	6
all	all	*MS*	Wins	3.08	0.31	0.48	7.01	2.14	6.14
all	all	*MS*	Losses	2.26	7.81	6.18	0.81	1.03	1.07
all	all	*MS*	Diff	0.82	-7.5	-5.7	6.2	1.11	5.07
all	all	*MS*	Rank	4	6	5	**1**	3	2
all	all	*acc*	Wins	6.61	2.95	4.29	5.08	3.98	4.65
all	all	*acc*	Losses	4.01	5.12	3.62	3.75	5.77	5.29
all	all	*acc*	Diff	2.6	-2.17	0.67	1.33	-1.79	-0.64
all	all	*acc*	Rank	**1**	6	3	2	5	4
all	all	*stab*	Wins	1.29	1.09	1.38	2.49	1.72	2.47
all	all	*stab*	Losses	1.87	2.45	2.36	1.4	1.3	1.06
all	all	*stab*	Diff	-0.58	-1.36	-0.98	1.09	0.42	1.41
all	all	*stab*	Rank	4	6	5	2	3	**1**
all	all	*NS*	Wins	0	0.03	0.03	2.13	1.29	3.18
all	all	*NS*	Losses	1.42	2.26	2.29	0.57	0.07	0.05
all	all	*NS*	Diff	-1.42	-2.23	-2.26	1.56	1.22	3.13
all	all	*NS*	Rank	4	5	6	2	3	**1**

A.3.3 Comparisons with the other DMOAs

This appendix lists experimental results for Type I, Type II, and Type III DMOPs by the MGPSO without any quantum particles, MGPSO with 10% and 50% of self-adaptive quantum particles, DMOES, DNSGA-II, and SGEA

Table A.55 Overall wins and losses for type I DMOPs across all performance measures and n_t-τ_t combinations

Results	**DMOAs**					
	MGPSO	**MGPSO$_{50}$**	**MGPSO$_{10}$**	**DMOES**	**DNSGA-II**	**SGEA**
Wins	226.14	271.91	265.58	178.8	109.25	214.04
Losses	148.51	112.79	87.26	319.68	352.11	245.37
Diff	77.63	159.12	178.32	-140.88	-242.86	-31.33
Rank	3	2	**1**	5	6	4

Table A.56 Overall wins and losses by various frequencies and severities of change for type I DMOPs across all performance measures

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**Results**	**DMOAs**					
			MGPSO	**MGPSO$_{50}$**	**MGPSO$_{10}$**	**DMOES**	**DNSGA-II**	**SGEA**
10	10	Wins	67.1	74.89	73.25	56.23	25.37	57.37
10	10	Losses	37.2	31.93	22.37	86.03	105.47	71.21
10	10	Diff	29.9	42.96	50.88	-29.8	-80.1	-13.84
10	10	Rank	3	2	**1**	5	6	4
10	25	Wins	24.78	26.29	26.36	23.11	12.85	26.46
10	25	Losses	13.81	15.98	11.95	33.46	39.18	25.47
10	25	Diff	10.97	10.31	14.41	-10.35	-26.33	0.99
10	25	Rank	2	3	**1**	5	6	4
10	50	Wins	11.56	11.42	13.54	11.78	7.09	14.79
10	50	Losses	8.14	9.59	6.76	15.79	18.56	11.34
10	50	Diff	3.42	1.83	6.78	-4.01	-11.47	3.45
10	50	Rank	3	4	**1**	5	6	2
1	10	Wins	61.63	91.07	80.41	28.61	39.51	63.4
1	10	Losses	56.25	20.62	25.03	109.87	89.78	63.08
1	10	Diff	5.38	70.45	55.38	-81.26	-50.27	0.32
1	10	Rank	3	**1**	2	6	5	4
20	10	Wins	61.07	68.24	72.02	59.07	24.43	52.02
20	10	Losses	33.11	34.67	21.15	74.53	99.12	74.27
20	10	Diff	27.96	33.57	50.87	-15.46	-74.69	-22.25
20	10	Rank	3	2	**1**	4	6	5

Table A.57 Overall wins and losses by various performance measures for type I DMOPs across all n_t-τ_t combinations

PM	**Results**	**DMOAs**					
		MGPSO	**MGPSO$_{50}$**	**MGPSO$_{10}$**	**DMOES**	**DNSGA-II**	**SGEA**
S	Wins	51.11	55.69	49.7	10.61	10.65	45.68
S	Losses	15.33	12.07	11.68	75.1	74.38	34.88
S	Diff	35.78	43.62	38.02	-64.49	-63.73	10.8
S	Rank	3	**1**	2	6	5	4
VD	Wins	47.99	51	57.6	27.86	18.12	49.63
VD	Losses	29.51	28.91	17.81	56.23	75.51	44.23
VD	Diff	18.48	22.09	39.79	-28.37	-57.39	5.4
VD	Rank	3	2	**1**	5	6	4
MS	Wins	35.52	48.53	46.86	29.61	27.96	16.59
MS	Losses	24.61	12.84	7.59	55.57	43.76	60.7
MS	Diff	10.91	35.69	39.27	-25.96	-15.8	-44.11
MS	Rank	3	2	**1**	5	4	6
acc	Wins	35.04	41.18	44.02	67.86	12.86	40.28
acc	Losses	34.37	34.52	24.34	29.37	72.45	46.19
acc	Diff	0.67	6.66	19.68	38.49	-59.59	-5.91
acc	Rank	4	3	2	**1**	6	5
$stab$	Wins	26.61	24.14	23.45	34.28	21.64	13.53
$stab$	Losses	15.98	15.94	16.3	25.27	31.89	38.27
$stab$	Diff	10.63	8.2	7.15	9.01	-10.25	-24.74
$stab$	Rank	**1**	3	4	2	5	6
NS	Wins	29.87	51.37	43.95	8.58	18.02	48.33
NS	Losses	28.71	8.51	9.54	78.14	54.12	21.1
NS	Diff	1.16	42.86	34.41	-69.56	-36.1	27.23
NS	Rank	4	**1**	2	6	5	3

Table A.58 Overall wins and losses for type II DMOPs across all performance measures and n_t-τ_t combinations

Results	**DMOAs**					
	MGPSO	**MGPSO$_{50}$**	**MGPSO$_{10}$**	**DMOES**	**DNSGA-II**	**SGEA**
Wins	644.09	665.89	701.71	593.8	337.99	665.53
Losses	390.43	447.82	324.35	823.29	972.49	650.63
Diff	253.66	218.07	377.36	-229.49	-634.5	14.9
Rank	2	3	**1**	5	6	4

Table A.59 Overall wins and losses by various frequencies and severities of change for type II DMOPs across all performance measures

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**Results**	**DMOAs**					
			MGPSO	**MGPSO$_{50}$**	**MGPSO$_{10}$**	**DMOES**	**DNSGA-II**	**SGEA**
10	10	Wins	180.01	189	201.83	167.12	85.7	171.64
10	10	Losses	99.52	117.87	81.35	221.7	281.16	193.7
10	10	Diff	80.49	71.13	120.48	-54.58	-195.46	-22.06
10	10	Rank	2	3	**1**	5	6	4
10	25	Wins	63.24	66.91	72.08	81.41	36.4	80.59
10	25	Losses	49.02	55.43	39.34	78.45	110.43	67.96
10	25	Diff	14.22	11.48	32.74	2.96	-74.03	12.63
10	25	Rank	2	4	**1**	5	6	3
10	50	Wins	29.77	30.92	32.9	41.03	21.2	41.77
10	50	Losses	25.96	29.5	22.65	37.75	49.45	32.28
10	50	Diff	3.81	1.42	10.25	3.28	-28.25	9.49
10	50	Rank	3	5	**1**	4	6	2
1	10	Wins	199.23	205.58	207.43	104.67	110.74	198.43
1	10	Losses	107.55	120.47	90.69	293.41	248.44	165.52
1	10	Diff	91.68	85.11	116.74	-188.74	-137.7	32.91
1	10	Rank	2	3	**1**	6	5	4
20	10	Wins	171.84	173.48	187.47	199.57	83.95	173.1
20	10	Losses	108.38	124.55	90.32	191.98	283.01	191.17
20	10	Diff	63.46	48.93	97.15	7.59	-199.06	-18.07
20	10	Rank	2	3	**1**	4	6	5

Table A.60 Overall wins and losses by various performance measures for type II DMOPs across all n_t-τ_t combinations

PM	**Results**	**DMOAs**					
		MGPSO	**MGPSO$_{50}$**	**MGPSO$_{10}$**	**DMOES**	**DNSGA-II**	**SGEA**
S	Wins	129.66	126.53	130.81	59.24	42.65	137.3
S	Losses	52.38	66.94	47.28	167.87	199.51	92.21
S	Diff	77.28	59.59	83.53	-108.63	-156.86	45.09
S	Rank	2	3	**1**	5	6	4
VD	Wins	128.53	105.94	132.11	119.1	46.31	153.82
VD	Losses	75.54	116.08	68.48	124.52	208.55	92.64
VD	Diff	52.99	-10.14	63.63	-5.42	-162.24	61.18
VD	Rank	3	5	**1**	4	6	2
MS	Wins	116.34	148.03	136.76	64.48	77.4	53.19
MS	Losses	44.49	17.53	26.61	196.57	118.69	192.31
MS	Diff	71.85	130.5	110.15	-132.09	-41.29	-139.12
MS	Rank	3	**1**	2	5	4	6
acc	Wins	111.6	117.63	135.94	201.38	40.24	117.67
acc	Losses	100.98	116.57	79.14	63.88	227.33	136.56
acc	Diff	10.62	1.06	56.8	137.5	-187.09	-18.89
acc	Rank	3	4	2	**1**	6	5
$stab$	Wins	64.37	69.83	64.47	113.16	66.44	70.87
$stab$	Losses	65.13	64.36	58.76	65.5	102.9	92.49
$stab$	Diff	-0.76	5.47	5.71	47.66	-36.46	-21.62
$stab$	Rank	4	3	2	**1**	6	5
NS	Wins	93.59	97.93	101.62	36.44	64.95	132.68
NS	Losses	51.91	66.34	44.08	204.95	115.51	44.42
NS	Diff	41.68	31.59	57.54	-168.51	-50.56	88.26
NS	Rank	3	4	2	6	5	**1**

Table A.61 Overall wins and losses for type III DMOPs across all performance measures and n_t-τ_t combinations

Results	**DMOAs**					
	MGPSO	**MGPSO$_{50}$**	**MGPSO$_{10}$**	**DMOES**	**DNSGA-II**	**SGEA**
Wins	221.17	227.17	236.09	146.14	108.63	139.9
Losses	60.83	67.1	41.96	323.18	288.54	297.49
Diff	160.34	160.07	194.13	-177.04	-179.91	-157.59
Rank	2	3	**1**	5	6	4

Table A.62 Overall wins and losses by various frequencies and severities of change for type III DMOPs across all performance measures

n_t	τ_t	Results	DMOAs					
			MGPSO	**$MGPSO_{50}$**	**$MGPSO_{10}$**	**DMOES**	**DNSGA-II**	**SGEA**
10	10	Wins	62.08	64.8	68.43	42.72	28.49	37.05
10	10	Losses	16.77	18.49	9.28	88	84.57	86.46
10	10	Diff	45.31	46.31	59.15	-45.28	-56.08	-49.41
10	10	Rank	3	2	**1**	4	6	5
10	25	Wins	23.81	24.45	26.41	16.45	11.49	16.23
10	25	Losses	6.99	6.97	4.74	34.58	33.06	32.5
10	25	Diff	16.82	17.48	21.67	-18.13	-21.57	-16.27
10	25	Rank	3	2	**1**	5	6	4
10	50	Wins	11.49	11.87	12.2	8.39	6.01	8.73
10	50	Losses	3.42	3.52	3.23	17.01	15.77	15.74
10	50	Diff	8.07	8.35	8.97	-8.62	-9.76	-7.01
10	50	Rank	3	2	**1**	5	6	4
1	10	Wins	63.97	62.06	63.13	37.18	34.2	39.49
1	10	Losses	15.45	20.09	14.13	97.08	74.07	79.21
1	10	Diff	48.52	41.97	49	-59.9	-39.87	-39.72
1	10	Rank	2	3	**1**	6	5	4
20	10	Wins	59.82	63.99	65.92	41.4	28.44	38.4
20	10	Losses	18.2	18.03	10.58	86.51	81.07	83.58
20	10	Diff	41.62	45.96	55.34	-45.11	-52.63	-45.18
20	10	Rank	3	2	**1**	4	6	5

Table A.63 Overall wins and losses by various performance measures for type III DMOPs across all n_t-τ_t combinations

PM	Results	DMOAs					
		MGPSO	**MGPSO$_{50}$**	**MGPSO$_{10}$**	**DMOES**	**DNSGA-II**	**SGEA**
S	Wins	56.38	61.88	57.76	21.02	13.3	27.71
S	Losses	6.6	4.75	4.8	73.28	80.45	68.17
S	Diff	49.78	57.13	52.96	-52.26	-67.15	-40.46
S	Rank	3	**1**	2	5	6	4
VD	Wins	50.82	46.37	48.65	12.04	19.48	16.45
VD	Losses	3.53	8.69	3.07	60.81	50.63	67.08
VD	Diff	47.29	37.68	45.58	-48.77	-31.15	-50.63
VD	Rank	**1**	3	2	5	4	6
MS	Wins	28.74	32.28	36.03	22.78	31.07	27.72
MS	Losses	14.81	8.98	5.98	68.66	22.13	58.06
MS	Diff	13.93	23.3	30.05	-45.88	8.94	-30.34
MS	Rank	3	2	**1**	6	4	5
acc	Wins	48.25	49.62	55.56	51.74	19.57	35.46
acc	Losses	22.57	27.74	12.97	49.69	81.94	65.29
acc	Diff	25.68	21.88	42.59	2.05	-62.37	-29.83
acc	Rank	2	3	**1**	4	6	5
$stab$	Wins	26.66	27.53	28.35	34.91	8.06	16.44
$stab$	Losses	9.76	10.87	9.75	23.58	51.58	36.41
$stab$	Diff	16.9	16.66	18.6	11.33	-43.52	-19.97
$stab$	Rank	2	3	**1**	4	6	5
NS	Wins	10.32	9.49	9.74	3.65	17.15	16.12
NS	Losses	3.56	6.07	5.39	47.16	1.81	2.48
NS	Diff	6.76	3.42	4.35	-43.51	15.34	13.64
NS	Rank	3	5	4	6	**1**	2