
**University of Tunis**

**High Institute of Management of Tunis**


# Incorporating Decision Maker's Preference Information in Evolutionary Multi-objective Optimization

**by**


**Slim BECHIKH**


**(Defense day: 08 January 2013)**

PhD thesis submitted, on 26 January 2012, to the computer science department of the high institute of management of Tunis in candidature for the diploma of:

## DOCTORATE

**in**

**Computer Science Applied to Management**

**Thesis committee:**

| | | |
|---|---|---|
| **Pr. Henda HAJJAMI BEN GHEZALA** | **University of Manouba** | **Chair** |
| **Pr. Khaled GHEDIRA** | **University of Tunis** | **Advisor** |
| **Pr. Jin-Kao HAO** | **University of Angers** | **Examiner** |
| **Dr. Saoussen KRICHEN KHALFALLAH** | **University of Jendouba** | **Examiner** |
| **Dr. Lamjed BEN SAID** | **University of Tunis** | **Member** |

# Abstract

Most optimization problems often involve multiple objectives to be considered simultaneously under some constraints. Unlike single objective problems, the resolution of this kind of problems gives rise to a set of trade-off solutions, called the Pareto front, rather than a single global optimum. During the two last decades, evolutionary algorithms have demonstrated a great success in approximating the whole Pareto front. Recently, researchers have remarked that providing the human decision maker with some hundreds or thousands of optimal solutions makes the decision making task very difficult especially when the number of objectives increases. In reality, since objective functions are not equally important from the decision maker's viewpoint, this latter is not interested in discovering the whole Pareto front rather than finding only the portion of the front that satisfies his/her preferences which is called the region of interest. For this reason, researchers have mentioned the necessity to hybridize optimization with decision making. The problematic of our PhD thesis is to articulate decision maker's preferences within multi-objective evolutionary algorithms in order to guide the search towards the region of interest; therefore not only facilitating the decision making task but also saving the computational cost required to explore the remainder of the Pareto frontier. In this research work, we categorise preferences into two main classes. The first class concerns explicit preferences which are expressed in a straightforward manner via one of the available preference modelling tools. The second class concerns implicit preferences which correspond to the desire of exploring special points from the Pareto front in the absence of explicit preferences, i.e., knee regions corresponding to the worthiest regions in terms of trade-offs between the objectives and the nadir point corresponding to the vector composed with the worst objective values at the Pareto optimality stage. Additionally, we consider, in this thesis, the case where there exists more than one decision maker each having his/her own preferences. All the proposed contributions are assessed through experimental studies including comparative experiments against the most prominent recent works by utilizing academic benchmark problems commonly used by the community in addition to a practical instance of the portfolio selection problem.

**Keywords:** Multi-objective optimization, evolutionary algorithms, decision making, explicit/implicit decision maker's preferences, region of interest, group preference aggregation/negotiation.

# Acknowledgements

***There is a big difference between being educated***
***and being wise. Education corresponds to knowledge***
***while wisdom corresponds to consciousness***
**(Inspired by Thomas Lombardo 2011).**

Thanks GOD because you gave me the strength to make the decision to initiate this venture and the strength to complete it. Along the way, there were many times in which your love and company were essential to continue. Thanks for holding me in difficult times.

I would like to thank my PhD supervisor, Pr. Khaled GHEDIRA, for giving me the chance to do such a work within the SOIE (Stratégies d'Optimisation et Informatique Intelligente) laboratory and for his valuable comments and critics during our meetings.

I would like to express my infinite gratitude to Dr. Lamjed BEN SAID who has co-supervised this PhD thesis and with whom I began research since my integration within the SOIE laboratory to achieve my Master thesis at that time. Dr. Lamjed was really a great co-supervisor with kind personality attitudes. His care and guidance were very constructive and rich so that they helped me to improve the quality of my works.

A special thank to my friend Mr. Nabil BELGASMI with whom I explored my research field during the achievement of my Master thesis. Scientific discussions with Mr. Nabil were very interesting, worth and at the same time very humorous.

I would like to thank Pr. Carlos Artemio Coello Coello from the National Polytechnic Institute of Mexico not only for answering all my e-mails but also for his great online EMO repository which helps hundreds or may be thousands of researchers all over the world.

I would like to thank Dr. Shamik CHAUDHURI from Kanpur Genetic Algorithms Laboratory (KanGAL), Indian Institute of Technology, Kanpur, India, for providing me useful source code for experimental comparisons.

I would like to thank my PhD committee members for accepting taking part of it and furnishing efforts for examining this work.

A special thank to my dear friend Lamia ELLOUMI, from the University of Twente, Netherlands, for providing me useful scientific resources and helping me to overcome some financial issues.

I would like to thank MoS (Modélisation et Simulation) team members for the nice atmosphere. I express my gratitude to my friends who have always supported me: Zahra KODIA, Mohamed HMIDEN, Rim HELALI, Ghanem SOLTANA, Wiem HAMMAMI, Fatma SIALA KALLEL, Souhir BEN SOUISSI, Héla CHAIEB, just to cite a few.

A special thank to Dr. Soumaya GHAOUI BEL HADJ HMIDA for all the help she has provided for me to overcome the fatigue caused by my tinnitus state since approximately ten years.

Last but not least, I would like to express my special thanks to my parents Béchir and Radhia; and my two brothers Anis and Wissem for all what they have provided for me.

# Contents

# List of figures

# List of tables

# List of acronyms

| | |
|---|---|
| AIS | Artificial Immune System |
| AS | Absolutely Satisfying objective value |
| ASF | Achievement Scalarizing Function |
| BC-EMOA | Brain–Computer Evolutionary Multi-objective Optimization Algorithm |
| CHIM | Convex Hull of Individual Minima |
| CR | Consensus Rate |
| DF | Desirability Function |
| DF-SMS-EMOA | Desirability Function-based SMS-EMOA |
| DM | Decision Maker |
| DTLZ | Deb-Thiele-Laumanns-Zitzler |
| DWA | Dynamic Weighted Aggregation |
| EA | Evolutionary Algorithm |
| EC-NSGA-II | Extremized-Crowded NSGA-II |
| EMO | Evolutionary Multi-objective Optimization |
| ER | Error Ratio |
| ES | Evolutionary Strategy |
| FDM | Finite Difference Method |
| FEs | Function Evaluations |
| GD | Generational Distance |
| G-MOEA | Guided MOEA |
| HLGA | Hajela and Lin Genetic Algorithm |
| HV | HyperVolume |
| IBEA | Indicator-Based Evolutionary Algorithm |
| IGD | Inverted Generational Distance |
| iTDEA | interactive Territory Defining Evolutionary Algorithm |
| JADE | Java Agent DEvelopment framework |
| LBS | Light Beam Search |
| MADA | Multi-Attribute Decision Analysis |
| MCDM | Multi-Criteria Decision Making |
| MI | Marginally Infeasible objective value |
| MMPSO | MaxiMin PSO |

| | |
|---|---|
| MOEA | Multi-objective Optimization Evolutionary Algorithm |
| MOGA | Multi-Objective Genetic Algorithm |
| MOLP | Multi-Objective Linear Programming |
| MONLP | Multi-Objective Non Linear Programming |
| MOP | Multi-objective Optimization Problem |
| MRPUS | Mobile Reference Points Updating Strategy |
| MR-NSGA-II$_N$ | Memetic R-NSGA-II for Nadir point estimation |
| MUA | Marginal Utility Approach |
| NAR | Number of Accepted Requests |
| NARec | Number of Accepted Recommendations |
| NBI | Normal Boundary Intersection |
| NDR | Number of Denied Requests |
| NM | Number of Manipulations |
| NOSGA | Non-Outranking Sorting Genetic Algorithm |
| NPGA | Niched Pareto Genetic Algorithm |
| NRRec | Number of Received Recommendations |
| NSGA | Non-dominated Sorting Genetic Algorithm |
| NSPSO | Non-dominated Sorting PSO |
| NSR | Number of Sent Requests |
| NSS-GPA | Negotiation Support System for Group Preference Aggregation |
| PAES | Pareto Archived Evolutionary Strategy |
| PBEA | Preference-Based Evolutionary Algorithm |
| PESA | Pareto Envelope-based Selection Algorithm |
| PI-EMOA | Progressively Interactive EMO algorithm |
| PLWSO | Parallel Local Weighted Sum Optimization approach |
| PSO | Particle Swarm Optimization |
| r-dominance | reference solution-based dominance |
| r-NSGA-II | reference solution-based NSGA-II |
| R-NSGA-II | Reference point-based NSGA-II |
| ROI | Region Of Interest |
| ROR | Robust Ordinal Regression |
| RPM | Reference Point Method |
| RPSO-SS | Reference point-based Particle Swam Optimization using a Steady State approach |
| S | Spacing metric |
| SC | Set Coverage |
| SBX | Simulated Binary Crossover |

| | |
|---|---|
| SD | Standard Deviation |
| SMS-EMOA | *S* Metric Selection-based EMO Algorithm |
| SOP | Single-objective Optimization Problem |
| SQP | Sequential Quadratic Programming |
| SQP-LS | SQP-based Local Search |
| SPEA | Strength Pareto Evolutionary Approach |
| SPEA2 | Strength Pareto Evolutionary Approach 2 |
| SPM | Simultaneous Perturbation Method |
| SRP | Social Reference Point |
| TDEA | Territory Defining Evolutionary Algorithm |
| TKR-NSGA-II | Trade-off-based KR-NSGA-II |
| T-MRPUS | Trade-off-based MRPUS |
| VEGA | Vector Evaluated Genetic Algorithm |
| VOES | Vector Optimized Evolutionary Strategy |
| WBGA | Weighted-Based Genetic Algorithm |
| WC-NSGA-II | Worst-Crowded NSGA-II |
| WSNA | Weighted Sum Niching Approach |
| ZDT | Zitzler-Deb-Thiele |

# Chapter 1

# Introduction and Overview

## 1.1 Problematic and motivations

Most real world optimization problems encountered in practice often involve multiple objectives to be minimized or maximized simultaneously with respect to a set of constraints (Deb 2001; Coello et al. 2007). These objectives are often conflicting and incommensurable. The decision on a cell phone purchase, for instance, among other examples, can be influenced by several criteria such as the price, the battery life, the weight, the performance and so fourth. Usually, there is no single solution that is optimal with respect to all these objectives at the same time, but rather many different designs exist which are incomparable. Consequently, contrary to Single-objective Optimization Problems (SOPs) where we look for the solution presenting the best performance, the resolution of a Multi-objective Optimization Problem (MOP) gives rise to a set of compromise solutions presenting the optimal trade-offs between the different objectives. When plotted in the objective space, the set of compromise solutions is called the Pareto front. The main goal in multi-objective optimization is to find a well-converged and well-distributed approximation of the Pareto front from which the Decision Maker (DM) will subsequently select his/her preferred alternative to realize. Several methods were proposed in the specialized literature in order to approximate the Pareto front for the discrete case and the continuous one. Mimicking the principles of biological evolution, Evolutionary Algorithms (EAs) have earned popularity in solving MOPs during the two last decades and beyond thanks to two reasons: (1) EAs are able to provide a set of compromise solutions as output on a single run and (2) EAs are insensitive to the shape of the objective functions such as non-convexity, discontinuity, multimodality, non-uniformity of the search space, etc (Deb 2001). As a consequence of the success of Multi-objective Optimization EAs (MOEAs) in handling MOPs, a new branch in the optimization research field has appeared which is called Evolutionary Multi-objective Optimization (EMO). The final goal of MOEAs is to assist the DM to select the final solution which matches at most his/her preferences. Since MOEAs supply the DM with a huge number of solutions, it seems to

be a difficult task to choose the final preferred alternative (Fonseca 2007). In order to facilitate the decision making task, the DM would like to incorporate his/her preferences into the search process. These preferences are used to guide the search towards the preferred part of the Pareto front, i.e., the Region Of Interest (ROI). The ROI is defined as the preferred part of the Pareto optimal region from the DM's perspective (Adra et al. 2007). Our research works focus on the incorporation of DM's preference information in MOEAs in order to direct the search towards the ROI; thereby facilitating the task of selecting the solution to realize.

## 1.2 Research goals and main contributions

Our research goals are essentially the following:

1) **Incorporating explicit DM's preferences in EMO:** This goal is achieved by proposing a new dominance relation based on DM's preferences expressed in an explicit manner (i.e., aspiration levels). This new dominance relation is then incorporated in a MOEA. The resulting preference-based MOEA has demonstrated its ability in providing the Pareto optimal ROI. Subsequently, the DM could select the solution to realize from this preferred region. Additionally, the proposed approach has been shown to outperform several recent works in this research area.

2) **Incorporating implicit DM's preferences in EMO:** DM's preferences could be expressed in an explicit manner (e.g., weights, aspiration levels, trade-off between objectives, and so on (Coello 2000)) or in an implicit manner (i.e., knee regions (Branke et al. 2004) or nadir point (Deb and Miettinen 2008)). Knee regions are potential parts of the Pareto representing the maximal trade-offs between objectives. Such characteristic renders knee regions almost always of particular interest to the DM in practical context (Rachmawati and Srinivasan 2009). Nadir point is the vector composed with the worst objective values over the Pareto optimal front. Hence, a particular DM could be interested only in discovering nadir objective values (Deb et al. 2006b); thereby the nadir point could be seen as another form of implicit DM's preferences. The integration of implicit preference information in EMO is achieved by proposing two preference-based MOEAs: (1) the first one allows the discovery of knee regions and (2) the second one permits the estimation of the nadir point. Comparative experiments show the outperformance of the two proposed algorithms over several recent approaches.

3) **Handling DM group preferences:** There are several decision making situations where there exist more than one DM. The DMs have usually different attitudes and behaviors. Consequently, their preferences are often conflicting. For this reason, we

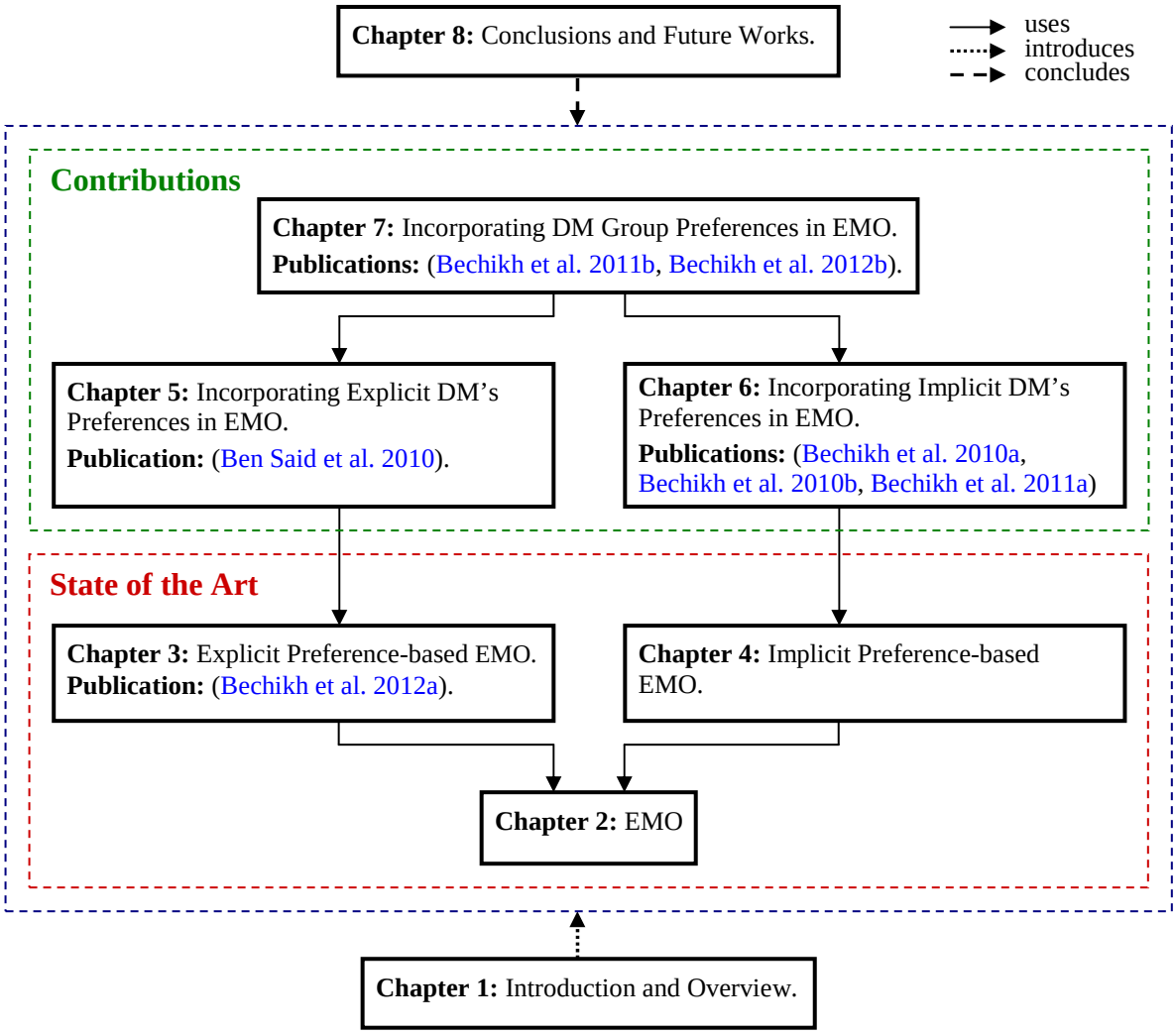



**Figure 1.1 Manuscript structure.**

propose a Negotiation Support System for Group Preference Aggregation (called NSS-GPA) based on software agent paradigm (Wooldridge and Jennings 1994). The designed agent-based system allows the DMs to communicate with each others and to adjust their preferences through a number of discussion rounds. The output of the system is a set of social explicit preferences which will be injected subsequently into the MOEA in order to guide the search towards a social ROI. A solution picked from this social region is considered to be a satisfying solution for all group members.

All proposed preference-based EMO methods in addition to NSS-GPA are validated through several experimental studies involving academic test problems commonly used by the EMO community and valorized by a case study based on a practical instance of the portfolio selection problem (Markowitz 1952; Deb et al. 2011). This latter is described and handled in the last contribution chapter.

## 1.3 Document organization

This document is structured as follows (cf. figure 1.1). Chapter 2 gives a review of MOEAs. Chapter 3 presents a survey of explicit preference-based MOEAs where the algorithms are classified based on the way the preference are expressed and designed. Additionally, this chapter gives a brief review of DM group preference handling. Chapter 4 presents a brief survey of implicit preference-based MOEAs. Chapter 5 is dedicated to present our new preference-based dominance relation in addition to the resulting preference-based MOEA. Chapter 6 focuses on incorporating implicit DM's preferences in EMO and is subdivided into two main parts. The first one is devoted to approximating knee regions. The second one is dedicated to nadir point estimation. Chapter 7 describes the NSS-GPA system. Finally, chapter 8 concludes this thesis and gives some avenues for future research.

# Part I
# State of the Art

# Chapter 2

# Evolutionary Multi-objective Optimization

## 2.1 Introduction

This chapter is devoted to survey evolutionary methods to handle MOPs. In order to solve a MOP, there are three goals to pursue: (1) convergence, (2) diversity and (3) solution distribution uniformity. In fact, the obtained non-dominated solutions should be as close as possible to the Pareto optimal front of the optimization problem. This goal is similar to the demand of convergence to the global optimum in single-objective optimization. Often, there exist an infinite number of Pareto optimal solutions. Naturally, only a finite number of solutions can be generated during an optimization process. Furthermore, the number of generated solutions must be limited otherwise the computational cost would become too large. Nevertheless, the largest possible freedom of choice should be offered to the DM. Therefore, a well-distributed approximation set is demanded which is a goal that consists itself of two requirements: (1) an extent that is as large as possible and (2) a distribution that is as evenly spaced as possible. Pareto optimal fronts may be disconnected, so in that case an exactly uniform distribution of solutions is not possible. Nevertheless, the non-dominated solutions should cover all regions of the Pareto-optimal front and reproduce the curvature of the underlying Pareto optimal front as correctly as possible. These demands do not have a counterpart in single-objective optimization since in that case only one solution is generated. This chapter is structured as follows. Section 2.2 gives some background definitions related to the multi-objective optimization research field. Section 2.3 provides a classification and a discussion of the different proposed methods to tackle MOPs. Section 2.4 discusses the issue of performance assessment by presenting well-cited benchmarks and some selected quality indicators used for MOEA evaluation.

## 2.2 Multi-objective optimization basic definitions

A MOP consists in minimizing or maximizing an objective function vector under some constraints. The general form of a MOP is as follows (Deb 2001):

$$\begin{cases} Min\ f(x) = [f_1(x), f_2(x),..., f_M(x)]^T \\ g_j(x) \geq 0 \qquad\qquad j = 1,...,P; \\ h_k(x) = 0 \qquad\qquad k = 1,...,Q; \\ x_i^L \leq x_i \leq x_i^U \qquad\quad i = 1,...,n. \end{cases} \tag{2.1}$$

where $M$ is the number of objective functions, $P$ is the number of inequality constraints, $Q$ is the number of equality constraints, $x_i^L$ and $x_i^U$ correspond respectively to the lower and upper bounds of the variable $x_i$ (This notation is assumed throughout the overall document). A solution $x_i$ satisfying the ($P$+$Q$) constraints is said feasible and the set of all feasible solutions defines the feasible search space denoted by Ω. In this formulation, we consider a minimization MOP since maximization can be easily turned to minimization based on the duality principle by multiplying each objective function by -1 and transforming constraints based on the duality rules.

The resolution of a MOP yields a set of trade-off solutions, called Pareto optimal solutions or non-dominated solutions, and the image of this set in the objective space is called the Pareto front. Hence, the resolution of a MOP consists in approximating the whole Pareto front. In the following, we give some background definitions related to multi-objective optimization:

**Definition 2.1: Pareto optimality**

A solution $x^* \in \Omega$ is Pareto optimal if $\forall\, x \in \Omega$ and $I = \{1,..., M\}$ either $\forall\ m \in I$ we have $f_m(x) = f_m(x^*)$ or there is at least one $m \in I$ such that $f_m(x) > f_m(x^*)$.

The definition of Pareto optimality states that $x^*$ is Pareto optimal if no feasible vector $x$ exists which would improve some objectives without causing a simultaneous worsening in at least another one.

**Definition 2.2: Pareto dominance**

A solution $u = (u_1, u_2,..., u_n)$ is said to dominate another solution $v = (v_1, v_2,..., v_n)$ (denoted by $f(u) \preceq f(v)$) if and only if $f(u)$ is partially less than $f(v)$. In other words, $\forall\, m \in \{1,..., M\}$ we have $f_m(u) \leq f_m(v)$ and $\exists\, m \in \{1,..., M\}$ where $f_m(u) < f_m(v)$.

**Definition 2.3: Pareto optimal set**

For a given MOP $f(x)$, the Pareto optimal set is $P^* = \{ x \in \Omega \mid \neg \exists\, x' \in \Omega,\ f(x') \preceq f(x) \}$.

**Definition 2.4: Pareto optimal front**

For a given MOP $f(x)$ and its Pareto optimal set $P^*$, the Pareto front is $PF^* = \{ f(x), x \in P^* \}$.

**Definition 2.5: Ideal point**

The ideal point $z^I = (z_1^I, ..., z_M^I)$ is the vector composed by the best objective values over the search space Ω. Analytically, the ideal objective vector is expressed by:

$$z_m{}^I = Min_{x \in \Omega} f_m(x), m \in \{1, ..., M\} \quad (2.2)$$

**Definition 2.6: Nadir point**

The nadir point $z^N = (z_1^N, ..., z_M^N)$ is the vector composed by the worst objective values over the Pareto optimal set. Analytically, the nadir objective vector is expressed by:

$$z_m{}^N = Max_{x \in P^*} f_m(x), m \in \{1, ..., M\} \quad (2.3)$$

**Definition 2.7: ε-dominance**

A solution $u$ is said to epsilon-dominate a solution $v$ ($u \preceq_{\varepsilon+} v$) if and only if $\forall m \in \{1, ..., M\}: u_m \leq v_m + \varepsilon$ for a given $\varepsilon > 0$, where $u_m / v_m$ is the $m^{th}$ objective value of solution $u / v$.

# 2.3 Resolution methods

## 2.3.1 Aggregative methods

Traditional multi-objective optimization methods aggregate the different objective functions into a single one. In order to generate a representative approximation of the whole Pareto front, the user must perform several runs with different parameter settings. Some representatives of this class of methods are the weighted sum method (Cahon 1978), the ε-constraint method (Cahon 1978), the goal programming (Charnes et al. 1955), the reference point method (Wierzbicki 1980), the reference direction method (Korhonen and Laakso 1986a) and the light beam search method (Jaszkiewicz and Slowinski 1999) which are briefly discussed in this subsection.

- **The weighted sum method**

This method converts the MOP to a SOP by forming a linear aggregation of the objectives as follows:

$$\begin{cases} Min \ \ f(x) = w_1 f_1(x) + w_2 f_2(x) + ... + w_M f_M(x) \\ x \in \Omega \end{cases} \quad (2.4)$$

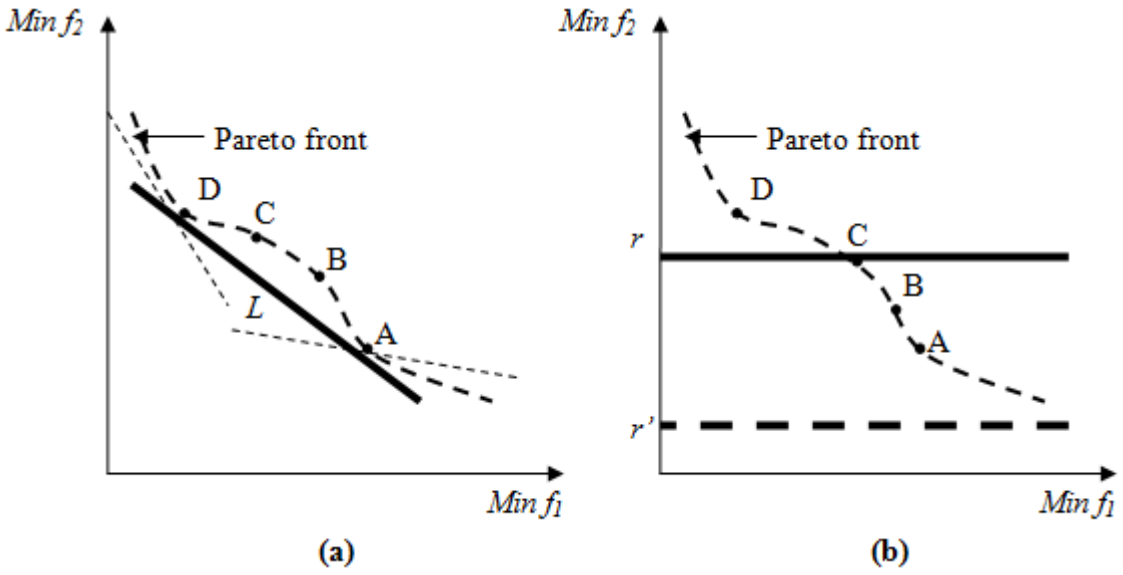



**Figure 2.1 Graphical interpretation of: (a) the weighted sum method and (b) the ε-constraint method (inspired by (**Zitzler 1999**)).**

where $w_m$ corresponds to the weighting coefficient of the $m^{th}$ objective such that $\sum_{m=1}^{M} w_m = 1$ and $w_m \geq 0 \quad \forall m \in \{1,...,M\}$. Solving (2.4) with different weighting coefficients sets yields a set of solutions. Under the condition that an exact optimization algorithm is used and all weighting coefficients are positive, it is easy to show that this method will only generate Pareto optimal solutions. Assuming that a feasible decision vector *u* minimizes $f$ for a given weight combination and is not Pareto optimal, then there is a solution *v* which dominates *u*, i.e., $\forall m \in \{1,...,M\}$ we have $f_m(v) \leq f_m(u)$ and $\exists m \in \{1,...,M\}$ where $f_m(v) < f_m(u)$. Therefore, $f(v) < f(u)$, which is a contradiction to the assumption that $f(u)$ is minimum.

The main disadvantage of this technique is that it cannot generate all Pareto optimal solutions with non-convex trade-off surfaces. This is illustrated in figure 2.1(a). For fixed weights $w_1$ and $w_2$, solution *x* is sought to minimize $y = w_1 f_1(x) + w_2 f_2(x)$. This equation can be formulated as $f_2(x) = -(w_1/w_2) f_1(x) + (y/w_2)$, which defines the line *L* (solid line in figure 2.1(a)) with a slope of $-(w_1/w_2)$ and an intercept of $(y/w_2)$ in the objective space. Graphically, the optimization process corresponds to moving this solid line downwards until no feasible objective vector is above it and at least one feasible objective vector (here *A* and *D*) is on it. However, the points *B* and *C* will never minimize *y*. In fact, if the slope is increased (upper dashed line), *D* achieves a lesser value of *y* than *B* and *C*. Besides, if the slope is decreased (lower dashed line), *A* has a lesser *y* value than *B* and *C*.

♦ **The ε-constraint method**

This method converts the MOP into a SOP by optimizing individually a selected objective while keeping the remaining (*M-1*) objectives' values less than or equal to some user-specified thresholds as follows:

$$\begin{cases} Min \ \ f(x) = f_h(x) & h \in \{1,...,M\} \\ f_m(x) \le \varepsilon_m & m \in \{1,...,M\}, m \ne h; \\ x \in \Omega \end{cases} \tag{2.5}$$

The upper bounds $\varepsilon_m$ are the parameters to be varied in each run in order to obtain multiple Pareto optimal solutions. As depicted in figure 2.1(b), the ε-constraint method is able to find solutions associated with non-convex parts of the Pareto front. Setting $h=1$ and $\varepsilon_2 = r$ (solid line in figure 2.1(b)) makes solution *D* infeasible while solution *C* minimizes $f_1$. Figure 2.1(b) also shows a problem with this technique. In fact, if the lower bounds are not chosen appropriately $(\varepsilon_2 = r')$, the obtained feasible set may be empty, i.e., there is no solution to the obtained SOP. In order to avoid this problem, a suitable range of values for the $\varepsilon_m$ quantities has to be known beforehand.

♦ **The goal-programming method**

For each objective function, the user provides a goal $G_i$ to be achieved. The goal-programming method transforms the MOP into a SOP by minimizing individually the weighted sum of deviations from goals as follows:

$$\begin{cases} Min \ \ f(x) = \sum_{m=1}^{M} w_m \left| f_m(x) - G_m \right| \\ x \in \Omega \end{cases} \tag{2.6}$$

where $w_m$ corresponds to the weighting coefficient of the $m^{th}$ objective such that $\sum_{m=1}^{M} w_m = 1$ and $w_m \ge 0 \ \ \forall m \in \{1,...,M\}$.

As discussed by (Miettinen 1999), if the optimal objective function value of the goal programming method equals zero, then some caution is in order since the obtained solution may not be Pareto optimal. In fact, if all settled goals are feasible, then the value zero for all the deviational variables gives the minimal value (zero) for the goal programming objective function. Hence, the solution is equal to the reference point (the vector composed with all user-specified goals) and normally there exist many feasible solutions that are non Pareto optimal. If the solutions are intended to be Pareto optimal

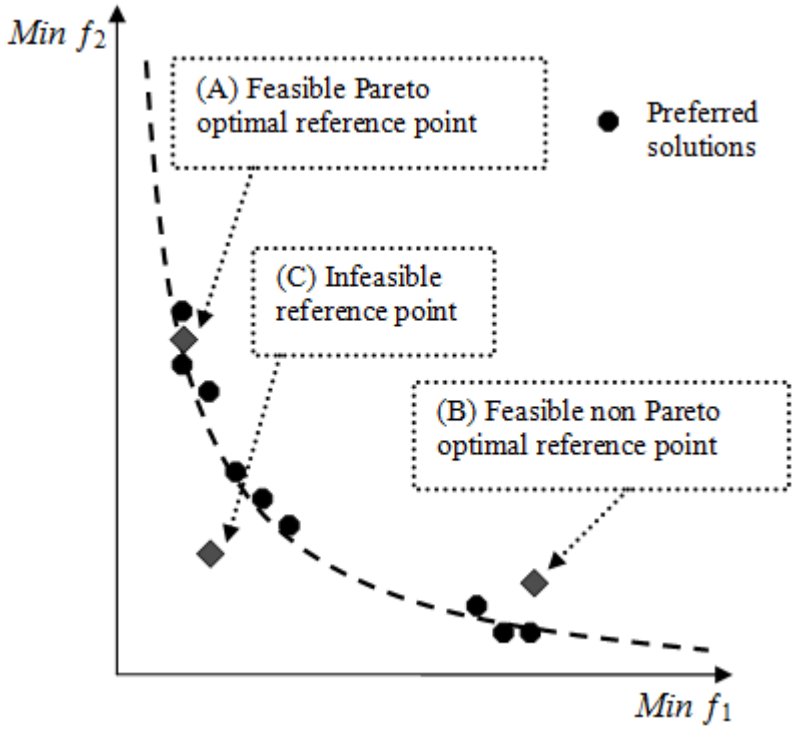



**Figure 2.2 The reference point method.**

independently of the selection of goals, then if the goals are feasible, the function $f$ is to be maximized; else if the goals are infeasible the function $f$ is to be minimized.

♦ **The reference point method**

The classical Reference Point Method (RPM) was proposed by (Wierzbicki 1980). A reference point *g* for a particular MOP consists of an aspiration level vector. Aspiration levels represent the DM's desired values for each objective. This method projects the reference point onto the Pareto optimal region via the minimization of an Achievement Scalarizing Function (ASF). Among the most commonly known forms of an ASF is the following:

$$Min\ \ s(f(x), g) = \underset{m=1,\ldots,M}{Max}\left[w_m\left(f_m(x) - g_m\right)\right] \tag{2.7}$$

where $g_m$ is the $m^{th}$ component of the reference point and $w_m$ is the weight associated with the $m^{th}$ objective.

As shown in figure 2.2, the reference point could be feasible belonging to the Pareto front (A), feasible not belonging to the Pareto front (B) or infeasible (C). For a chosen reference point, the RPM tries to find the closest Pareto optimal solution. The main drawback of this method is that it provides only one solution in a single run. Hence, if the DM is dissatisfied with the obtained solution and/or he/she would like to obtain a

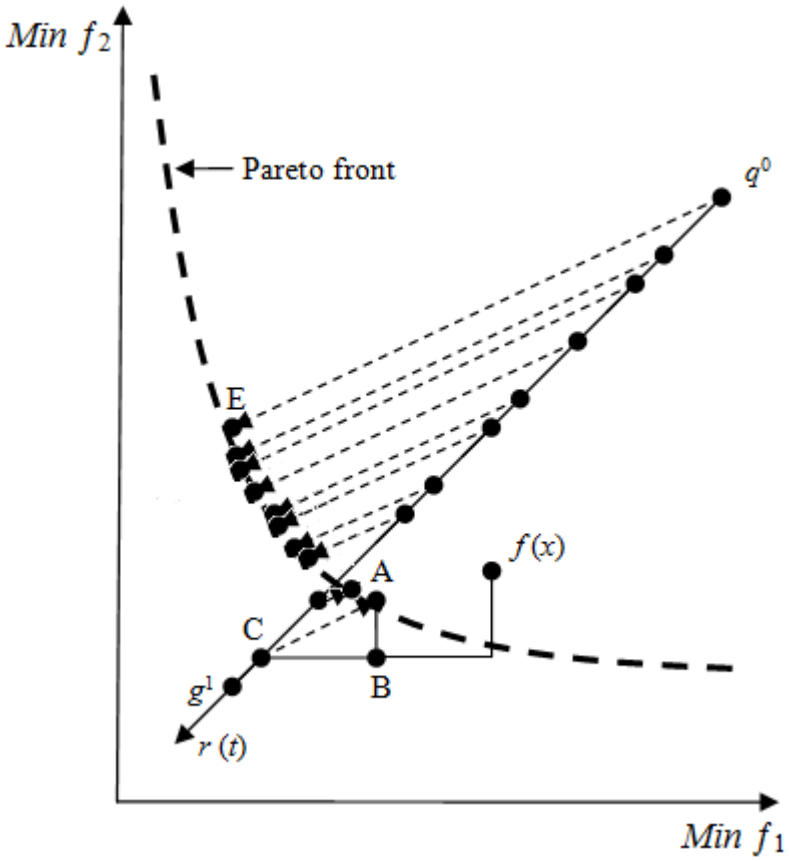



**Figure 2.3 The reference direction method (from** (Deb and Kumar 2007a)**).**

small sample of Pareto optimal solutions near each reference point then he/she must perform several runs of the algorithm. It should be noted that the DM could obtain a sample of near reference point solutions by perturbing the reference point and/or the weights and performing several runs of this method. Besides, in order to make this method interactive, Wierzbicki (1980) suggested a procedure to update the reference point automatically which facilitates the DM's task. When using the reference point approach in practice, the DM is asked to supply a reference point and a weight vector at a time. The reference point guides the search towards the desired region while the weight vector provides more detailed information about which Pareto optimal point to converge to.

♦ **The reference direction method**

Korhonen and Laakso (1986a) suggested a reference direction-based approach for multi-criterion optimization using the principle of solving ASFs repeatedly. This method is described as follows:

- **Step 1**: Choose an initial arbitrary point $q^0$ in the objective space and let $k \leftarrow 1$;

- **Step 2**: Specify another vector $g^k$ and determine the reference direction $d^k = g^k - q^{k-1}$;
- **Step 3**: Determine a set $Q^k$ of efficient solutions $q$ which solves the following ASF:

$$\begin{cases} Min\ \left(f(x), r, w\right) = \underset{w_m > 0, m=1,\ldots,M}{Max} \left[\left(f_m(x) - r_m(t)\right)/w_m\right] \\ r(t) = q^{k-1} + td^k \end{cases} \tag{2.8}$$

  where $t$ is an integer parameter increased from zero to infinity, $w$ is a weighting vector and $r_m(t)$ is the $m^{th}$ component of $r(t)$;
- **Step 4:** Find the most preferred solution $q^k$ in $Q^k$ using a particular utility function or by other mean;
- **Step 5:** If $q^{k-1} \neq q^k$, set $k \leftarrow k + 1$ and go to **step 2**. Otherwise, check for optimality conditions (Kuhn-Tucker conditions (Miettinen 1999) or other optimality conditions (Korhonen and Laasko 1986a)) of the solution $q^k$. If $q^k$ is optimal then terminate the optimization run. Otherwise, increment $k$, determine a new reference direction and go to **Step 3**.

Figure 2.3 shows a sketch of **Step 3** of the above optimization procedure. For each point (say point *C*) marked on the reference direction (from $q^0$ towards $g^1$), a Pareto optimal solution (point *A*) is found by solving the ASF given in equation (2.8). Step 3 of the above procedure involves multiple application of a single-objective optimization for different values of *t*, thereby finding a range of efficient solutions (*A* till *E*). The idea of finding an efficient solution corresponding to a point on a reference direction is similar to the reference point approach of Wierzbicki (1980). Although the original study of the reference direction approach and subsequent studies of Korhonen and his co-authors (Korhonen and Laasko 1986b; Korhonen and Yu 1997) concentrated on parametric solutions for multiple points on the reference direction, the principle can be used by forming multiple ASFs and solving them by a single-objective optimizer independently. An analytical hierarchy process was also used to determine the reference direction (Korhonen 1987). Interestingly, the reference direction approach corresponds to the process of projecting the reference direction on the Pareto optimal frontier.

### ♦ The light beam search method

The Light Beam Search (LBS), as described in (Jaszkiewicz and Slowinski 1999), combines the reference point idea and tools of Multi-Attribute Decision Analysis (MADA). It enables an interactive analysis of MOPs thanks to the presentation of samples of a large set of non-dominated points to the DM in each iteration. An aspiration point and a reservation one should be supplied by the DM. These two points

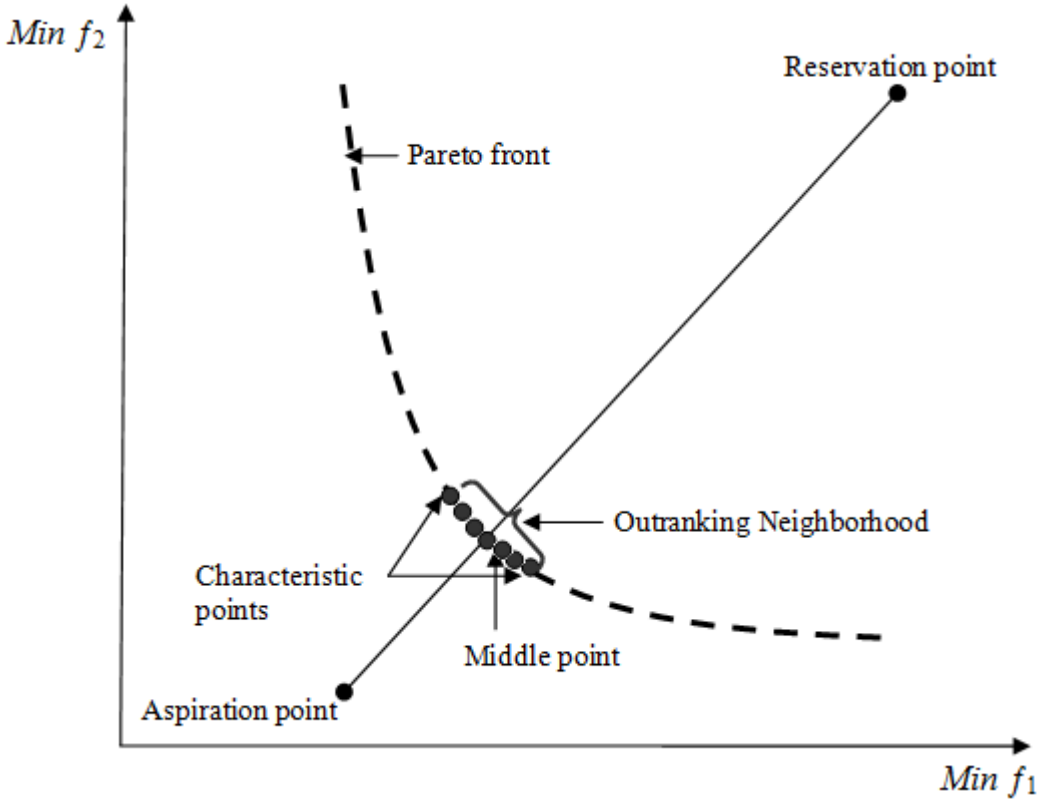



**Figure 2.4 The light beam search method.**

define the direction of the search in a particular iteration. If these two points are not suggested, the ideal point and the nadir point (or a worse point than the nadir one) can be assumed as aspiration and reservation points respectively. Initially a non-dominated middle point is determined by projecting the aspiration point on to the non-dominated front by using an augmented version of Wierzbicki's ASF. Thereafter, a local preference model in the form of an outranking relation *S* is used to obtain neighboring solutions of the current non-dominated point, or the middle point. It is said that *a* outranks *b* (or *a S b*), if *a* is considered to be at least as good as *b*. To define an outranking relation, the DM has to specify three preference thresholds for each objective: (1) indifference threshold, (2) preference threshold and (3) veto threshold. In the LBS procedure, they are considered to provide only local information, thus they are assumed to be constants. The extreme points or characteristic neighbors are found one for each objective by considering the maximum allowed improvement in a particular objective in relation to the middle point. The DM can control the search by either modifying the aspiration and/or reservation points, or by shifting the middle point to selected better point from its neighborhood or by modifying the preference threshold values. Figure 2.4 illustrates the LBS method mechanism. The LBS procedure is as follows:

- **Step 1:** Ask the DM to specify starting aspiration and reservation points;
- **Step 2:** Compute the starting middle point on the Pareto optimal front;
- **Step 3:** Ask DM to specify the local preferential information used to build an outranking relation;
- **Step 4:** Present the middle point to the DM;
- **Step 5:** Calculate the characteristic neighbors of the middle point and present them to the DM;
- **Step 6:** If DM is satisfied, terminate the procedure; else ask the DM to: (1) choose one of the neighboring points to be the new middle point, (2) update the preferential information or (3) define a new aspiration point and/or a new reservation point. The algorithm proceeds by moving to **Step 5** for the case (1) and to **Step 4** otherwise.

### 2.3.2 Evolutionary methods

♦ **Non Pareto-based evolutionary methods**

- **VEGA: Vector Evaluated Genetic Algorithm**

Schaffer (1985) proposed one of the first alternatives to adapt EAs to handle MOPs called VEGA. The basic idea is to divide the population into *M* subpopulations of equal sizes. Then, in each one of them, the selection operates by taking into account only the unique corresponding objective. Once the selection mechanism was performed, the population is mixed to apply the rest of the evolutionary operators. All this process is repeated in each generation. An evident VEGA problem is that it does not promote the survival of good trade-off solutions, but it prefers the best solutions of each objective separately. This problem is known as speciation (by its analogy in genetics). This problem was identified and attacked by Schaffer, using mating restrictions (i.e., not allowing recombination between individuals of the same subpopulation) as well as other heuristic rules applied during the selection mechanism. In another work (Richardson et al. 1989), it was also demonstrated that, if proportional selection is used, VEGA's scheme is equivalent to a linear combination of objective functions which means that it has limitations regarding non-convex Pareto fronts.

- **VOES: Vector Optimized Evolutionary Strategy**

Few years after the VEGA studies, Kursawe (1991) proposed the Vector Optimized Evolutionary Strategy for multi-objective optimization (VOES). The VOES fitness assignment mechanism is similar to VEGA one, but Kursawe used other genetic aspects from nature. In VOES, a solution is represented by a diploid chromosome, each having a dominant string and recessive one. Two different solution vectors (each with a decision variable $x$ and the corresponding strategy vector $\sigma$) are used as an individual in a population. Hence, a solution $x$ is evaluated by calculating: (1) $f^d$ based on the

dominant genotype and (2) $f^r$ based on the recessive genotype. In the following, we present the evaluation and the selection mechanisms. The selection process is performed in *M* steps. For each step, a user-supplied probability vector is used to choose an objective. This vector can be fixed or varied across generations. Assuming the $m^{th}$ objective is selected, the fitness of certain solution *x* is computed as the weighted sum of the dominant objective value and the recessive one as follows:

$$f(x) = \frac{2}{3} f_m^d(x) + \frac{1}{3} f_m^r(x) \tag{2.9}$$

For each selection step, the population is sorted based on each objective function and the best $\left(\frac{M-1}{M}\right)^{th}$ portion of the population is selected as parents. This procedure is repeated *M* times, every time using the survived population from the previous sorting. Thus, the relation between the number of parents $\mu$ and the number of children $\lambda$ can be expressed as follows:

$$\mu = \left(\frac{M-1}{M}\right)^M \lambda \tag{2.10}$$

For example, for the bi-objective case, we obtain $\mu = 0.25\lambda$. All new $\mu$ solutions are copied into an external archive which stores the non-dominated individuals found since the beginning of the simulation run. After adding such solutions to this archive, a non-domination check is performed and only new non-dominated solutions are retained. If the size of the external archive exceeds the archive size, a niching mechanism is used to eliminate crowded solutions in order to promote diversity.

VOES uses non-domination check to ensure convergence and niching to encourage diversity. These features are essential to design a good MOEA. Unfortunately, Kursawe assessed the performance of his algorithm on a single test problem and no further experimental assessments were pursued since Kursawe's original study.

- **WBGA: Weight-Based Genetic Algorithm**

WBGA, also called HLGA (Hajela and Lin Genetic Algorithm), was introduced by (Hajela and Lin 1992). For each objective function, a weighting coefficient is assigned. Unlike the classical weighted sum method, each individual from the population has its own weighting coefficient vector which is coded in its string concatenated to its decision variables. This fact makes the WBGA able to find multiple non-dominated solutions in a single run. The key issue in this algorithm is how to maintain the diversity of weighting coefficients among the population individuals. Tow approaches

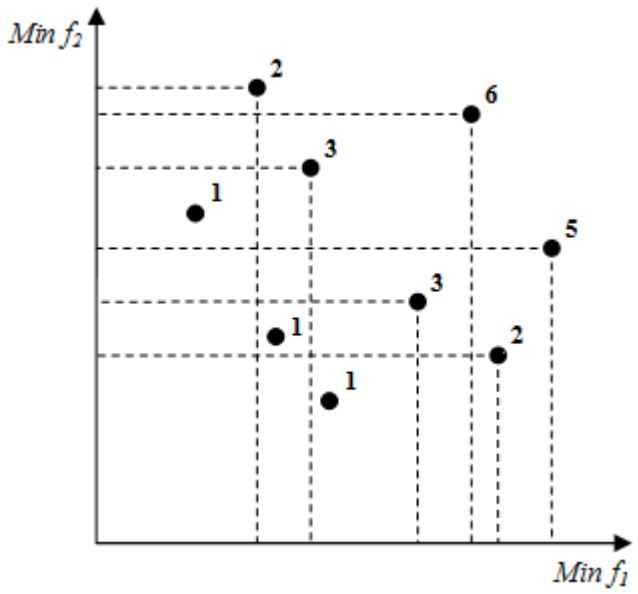



**Figure 2.5 MOGA ranking process.**

were suggested for this sake. In the first approach, a niching mechanism is used on the substring representing the weight coefficient vector. In the second approach, carefully chosen subpopulations are evaluated for different pre-defined weight vectors in a similar way to VEGA. Unfortunately, WBGA is a weight-based approach; hence it fails in finding Pareto optimal solutions residing in the non-convex parts of the front.

- **Pareto-based evolutionary methods**
  - **Non elitist methods**
    - **MOGA: Multi-Objective Genetic Algorithm**

MOGA (Fonseca and Fleming 1993) is the first MOEA which explicitly used Pareto-based ranking and niching techniques together to encourage the search towards the true Pareto front while maintaining diversity in the population. In fact, each individual is assigned a rank which is expressed as a function of the number of individuals dominating it. Assuming $Ndom^t$ to be the number of solutions dominating a certain solution $x$ at a generation $t$, the rank at $t$ of $x$ is given by:

$$rank^t(x) = 1 + Ndom^t \tag{2.11}$$

With such ranking mechanism, non-dominated solutions have a rank of 1 (cf. figure 2.5). The fitness assignment method used in MOGA takes into account the rank of the population member and the average fitness value of the population. The process for computing the fitness values is as follows. Firstly, the population is sorted by rank. Then, a fitness value is assigned to each individual based on an interpolation of the best rank to the worst rank according to some specified function. Finally, individuals

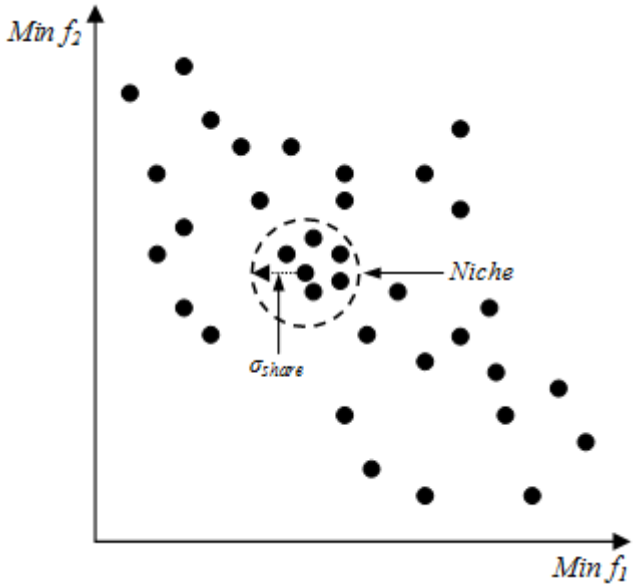



**Figure 2.6 Fitness sharing strategy.**

assigned the same rank receive an averaged fitness value. This ensures that all population members of the same rank are sampled with an identical frequency. This information is used to maintain constant global population fitness with an appropriate amount of selective pressure. Additionally, MOGA implements the concept of fitness sharing (also referred to as crowding or niching) and uses a $\sigma_{share}$ parameter called the niche radius which must be carefully specified. The niching mechanism is applied in the objective space in order to obtain a uniform distribution of the Pareto front approximation. Figure 2.6 illustrates the fitness sharing mechanism. In fact, solutions residing inside the niching radius are penalized in their fitness values.

Although in MOGA fitness assignment is explicitly based on Pareto dominance, solutions having the same rank may not have the same assigned fitness. This may cause an unwanted bias towards a certain zone of the search space. Particularly, MOGA may be sensitive to the geometry of the Pareto front in addition to the density of solutions over the search space. Besides, the fitness sharing mechanism favors solutions with poor ranks over solutions with higher ranks if these latter are more crowded, thereby worsening the converging.

- **NPGA: Niched Pareto Genetic Algorithm**

Horn and Nafpliotis (1994) proposed NPGA which differs from the previously discussed MOEAs in the selection operator. This algorithm uses the binary tournament selection instead of proportionate selection methods used in VEGA and MOGA. During the tournament selection, two solutions *x* and *y* are picked randomly from the

parent population *P*. Then, these two solutions are compared based on Pareto dominance to each individual of a randomly selected subpopulation *T* of size *tdom* (where $tdom << |P|$). If one of the two solutions is non-dominated with respect to all the subpopulation individuals and the other one is dominated by at least one individual, the non-dominated solution is retained. In the cases where neither or both members are dominated by the subpopulation members, a niching mechanism is implemented to select the least crowded solution among *x* and *y*.

NPGA is found to be sensitive to the $\sigma_{share}$ value in addition to the *tdom* one. The numerical results reported in (Horn and Nafpliotis 1994) suggest that *tdom* should be an order of magnitude smaller than the population size. On one hand, if *tdom* is too small, the non-domination check would be so noisy which may not emphasize non-dominated solutions sufficiently. On the other hand, if *tdom* is too large, non-dominated solutions will be well-emphasized but the computational complexity will increase. Additionally, *tdom* depends on the number of objectives to optimize.

- **NSGA: Non-dominated Sorting Genetic Algorithm**

NSGA (Srinivas and Deb 1994) is based on the non-dominated sorting strategy (cf. figure 2.7). This strategy classifies the population members into several fronts. The non-dominated sorting algorithm begins by identifying the non-dominated individuals from all population members. These individuals have the rank of one and are assigned a large dummy fitness value. After that, the first front members are discarded temporary from the population and the non-dominated individuals from the truncated population are identified and assigned the rank of 2 (eventually assigned a dummy fitness value smaller than the one of the first front). This process continues until classifying all population members. The diversity maintenance is achieved in NSGA by applying the fitness sharing front-wise in the decision space (instead of the objective space) in order to degrade the fitness values based on a user-defined niche radius value $\sigma_{share}$. The sharing in each front is achieved by calculating a sharing function value between two individuals *i* and *j* in the same front as follows:

$$Sh_{d_{ij}} = \begin{cases} 1-\left(\dfrac{d_{ij}}{\sigma_{share}}\right)^2 & \text{if } d_{ij} < \sigma_{share} \\ 0 & \text{otherwise} \end{cases} \qquad (2.12)$$

where $d_{ij}$ is the Euclidean distance separating *i* and *j*. After that, a parameter niche count is calculated by adding the above sharing function values for all individuals in the current front. Finally, the shared fitness value of each individual is computed by

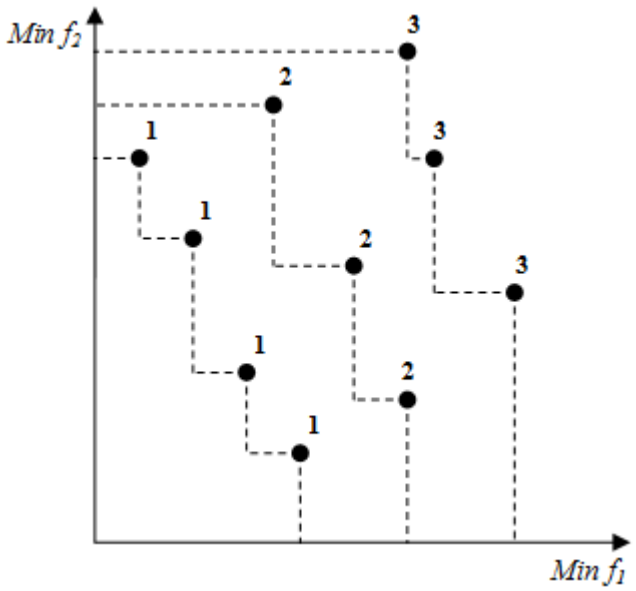



**Figure 2.7 Non-dominated sorting strategy.**

dividing its dummy fitness value by its niche count. The best individuals are always preferred over other solutions, thereby favoring the generation of new individuals near the non-dominated solutions. The fitness sharing mechanism helps the algorithm to distribute the non-dominated solutions along the Pareto front. However, the high sensitivity to the $\sigma_{share}$ parameter yields to a less efficient performance of NSGA.

- **Elitist methods**

Elitism means that elite individuals cannot be excluded from the archive gene pool of the population in favour of worse individuals (Holland 1975). In the following, we review the most representative elitist MOEAs.

  - **SPEA/SPEA2: Strength Pareto Evolutionary Algorithm**

(Zitzler and Thiele 1999) proposed the strength Pareto approach which uses two populations: (1) a main population *P* and (2) an archive population *A* which contains the non-dominated individuals found so far during the evolutionary process. Initially, the population *P* is generated randomly and the archive *A* is empty. Then, *A* is filled with non-dominated members from *P*. After that, solutions from *A* which are dominated by any other member from *A* are deleted. Besides, if the number of externally stored non-dominated solutions exceeds the archive size $|A|$, then *A* is pruned by means of a clustering procedure which will be discussed next. Once all population and archive members are each assigned a fitness value, binary tournament selection with replacement is applied to fulfil the mating pool. After applying genetic operators, a new population *P* is generated. If a stopping condition is met then the evolutionary process

is stopped, else non-dominated vectors from *P* are copied to the archive *A* as usual and the overall process is repeated.

The fitness assignment in SPEA is a two-stage process. First, the non-dominated individuals from the archive *A* are ranked. Then, the population *P* members are evaluated. In fact, every solution *i* from the archive *A* is assigned a strength value $s_i \in [0,1[$ which is proportional to the number of individuals in *P* which are dominated by *i*. The strength $s_i$ is given by:

$$s_i = \frac{nd}{|P|+1} \tag{2.13}$$

where $nd$ denotes the number of individuals in *P* that are covered by *i* and $|P|$ is the main population size. The fitness of population individual $j \in P$ is obtained by summing the strengths of all non-dominated solutions $i \in A$ that dominates *j*. The obtained sum is raised by 1 in order to guarantee that archive members have better performance than *P* members. This fitness is to be minimized and is given by:

$$f_j = 1 + \sum_{i, i \preceq j} s_i \tag{2.14}$$

The clustering mechanism is applied to reduce the size of the archive while keeping its characteristics. The general idea is to partition the archive into *C* groupings (clusters), where $C < |A|$ and all individuals of the same grouping have the same characteristics. The clustering procedure begins by making each element of the initial non-dominated archive a cluster. Following this, two clusters are chosen via a distance measurement to be combined into one cluster. The distance is calculated as average Euclidean distance between pairs of individuals across the clusters. At the completion of the clustering process, the new non-dominated archive consists of the centroid members of each cluster. The authors show favorable results compared to other MOEAs.

In another study (Zitzler et al. 2001) have identified three weaknesses for SPEA. Firstly, for the fitness assignment strategy, individuals that are dominated by the same archive members have identical fitness values. Hence, in the case when the archive contains only a single individual, all population members have the same rank independently of whether they dominate each other or not. Consequently, the selection pressure is decreased substantially and in this particular case SPEA behaves like a random search algorithm. Secondly, for the density estimation, if many individuals of the current generation are Pareto equivalent, none or very little information can be obtained on the basis of the partial order defined by the dominance relation. In this

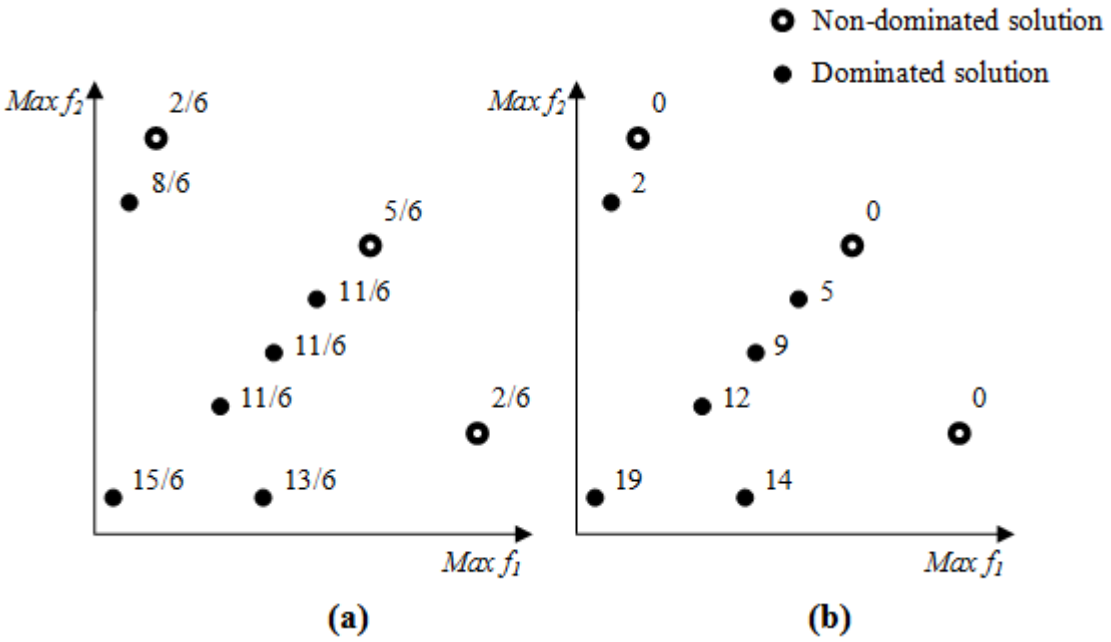



**Figure 2.8 Comparison of fitness assignment mechanisms: (a) SPEA versus (b) SPEA2 (from** (Zitzler et al. 2001)**).**

situation, which is very likely to occur when the number of objectives exceeds two, density information has to be used in order to guide the search more effectively. Clustering makes use of this information, but only with regard to the archive and not to the main population. Thirdly, for the archive truncation strategy, although the clustering mechanism used in SPEA is able to reduce the non-dominated set without destroying its characteristics, it may lose extreme (outer) solutions. However, these solutions should be kept in the archive in order to obtain a good spread of non-dominated solutions. In response to the above mentioned SPEA weaknesses, Zitzler et al. (2001) have proposed an improved version of SPEA, called SPEA2. In contrast to SPEA, SPEA2 uses a fine-grained fitness assignment strategy which incorporates density information. Furthermore, the archive size is fixed, i.e., whenever the number of non-dominated individuals is less than the predefined archive size, the archive is filled up by dominated individuals; with SPEA, the archive size may vary over time. In addition, the clustering technique, which is invoked when the non-dominated front exceeds the archive limit, has been replaced by an alternative truncation method which has similar features but preserves boundary solutions. Finally, another difference to SPEA is that in SPEA2 only members of the archive participate in the mating selection process.

The SPEA 2 fitness assignment for a certain solution *i* takes into account the number of individuals dominating *i* in addition to the number of individuals dominated by *i*. Each

solution $i$ from the population $P$ and the archive $A$ is assigned a strength value $s_i$ representing the number of individuals dominated by $i$:

$$s_i = \left| j \middle| j \in P \cup A \wedge i \preceq j \right| \tag{2.15}$$

After that, the raw fitness $R_i$ is computed as flows:

$$R_i = \sum_{j \in P+A,\, j \preceq i} s_j \tag{2.16}$$

This raw fitness is determined by the strengths of its dominators in both archive and population, as opposed to SPEA where only archive members are considered in this context. It is important to note that fitness is to be minimized here, i.e., $R_i = 0$ corresponds to a non-dominated individual, while a high $R_i$ value means that $i$ is dominated by many individuals (which in turn dominate many individuals). This scheme is illustrated in figure 2.8(b).

The raw fitness assignment strategy supplies a sort of niching based on the Pareto dominance concept. However, this strategy becomes inefficient when most individuals are non-dominated with each other. For this reason, additional density information is incorporated to discriminate between individuals having identical raw fitness values. The density estimation technique used in SPEA2 is an adaptation of the $k^{th}$ nearest neighbor method where the density at any point is a decreasing function of the distance to the $k^{th}$ nearest points. The density estimate corresponds to the inverse of the distance to the $k^{th}$ nearest neighbor. In fact, for each individual $i$, the distances in objective space to all individuals $j$ from $P \cup A$ are computed then stored in a list in an increasing order. After that, the $k^{th}$ nearest neighbor gives the sought distance denoted by $\sigma_i^k$. The $k$ parameter value is usually set to $\sqrt{|P| + |A|}$. The density $D_i$ of solution $i$ is:

$$Di = \frac{1}{\sigma_i^k + 2} \tag{2.17}$$

In the denominator, two is added to ensure that its value is greater than zero and that $D_i < 1$. Finally, the fitness of a certain solution $i$ is obtained by summing the raw fitness and the density information as follows:

$$F_i = R_i + D_i \tag{2.18}$$

The SPEA2 environmental selection mechanism differs from SPEA one by preserving the boundary solutions and by the fact that the number of stored external solutions is constant over time.

- **NSGA-II: Non-dominated sorting Genetic Algorithm II**

NSGA-II is the improved version of NSGA (Deb et al. 2000; 2002a). NSGA-II is one of the most cited MOEAs. The most prominent features of NSGA-II are its low computational complexity, elitist approach and a method for diversity that does not need additional parameters. The general principle of NSGA-II is as follows. The NSGA-II algorithm begins by creating an offspring population $Q_0$ by applying genetic operators to a randomly generated parent population $P_0$. From the first generation award, the basic iteration of NSGA-II is different. First, the two populations $P_t$ and $Q_t$ are combined to form a population $R_t$ of size $2N$ ($|P_t| = |Q_t| = N$). Second, a non-dominated sorting is performed to classify the entire population $R_t$. Once, the non-dominated sorting is over, the population $R_t$ becomes subdivided in several categories in the same manner of NSGA. After that, the new parent population $P_{t+1}$ is filled with individuals of the best non-dominated fronts, one at a time. Since the overall population size is $2N$, not all fronts may be accommodated in $N$ slots available in the new population $P_{t+1}$. When the last allowed front is being considered, it may contain more solutions then the remaining available slots in $P_{t+1}$. Instead of discarding arbitrary some elements from the last front, NSGA-II uses a niching strategy to choose individuals from the last front which reside in the least crowded regions in this front. In fact, for each ranking level, a crowding distance is estimated by calculating the sum of the Euclidean distances between the two neighboring solutions from either side of the solution along each of the objectives as demonstrated by figure 2.9. In order to preserve boundary solutions, these latter are each assigned an infinite crowding distance. The crowding distance assignment procedure can be summarized by the three following steps:

- **Step 1:** For each solution $i$ from the considered front $F$, initialize its crowding distance $CD_i$ to zero: $CD_i \leftarrow 0$;
- **Step 2:** For each objective function, sort the front members in a decreasing order of $f_m$, and find the sorted indices vector: $I^m = sort(f_m, >)$;
- **Step 3:** For $m = 1,...,M$, assign an infinite crowing distance to extreme solutions $(CD_{I_1} = CD_{I_{|F|}} = \infty)$ and for the other solutions $j = 2,...,|F|-1$, assign:

$$CD_{I_j^m} = CD_{I_j^m} + \frac{f_m^{I_{j+1}^m} - f_m^{I_{j-1}^m}}{f_m^{\max} - f_m^{\min}} \tag{2.19}$$

where $I_j^m$ corresponds to the index of the $j^{th}$ member in the list sorted based on the $m^{th}$ objective function.

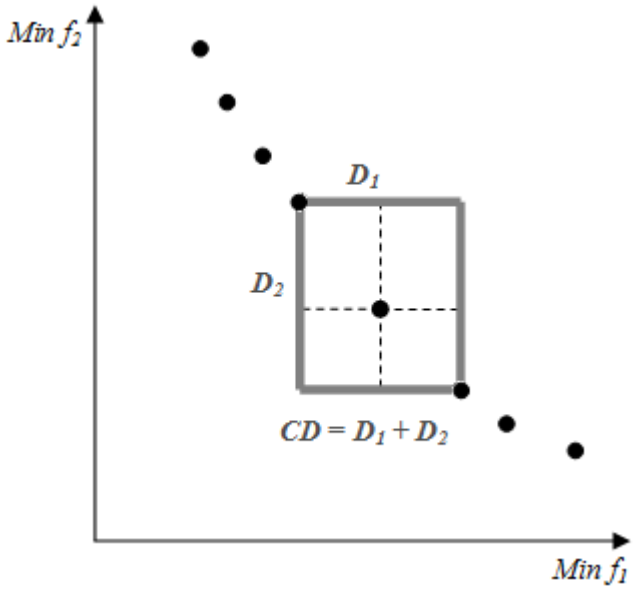



**Figure 2.9 Crowding distance.**

NSGA-II is demonstrated to be one of the most competitive MOEAs through the specialized literature. The main weakness of NSGA-II was reported in (Deb 2001). In fact, when the cardinality of the first front from the combined population $R_t$ exceeds the population size $|P|$, some closely packed Pareto optimal solutions may give their places to some non-dominated yet non Pareto optimal solutions since the replacement becomes based only on the crowding distance criterion.

- **PAES/PESA: Pareto Archived Evolutionary Strategy/Pareto Envelope-based Selection Algorithm**

(Knowles and Corne 1999; 2000) proposed a (1+1)-Evolutionary Strategy ((1+1)-ES), named PAES, to approximate the whole Pareto front. This work was motivated by the success of (1+1)-ES in resolving mono-objective problems. For this reason the authors have adapted such search method for the multi-objective case. PAES begins by producing a child $c_0$ from a randomly generated parent $p_0$. In each generation $t$, non-dominated solutions found are stored in an archive with a pre-specified size. The two individuals $p_t$ and $c_t$ are firstly compared. If one solution dominates the other, the dominated individual is discarded and the dominant one is retained as parent for the next generation. In the case where $p_t$ and $c_t$ are non-dominated, the new candidate solution is compared with a reference population of previously archived non-dominated solutions, i.e., archive members. If comparison to the population in the archive fails to favor one solution over the other, the tie is split to favor the solution which resides in the least crowded region of the search space. The archive has a user-specified maximum size which reflects the desired number of final solutions. Each child $c_t$ which

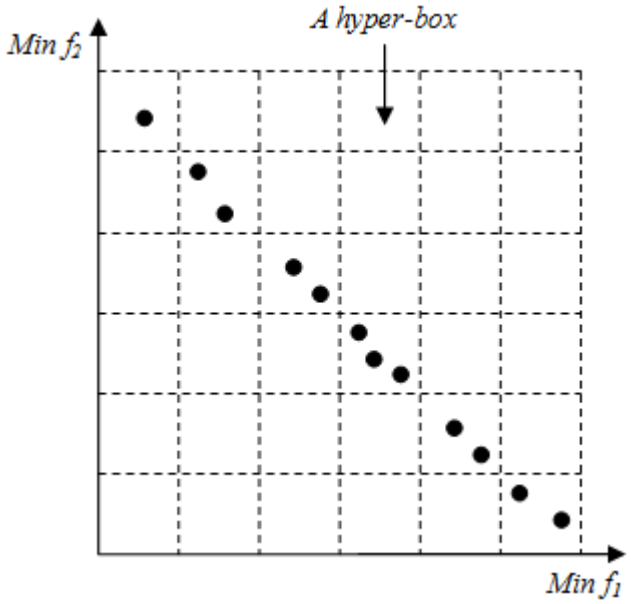



**Figure 2.10 PAES hyper-gridding system with *d* = 6.**

is not dominated by its parent $p_t$ is compared with each member of the archive. Candidates which dominate the archive members are always accepted (as parents) and archived. Candidates which are dominated by the archive members are always rejected, while those which are non-dominated are accepted and/or archived based on the degree of crowding in their grid location. The major feature of PAES is its strategy for promoting diversity in the approximation set. PAES uses an adaptive hyper-gridding system in the objective space to divide it into *d* non-overlapping hyper-boxes. The belonging of a certain solution to a certain region in the hyper-box is determined by the objectives' values which define the solution's coordinates. In the case where an offspring solution is non-dominated with respect to the archive members, a crowding measure based on the number of solutions residing in a certain hyper-box is applied to determine whether the offspring solution is accepted or not.

The major advantage of this diversity maintenance technique is that it does not require any additional parameters such as the niche size parameter $\sigma_{share}$. However, the main crux of PAES is the sensitivity of the performance of such algorithm to the *d* parameter of the hyper-gridding system (cf. figure 2.10).

The same authors (Corne and Knowles 2000) have proposed PESA which is a modified version of PAES. PESA has the same archiving and diversity preserving mechanisms of PAES. In PESA, like SPEA2, only archive members participate in genetic operations. PESA begins by randomly generating a small internal population *IP*. PESA uses also a large external population *EP* which is initially empty. After that, the archive *EP* is

updated with elite solutions in the same manner as PAES. If the stopping criterion is met then the algorithm returns *EP*, else *IP* is fulfilled with new individuals by the following operations. With probability $p_c$, two parents are selected from *EP*. A single child is subsequently created by crossover. This child is then mutated. With probability $(1-p_c)$, a selected parent from *EP* is mutated. After that, the archive *EP* is updated and the overall process is repeated.

As PAES, PESA necessitates the tuning of the archive size and the *d* parameter of the gridding system. We note that the number of hyper-boxes changes exponentially with the modification of *d* value which influences the final population distribution. An improved version of PESA, called PESA-II, was proposed by Corne et al. (2001) where selection is region-based and the subject of selection is now a hyper-box not only an individual (i.e., first selecting a hyper-box, then an individual is chosen from the selected hyper-box). The motivation behind PESA is to reduce the computational cost of Pareto ranking.

- **IBEA: Indicator-Based Evolutionary Algorithm**

(Zitzler and Künzli 2004) proposed a MOEA where selection is based on solution contribution to a certain quality indicator. Indicator-based MOEAs can, therefore, be seen as a third generation of MOEAs. IBEA begins by randomly generating a population *P*. After that, for each solution *i* from *P*, the algorithm computes the fitness of *i* corresponding to the loss in quality if *i* is removed from the population *P*. The solution with the lowest fitness is removed from the population and then the population members' fitness values are recomputed since the population is truncated. This selection strategy is used in creating the mating pool and in environmental selection. The main crux of IBEA is its sensitivity to the $\kappa$ parameter which is used to scale the fitness function values since the algorithm performance largely depends on this parameter which is reported to depend of the considered MOP. Another indicator-based selection algorithm is the *S* Metric Selection-based Evolutionary Multi-Objective Algorithm (SMS-EMOA) (Beume et al. 2007) which combines non-dominated sorting with indicator-based selection mechanism. IBEAs can be seen as the last generation of MOEAs. The main critical point in this type of algorithms is the important required computational effort for computing the quality indicator values for a certain non-dominated solution set.

## 2.4 Performance assessment of MOEAs

### 2.4.1 Test functions

**Table 2.1 Bi-objective ZDT test problems' characteristics.**

| Name | Features |
|---|---|
| ZDT1 | The Pareto front is convex. |
| ZDT2 | The Pareto front is concave. |
| ZDT3 | The Pareto front is formed by several disjoint convex parts. |
| ZDT4 | There are $21^9$ local fronts. |
| ZDT5 | The Pareto front is convex. ZDT5 is a discrete problem with a deceptive landscape. |
| ZDT6 | The Pareto front is concave. This problem is characterized by the non-uniformity not only of the search space but also of the solution distribution along the Pareto front. |

**Table 2.2 Scalable DTLZ test problems' characteristics.**

| Name | Features |
|---|---|
| DTLZ1 | The Pareto front is linear (Hyper-plane). There are $(11^k - 1)$ local optimal fronts where *k* is a user-specified parameter. |
| DTLZ2 | For $M > 3$, the Pareto optimal solutions lie inside the first quadrant of the unit sphere in a three-objective plot with $f_M$ as one of the axes. |
| DTLZ3 | There are $(3^k - 1)$ local fronts that are parallel to the global Pareto front where *k* is a user-specified parameter. |
| DTLZ4 | The Pareto optimal solutions are non-uniformly distributed along the Pareto front. |
| DTLZ5 | The front is a curve and the Pareto optimal solutions are non-uniformly distributed along the Pareto front. |
| DTLZ6 | The front is a curve and the solution density gets thinner towards the Pareto front. |
| DTLZ7 | The Pareto front is formed by $2^{M-1}$ disjoint regions in the objective space. |
| DTLZ8 | The Pareto front is a combination of a straight line and a hyper-plane. The straight line is the intersection of the first (*M-1*) constraints with $f_1 = f_2 = ... = f_{M-1}$ and the hyper-plane is represented by another constraint $g_M$. |
| DTLZ9 | The Pareto front is a curve with $f_1 = f_2 = ... = f_{M-1}$. The solution density gets thinner towards the Pareto front. |

Several test functions are proposed to challenge MOEA capabilities in approximating the Pareto front. The most cited test function suites are: (1) the bi-objective ZDT (Zitzler-Deb-Thiele) suite (Zitzler et al. 2000) and (2) the scalable DTLZ (Deb-Thiele-Laumans-Zitzler) suite (Deb et al. 2002b) where the Pareto optimal front can be

determined analytically (cf. appendices A and B). Such test functions encapsulate several characteristics such as non-convexity, multimodality, non-uniformity of the search space and discontinuity which cause difficulties to a MOEA. These test functions do not reflect necessarily the main features of real world MOPs. It is true that some of these functions contain important characteristics that make them particularly difficult to solve. Thus, if a MOEA can resolve such test functions, it should also be able to tackle real world MOPs; although this is not necessarily true. Tables 2.1 and 2.2 present the ZDT and DTLZ test functions' characteristics respectively. We notice that for DTLZ test problems, the parameters can be modified in order to increase or decrease the problem's difficulties (e.g., modifying the number of local optimal Pareto fronts).

### 2.4.2 Performance indicators

When evaluating the performance of a MOEA, there are two main goals to pursue: (1) closeness of the provided non-dominated solution set to the Pareto optimal front and (2) diversity of the obtained solution set (with a good distribution) along the Pareto optimal front. Several performance measures are proposed in the EMO literature to evaluate one or both of these goals (Zitzler et al. 2003). Table 2.3 presents a classification of selected representative performance measures. The classification criteria are the following:

- *unary* which indicates if it is a unary performance indicator (i.e., performance measure which assigns a single value to each non-dominated solution set);
- *binary* which indicates if it is a binary performance indicator (i.e., performance measure which assigns a single value to a pair of non-dominated solution sets);
- *convergence* which indicates that the performance indicator assigns a single value corresponding to the convergence of the non-dominated solution set;
- *diversity* which indicates that the performance indicator assigns a single value corresponding to the diversity of the non-dominated solution set;
- *reqPFtrue* which indicates if the performance measure requires the true Pareto optimal front $PF_{true}$ to assign a single value to the non-dominated solution set;
- *best value* which indicates the best value that can be obtained from the performance indicator;
- *Pareto compliant* which indicates whether the performance measure is Pareto dominance compliant. Before defining the notion of Pareto dominance compliance, we give the definitions of compatibility and completeness. The definitions are derived from the study of Zitzler et al. (2003):

**Definition 2.8: Compatibility and Completeness**

Assuming $W$ and $Z$ two approximation sets, a quality indicator $I : \Omega \rightarrow \Re$ (assuming

**Table 2.3 Main features of performance indicators.**

| Performance indicators | unary | binary | convergence | diversity | reqPFtrue | best value | Pareto compliant |
|---|---|---|---|---|---|---|---|
| ER | X | | X | | X | 0 | X |
| SC | | X | X | | | 1 | X |
| $I_{\varepsilon^+}$ | | X | X | | | - | X |
| GD | X | | X | | X | 0 | |
| IGD | X | | X | | X | 0 | |
| Δ | X | | | X | | 0 | |
| HV | X | | X | X | | 1 | X |
| S | X | | | X | | 0 | |
| $\chi^2$–like deviation | X | | | X | X | 0 | |

higher values of the indicator mean better performance) is said to be *compatible* with the Pareto dominance relation if and only if:

$$I(W) > I(Z) \Rightarrow W \preceq Z \tag{2.20}$$

The quality indicator *I* is said to be *complete* if and only if:

$$W \preceq Z \Rightarrow I(W) > I(Z) \tag{2.21}$$

**Definition 2.9: Compliance**

A quality indicator *I* is said to be Pareto dominance *compliant* if *I* is both compatible and complete with the Pareto dominance relation.

***Error Ratio* (*ER*):** This indicator is proposed by Van Veldhuizen and Lamont (2000). It corresponds to the ratio of the number of solutions that are not members of the true Pareto optimal front $PF_{true}$ by the cardinality of the obtained solution set. Mathematically, *ER* is expressed as follows:

$$ER = \frac{\sum_{i=1}^{N} e_i}{N} \tag{2.22}$$

where *N* is the number of non-dominated solutions provided by the MOEA and $e_i = 1$ if solution *i* is dominated by any member from $PF_{true}$ and $e_i = 0$ otherwise. $ER = 1$ means that no solution belongs to the true front $PF_{true}$. and $ER = 0$ when all solutions are in the true front.

***Set Coverage* (*SC*):** This indicator can be termed relative coverage of two solution sets (Zitzler et al. 2000). *SC* is defined as the mapping of the pair (*W*, *Z*) to the interval [0,1]

as follows:

$$SC(W,Z) = \frac{\left|\{z \in Z; \exists w \in W : z \preceq w\}\right|}{|Z|} \tag{2.23}$$

$SC(W,Z)$ expresses the percentage of solutions from *Z* that dominates solutions in *W*. $SC(W,Z)=1$ means that each solution in *Z* dominates at least one solution from *W*; while $SC(W,Z)=0$ means the opposite (i.e., there is no solution from *Z* dominating solutions from *W*).

***Binary additive epsilon indicator*** $(I_{\varepsilon^+})$**:** This metric takes a pair of non-dominated solution sets *W* and *Z* as inputs and returns a pair of numbers as outputs $(I_W, I_Z)$ such that (Zitzler et al. 2003):

$$I_W = I_{\varepsilon+}(W,Z) = \inf_{\varepsilon \in \Re} \{\forall z \in Z, \exists w \in W : w \preceq_{\varepsilon+} z\} \tag{2.24}$$

$$I_Z = I_{\varepsilon+}(Z,W) = \inf_{\varepsilon \in \Re} \{\forall w \in W, \exists z \in Z : z \preceq_{\varepsilon+} w\} \tag{2.25}$$

$I_{\varepsilon^+}(W,Z)$ expresses the minimum quantity $\varepsilon$ by which each solution from *W* must be translated in the objective space so that each solution from *Z* becomes dominated by (or equal to) at least one member from *W*. A pair of numbers $(I_W \leq 0, I_Z > 0)$ indicates that *W* is strictly better than *Z*, while a pair of numbers $(I_W > 0, I_Z > 0)$ means that *W* and *Z* are incomparable. Nevertheless, if $I_W$ is less than $I_Z$, then in a weaker sense, we can say that *W* is better than *Z* because the minimum $\varepsilon$ value needed so that *W* ε-dominates *Z* is smaller than the $\varepsilon$ value needed so that *Z* ε-dominates *W*.

***Generational distance*** **(*GD*):** This indicator estimates how far are the elements in the Pareto front produced by the MOEA from those in the true Pareto front of the problem (i.e., $PF_{true}$) (Van Veldhuizen and Lamont 2000). It is given by the following equation:

$$GD = \frac{\sqrt{\sum_{i=1}^{N} d_i^2}}{|PF_{true}|} \tag{2.26}$$

where *N* is number of non-dominated solutions provided by the MOEA and $d_i$ is the distance between each of these solutions to its nearest member from $PF_{true}$. A variant of this indicator is the *Inverted Generational Distance* (*IGD*) in which a reference true Pareto front is used and its elements are compared with respect to the approximation produced by the MOEA.

***Spread* (Δ):** The metric Δ measures the deviation among consecutives solutions in the Pareto front *PF* furnished by the MOEA (Deb 2001). Analytically, Δ is stated as follows:

$$\Delta = \sum_{i=1}^{|PF|} \frac{\left| dist_i - \overline{dist} \right|}{|PF|} \tag{2.27}$$

where $dist_i$ is the Euclidean distance between two consecutive solutions in *PF* and $\overline{dist}$ is the average of these distances. In order to ensure that this calculation takes into account the spread of solutions in the entire region of the true front, the boundary solutions in the non-dominated front are included. For a perfect distribution, $\Delta = 0$ which means that $dist_i$ is constant for all *i*.

***HyperVolume* (*HV*):** This indicator, called also *S metric*, estimates the hypervolume of the portion of the objective space which is dominated by an approximation set (Zitzler and Thiele 1999). The larger *HV* value is, the better the result is. This metric assesses both convergence and diversity. The *HV* indicator can be expressed as follows:

$$HV = \bigcup_i vol_i \,\middle|\, i \in PF \tag{2.28}$$

where $vol_i$ corresponds to hyperarea bounded by a pre-specified reference point and a solution *i*. The *HV* metric is *compatible* and *complete* with the Pareto dominance relation; thereby *HV* is said to be Pareto *compliant* which is an important feature for this indicator.

***Spacing* (*S*):** This metric assesses the solution distribution along the Pareto front and it is given by:

$$S = \sqrt{\frac{1}{|PF| - 1} \sum_{i=1}^{|PF|} \left( dis_i - \overline{dis} \right)^2} \tag{2.29}$$

where $dis_i = \min_{j \in PF} \sum_{m=1}^{k} \left| f_m^i - f_m^j \right|$ and $\overline{dis}$ is the mean of these distances. The distance measure is the minimum value of the sum of the absolute differences in objective function values between solution *i* and any other solution in the Pareto optimal set. $S = 0$ means that all solutions are equally distributed along the Pareto front.

***Chi-square-like deviation measure* ($\chi^2$ *– like deviation*):** Proposed by Srinivas and Deb (1994), this indicator evaluates the diversity of the obtained solution set *PF*. *PF* solutions are compared with respect to a uniformly distributed set of $PF_{true}$ called *F*.

For each $i \in \{1,2,...,|F|\}$, we denote by $n_i$ the number of solutions in *PF* whose distance from *i* is less than a user-specified quantity $\omega$. Then, the measure is computed as follows:

$$\chi = \sqrt{\sum_{i=1}^{|F|+1} \left( \frac{n_i - \overline{n_i}}{\sigma_i} \right)^2} \tag{2.30}$$

The ideal distribution is achieved when all the neighborhoods of points in *F* have the same cardinality, i.e., if for each solution *i* in *F* there is $\overline{n_i} = \frac{|PF|}{|F|}$ points whose distance from *i* is less than $\omega$, then $\chi = 0$. The variance $\sigma_i^2$ is proposed to be $\sigma_i^2 = \overline{n_i}\left(1 - \frac{\overline{n_i}}{|PF|}\right)$ for all $i \in \{1,2,...,|F|\}$. The lower the $\chi$ value is, the better the distribution is.

## 2.5 Conclusion

Through this chapter, we have provided a comprehensive review of the EMO research field. We classified MOEAs based on two main criteria: (1) the use of the Pareto dominance as a selection criterion and (2) the elitism. Figure 2.11 illustrates a cartography of the different discussed MOEAs. Non-Elitist approaches are seen as a first generation of MOEAs while the second generation corresponds to the elitist methods. The use of a performance indicator as a selection criterion can be considered as the selection mechanism of the third generation of MOEAs and several studies are recently conducted in this direction. We have presented how MOEA output can be assessed by means of quality metrics and difficult test functions with predefined Pareto optimal fronts each having some geometrical features presenting challenges to every search method. As discussed through this chapter, most of the described MOEAs have shown their effectiveness and efficiency in ensuring not only convergence towards the Pareto front but also diversity between the final obtained solutions. However, this fact does not resolve the problem of decision making since the DM has to choose a single solution from a huge set of non-dominated solutions. The next chapter is dedicated to review the incorporation of DM's preferences in MOEAs since in reality the DM is not interested in discovering the whole Pareto front but rather finding only the portion of the front that matches at most his/her preferences.

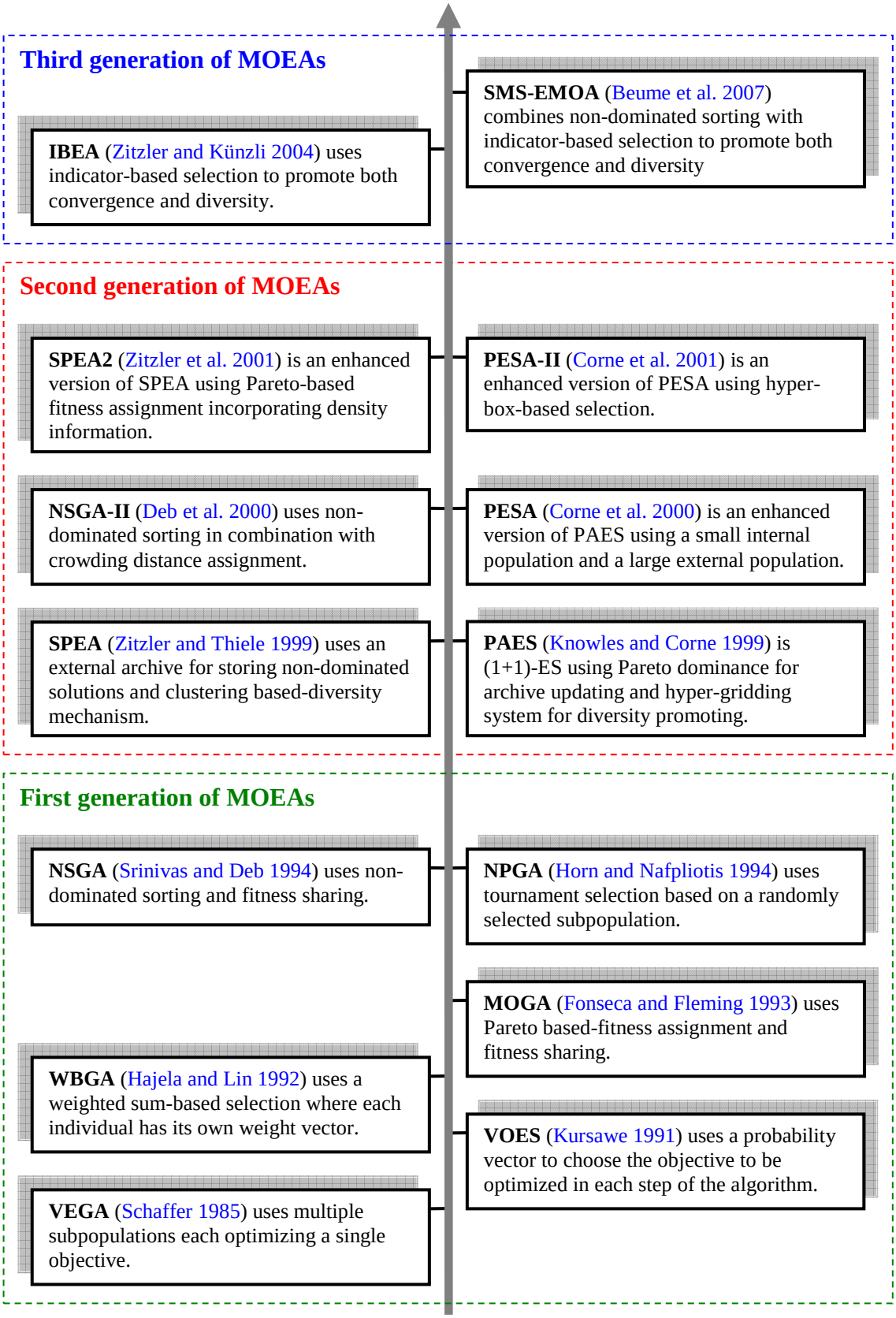



**Figure 2.11 MOEA cartopgraphy.**

# Chapter 3

# Explicit Preference-based Evolutionary Multi-objective Optimization

## 3.1 Introduction

Over the two past decades, MOEAs have demonstrated their effectiveness and efficiency in providing well-converged and well-diversified approximations of the Pareto front. Recently, there has been an increased emphasis in addressing the decision making task by injecting DM's preference information in the evolutionary process. This chapter surveys existing preference-based MOEAs (Bechikh et al. 2012a). In such type of algorithms, the DM can provide his/her preferences before (a priori), after (a posteriori) or during the MOEA run (interactively). These preferences are used to guide the search towards the preferred part(s) of the Pareto optimal front, i.e., the ROI(s). Each solution belonging to a ROI is considered to be a preferred and satisfying solution for the DM. The DM's preference information can be expressed in several ways. Most of these ways are issued from the classical Multi-Criteria Decision Making (MCDM) literature (Miettinen 1999). In the following, we cite the commonly used preference information structures in the EMO community:

- *Weights*: Each objective is assigned a weighting coefficient expressing its importance. The larger the weight is, the more important the objective is.
- *Solution ranking*: The DM is provided with a sample of solutions (a subset of the current MOEA's population) and is invited to perform pairwise comparisons between pairs of solutions in order to rank the sample's solutions where incomparability and indifference may exist between the solutions to rank.
- *Objective ranking*: Pairwise comparisons between pairs of objectives are performed in order to rank the MOP's objectives where incomparability and indifference may exist between some objectives.
- *Reference point* (also called a *goal* or an *aspiration level vector*): The DM supplies, for each objective, the desired level that he/she wishes to achieve. This desired level is called *aspiration level*.

- *Reservation point* (also called *a reservation level vector*): The DM supplies, for each objective, the accepted level that he/she wishes to reach. This accepted level is called *reservation level*.
- *Trade-off between objectives*: The DM specifies that the gain of one unit in one objective is worth a degradation in some others and vice versa.
- *Outranking thresholds*: The DM specifies the necessary thresholds to design a fuzzy predicate modelling the truth degree of the predicate "solution *x* is at least as good as solution *y*".
- *Desirability thresholds*: The DM supplies: (1) an absolutely satisfying objective value and (2) a marginally infeasible objective value. These thresholds represent the parameters that define the Desirability Functions (DFs).

The next section provides a classification of preference-based MOEAs based on the structure of the DM's preference information. We focus on the way the preferences are supplied and the mechanism adopted to incorporate these preferences so that the population is guided towards the ROI(s).

## 3.2 Preference-based MOEAs

### 3.2.1 Weight-based approaches

♦ **Deb (**1999**) work: the biased sharing-based approach**

In this work, the author incorporated the relative importance of each objective in the form of weight. In fact, he modified the Euclidean distance computation in the sharing mechanism of NSGA. Originally, the distance between two decision variable vectors *x* and *y* is computed as follows:

$$d(x,y)=\sqrt{\sum_{m=1}^{M}\left(\frac{f_m(x)-f_m(y)}{f_m^{\max}-f_m^{\min}}\right)^2} \tag{3.1}$$

The quantity (3.1) is modified by incorporating weighting coefficients into it. The obtained distance metric is called the *weighted Euclidean distance* and is expressed by:

$$d(x,y)=\sqrt{\sum_{m=1}^{M}w'_m\left(\frac{f_m(x)-f_m(y)}{f_m^{\max}-f_m^{\min}}\right)^2} \tag{3.2}$$

where $w'_m$ is the normalized weighting coefficient and is expressed by:

$$w'_m=\frac{(1-w_m)}{\max_{m=1}^{M}(1-w_m)} \tag{3.3}$$

where $w_m$ is the user-specified weight assigned to the $m^{th}$ objective expressing its importance degree. We note that $w_m \in [0,1]$ and $\sum_{m=1}^{M} w_m = 1$. The sharing mechanism (cf. equation (2.12)) can then be used in order to bias the Pareto optimal solution distribution towards the preferred part of the front. However, this approach was assessed only on two bi-objective problems having convex Pareto fronts with very restricted weight sets that are (0.9, 0.1) and (0.1, 0.9). Hence, further experiments with higher dimension problems and diversified weight sets are required for validation.

♦ **Branke and Deb (**2004**) work: the biased crowding-based approach**

The authors modified the crowding distance calculation in NSGA-II in order to focus the search on the preferred part of the front. For an objective vector *u* from a particular front, a biased crowding distance *D*(*u*) is defined as follows. Let $\eta$ be a DM-specified direction vector indicating the most probable or central linearly weighted utility function and let $\theta$ be a parameter controlling the bias intensity, then:

$$D(u) = d(u)\left(\frac{d'(u)}{d(u)}\right)^{\theta} \tag{3.4}$$

where $d(u)$ and $d'(u)$ are, respectively, the original crowding distance and the crowding distance computed based on the locations of the individuals projected onto the (hyper-)plane with $\eta$ as a direction vector. Figure 1 illustrates this concept. In fact, for a solution from the front more or less parallel to the projected plane (such as solution a), the original crowding distance *d*(a) and the projected crowding distance *d′*(a) are more or less the same, thereby making the ratio *d′*(a)/*d*(a) close to one. Consequently, according to equation (3.4), solution a will have a biased crowding distance *D*(a) almost the same as that in the original objective space, i.e., *d*(a). Contrariwise, for a solution having a large difference in slope on the Pareto optimal front where the tangent has an orientation significantly different from the chosen plane (such as solution b), the projected crowding distance *d′*(b) is much smaller than the original crowding distance *d*(b), thereby making the ratio *d′*(b)/*d*(b) so smaller than one. For such a solution, the biased crowding distance will be a small quantity which means that solution b is assumed to be artificially crowded. Figure 3.1 shows also the biased crowding distance values for all non-dominated points and how would they typically be distributed for a certain front and a chosen plane. Solutions with large crowding distance are preferred which allows solutions situated near the tangent point to survive. The parameter $\theta$ controls the extent of the obtained solutions. The larger $\theta$ is, the smaller the extent is. The main advantages of this approach are: (1) its scalability with

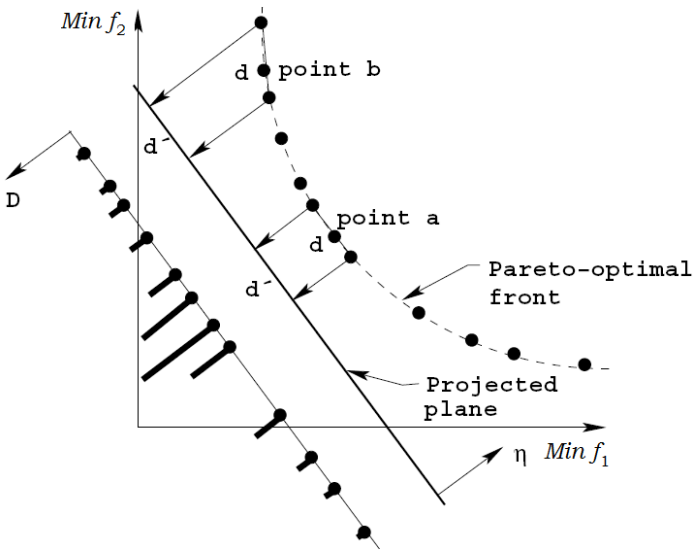



**Figure 3.1 Illustration of the biased crowding based-approach for the bi-objective case (from** (Branke and Deb 2004)**).**

the number of objectives and (2) its insensibility to the non-convexity of the Pareto optimal front. However, the approach was shown to be inferior to the G-MOEA (Branke et al. 2001) in terms of convergence.

♦ **Zitzler et al. (**2007**) work: the weighted hypervolume-based approach**

The hypervolume indicator is a performance measure that computes the surface of the objective space dominated by a solution set and bounded by a reference point (Zitzler et al. 2003). The main feature of this performance measure is its Pareto compliance, i.e., it does not contradict the order induced by the Pareto dominance relation (Zitzler and Thiele 1999). In Zitzler et al. (2007), the authors proposed a weighted version of the hypervolume metric in order to guide the search based on the DM's preferences expressed by: (1) a weighting coefficient vector or (2) a reference point. Three different weighting schemes were proposed for the bi-objective case: (1) a weight distribution which favors extreme solutions, (2) a weight distribution which favors one objective over the other (but still keeping the best solution with respect to the less important objective), and (3) a weight distribution based on a reference point, which generates a ridge-like function through a reference point parallel to the diagonal. In the following, we give the definitions of the hypervolume measure followed by the weighted version of this indicator. The classical definitions of the hypervolume indicator are based on volumes of polytopes (Zitzler and Thiele 1999) or hypercubes (Fleischer 2003) and assume that Pareto dominance is the underlying preference relation. Here, we give a generalized definition based on attainment functions that allows considering arbitrary

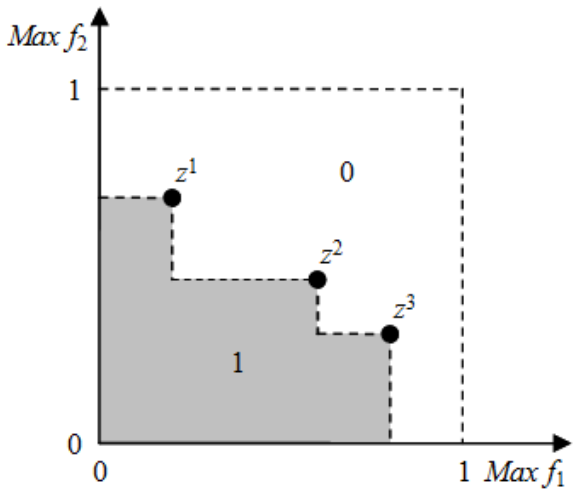



**Figure 3.2 Illustration of the attainment function $\alpha_A$ for $A = \{z^1, z^2, z^3\}$ for the bi-objective maximization case.**

dominance relations. The attainment function (da Fonseca et al. 2003) gives, for each objective vector the probability that it is dominated by the outcome of a particular multi-objective optimizer. As only single sets are considered here, we can take a slightly simplified definition of the attainment function as follows:

**Definition 3.1: Attainment function**

Assuming $A$ to be an objective vector set and $z$ to be an objective vector, the attainment function $\alpha_A(z):[0,1]^M \rightarrow \{0,1\}$ for $A$ is defined as:

$$\alpha_A(z) = \begin{cases} 1 & \text{if } A \succeq \{z\} \\ 0 & \text{else} \end{cases} \tag{3.5}$$

This definition is illustrated in figure 3.2. The concept of attainment function can be used to give a formal definition of the hypervolume indicator. In fact, this latter is defined as the volume of the objective space enclosed by the attainment function and the axes.

**Definition 3.2: Hypervolume indicator**

The hypervolume indicator $I_H^*$ with the reference point (0, ..., 0) could be formulated via the attainment function as:

$$I_H^*(A) = \int_{(0,\ldots,0)}^{(1,\ldots,1)} \alpha_A(z)dz \tag{3.6}$$

The attainment function, the integration over which gives the hypervolume for a given set $A$, is a binary function such that all weakly dominated objective vectors are assigned 1, while the remaining objective vectors are assigned 0. That means all weakly dominated objective vectors have the same weight and contribute equally to the overall

indicator value. The main idea behind the weighted hypervolume approach is to give different weights to different regions in the objective space. This can be achieved by defining a weight distribution over the objective space such that the value that a particular weakly dominated objective vector contributes to the overall indicator value can be any strictly positive real value. To this end, the authors introduced a weight distribution function $w: Z \rightarrow \Re^{+}$. The hypervolume is calculated as the integral over the product of the weight distribution function and the attainment function:

$$I_H^w(A) = \int_{(0,\ldots,0)}^{(1,\ldots,1)} w(z) \cdot \alpha_A(z)\, dz \tag{3.7}$$

The weighted hypervolume is integrated in IBEA (cf. section 2.3.2) and the resulting algorithm has shown its ability to drive the search as expected. However, as noted by the authors, this approach is restricted to the bi-objective case. Moreover, there is no control over the ROI spread. It is worth noting that, in a more recent study (Auger et al. 2009), this work was extended for the *M*-objective case by defining general indicator classes for an arbitrary number of objectives. Furthermore, this extension enables the DM to control the ROI breadth. The main disadvantage of this more recent approach is that, for the case where the DM would like to find a ROI near his/her expressed reference point, the obtained solution distribution highly depends of the position of this preference point.

### 3.2.2 Solution ranking-based approaches

- **Greenwood et al. (**1997**) work**

In this study, the authors proposed an imprecisely specified multi-attribute utility theory-based weighted sum approach where the ranking of objectives is implicitly derived from the ranking of some candidate solutions. The imprecisely specified weighting coefficients are characterized by a set of constraints describing preferences as revealed in pairwise comparisons of the candidate solutions. The used utility function is called imprecise because weights do not have specific values but they are constrained by the DM's preferences. A minimization of the difference in the weighted sums of a pair of solutions, subject to the pre-determined constraints, is performed in the fitness computation. This linear optimization is performed for every solution pair in the archive and in the population to obtain the solution fitness values. Assuming $u$ and $v$ to be two normalized objective vectors (i.e., mapped to the interval [0,1]) where the DM prefers $u$ to $v$, we obtain:

$$u \preceq_{\text{preferred}} v \;\Rightarrow\; \sum_{m=1}^{M} w_m (u_m - v_m) \leq 0 \tag{3.8}$$

The expression (3.8) defines a constraint for the objective weights. When several solution pairs are ranked by the DM, a series of such constraints are defined. These constraints confine the objective weighting coefficients to a subspace $W \in \Re_+^M$ where $\Re_+^M$ is the *M*-dimensional space of positive real numbers. Using the normalized objective values and the constraint subspace *W*, other configurations created from running a MOEA may be evaluated. More specifically, by definition:

$$\sum_{m=1}^{M} w_m (u_m - v_m) \leq 0 \;\Rightarrow\; u \preceq_{\text{preferred}} v \tag{3.9}$$

It follows that two alternatives $u$ and $u'$ can be compared by solving the following linear programming problem:

$$\begin{cases} Min \;\; \sum_{m=1}^{M} w_m (u'_m - u_m) \leq 0 \\ w_m \in W \end{cases} \tag{3.10}$$

Let $z = \min_{w_m} \sum_{m=1}^{M} w_m (u'_m - u_m)$ and $\bar{z} = \min_{w_m} \sum_{m=1}^{M} w_m (u_m - u'_m)$ then:

1) If $z < 0$ then $u'$ is preferred to $u$;
2) If $z \geq 0$ and $\bar{z} < 0$ then $u$ is preferred to $u'$;
3) If $z \geq 0$ and $\bar{z} \geq 0$ then $u$ and $u'$ are indifferent.

In summary, the DM is invited to make pairwise comparisons in order to define the constraint subspace *W*. *W* is subsequently used in the series of linear programming problems that should be solved to conduct pairwise comparisons between solutions. The authors noted that alternatives compared by the DM should be selected carefully so that they can be ranked consistently, unless conflicting constraints may be produced. Consequently, there will be no solution for the resulting linear programming problem, i.e., this latter would be infeasible. The authors implemented an algorithm for identifying the inconsistent preference statements. This algorithm identifies the minimum sets of preference statements that, if removed, would result in a feasible solution to the linear programming instance. However, this algorithm has an exponential time complexity.

- **Deb et al. (**2010**) work : the Progressively Interactive EMO Algorithm (PI-EMOA)**

The authors proposed a preference based-MOEA based on the concept of value function. Every few $\tau$ generations, the DM is provided with a sample of $\eta$ solutions and is asked to rank these solutions from the best to the worst where the incomparability

between solutions is allowed. This step is termed "*DM call*". Based on this preference information, an optimization problem is formulated and solved to find a suitable value function which optimally captures DM's preference information by maximizing the value function value between ranked points. From this iteration till the next DM call, the derived value function is utilized to drive the MOEA in: (1) modifying the domination principle which directly affects MOEA's convergence and diversity preserving operators, thereby guiding the search towards the preferred solutions and (2) determining the termination criterion of the overall procedure.

During the preference elicitation step (i.e., the DM call), the DM is provided with a sample of $\eta$ points and for each pair of alternatives $(x, y)$, he/she can precise if $x$ is preferred to $y$ (denoted $P_x \succ P_y$) or $x$ and $y$ are incomparable (denoted $P_x \equiv P_y$). For the bi-objective case, the authors proposed the following value function structure:

$$V(f_1, f_2) = (f_1 + k_1 f_2 + l_1)(f_2 + k_2 f_1 + l_2) \tag{3.11}$$

where $f_1$ and $f_2$ are the considered objective functions and $k_1$, $k_2$, $l_1$ and $l_2$ are unknown parameters and should be determined from the DM's preferences. For this purpose, the following Value Function Optimization Problem (VFOP) should be solved:

$$\begin{cases} Max\ \varepsilon \\ V \text{ is non negative at every point } P_x; \\ V \text{ is strictly increasing at every point } P_x; \\ V(P_x) - V(P_y) \geq \varepsilon, \text{ for all pairs } (x, y) \text{ staisfying } P_x \succ P_y; \\ \left|V(P_x) - V(P_y)\right| \leq \delta_V, \text{ for all pairs } (x, y) \text{ staisfying } P_x \equiv P_y; \\ \varepsilon > 0, \delta_V = 0.1\varepsilon. \end{cases} \tag{3.12}$$

The value function $V$, for two objectives shown above, is considered to be the product of two linear functions: $S_1 = f_1 + k_1 f_2 + l_1 : \Re^2 \to \Re$ and $S_2 = f_2 + k_2 f_1 + l_2 : \Re^2 \to \Re$. Considering all the expressions, we have the following optimization problem:

$$\begin{cases} Max\ \varepsilon \\ S_m(P_x) \geq 0, x = 1,...,\eta \text{ and } m = 1,2; \\ S_2(P_x) + k_2 S_1(P_x) \geq 0, x = 1,...,\eta; \\ k_1 S_2(P_x) + S_1(P_x) \geq 0, x = 1,...,\eta; \\ V(P_x) - V(P_y) \geq \varepsilon, \text{ for all pairs } (x, y) \text{ staisfying } P_x \succ P_y; \\ \left|V(P_x) - V(P_y)\right| \leq \delta_V, \text{ for all pairs } (x, y) \text{ staisfying } P_x \equiv P_y; \\ \varepsilon > 0, \delta_V = 0.1\varepsilon. \end{cases} \tag{3.13}$$

A little thought reveals that the above optimization problem attempts to find a value function for which the minimum difference in the value function values between the ordered pairs of points is maximal. For a general *M*-objective problem the value function can be written as follows:

$$\begin{aligned} V(f) = & \left(f_1 + k_{11} f_2 + k_{12} f_3 + ... + k_{1(M-1)} f_M + l_1\right) \times \\ & \left(f_2 + k_{21} f_3 + k_{22} f_4 + ... + k_{2(M-1)} f_1 + l_2\right) \times \\ & \cdots \\ & \left(f_M + k_{M1} f_1 + k_{M2} f_4 + ... + k_{M(M-1)} f_{M-1} + l_M\right) \end{aligned} \tag{3.14}$$

The above value function (3.14) can be expressed more elegantly as follows:

$$V(f) = \prod_{i=1}^{M} \left( \sum_{j=1}^{M} \left[ k_{ij} f_j + k_{i(M+1)} \right] \right) \tag{3.15}$$

For a general *M*-objective problem, the VFOP is expressed as follows:

$$\begin{cases} Max\ \varepsilon \\ S_m(P_x) \geq 0,\ x = 1,...,\eta \text{ and } m = 1,...,M; \\ k_{ij} \geq 0,\ i = 1,...,M \text{ and } j = 1,...,(M-1); \\ \left|V(P_x) - V(P_y)\right| \geq \varepsilon, \text{ for all pairs } (x,y) \text{ staisfying } P_x \succ P_y \\ \text{combinations satisfying } x < y; \\ \left|V(P_x) - V(P_y)\right| \leq \delta_V, \text{ for all pairs } (x,y) \text{ staisfying } P_x \equiv P_y; \\ \varepsilon > 0,\ \delta_V = 0.1\varepsilon. \end{cases} \tag{3.16}$$

Once the value function is build, the conventional dominance principle is modified in order to focus the search on preferred solutions. Let *V* be the value function found from the most recent decision-making interaction. Let $V_2$ be the value function value for the second best member ($P_2$ defined previously) from the sample of $\eta$ points. For the maximization case, any two feasible solutions *x* and *y* can be compared with their objective function values by using the following modified domination criteria:

1) If both solutions have a value function value less than $V_2$, then the two points are compared based on the usual Pareto dominance principle;
2) If both solutions have a value function value more than $V_2$, then the two points are compared based on the usual Pareto dominance principle;
3) If one has a value function value more than $V_2$ and the other has a value function value less than $V_2$, then the former dominates the latter.

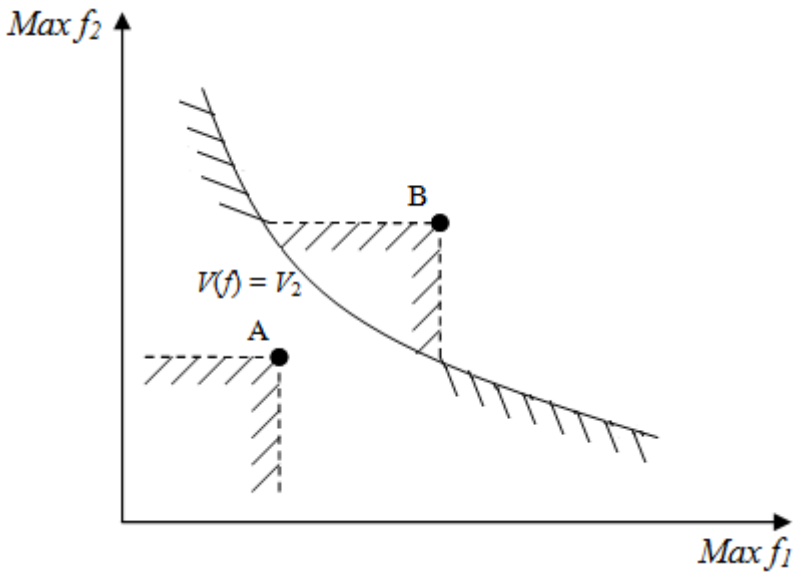
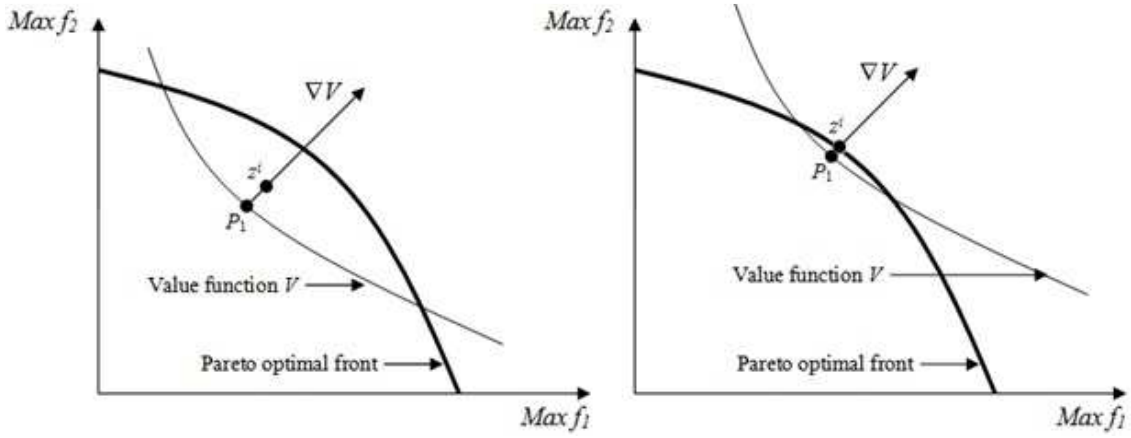



**Figure 3.3 Dominated regions for: (1) Pareto dominance with solution *A* and (2) value function-based dominance with solution *B*.**

Figure 3.3 illustrates a confrontation between the value function-based dominance dominated region and the Pareto dominance one for the bi-objective maximization case. This figure presents the region dominated by two points *A* and *B*. The value function contour having a value $V_2$ is shown by the curved line. The point *A* lies in the region in which the value function is smaller than $V_2$. The region dominated by point *A* is shaded. This dominated area is identical to that which can be obtained using the Pareto dominance principle. However, the point *B* lies in the region in which the value function is larger than $V_2$. For this point, the dominated region is different from that which would be obtained using the usual domination principle. In addition to the usual region of dominance, the dominated region includes all points having a smaller value function value than $V_2$.

Once the value function *V* is determined, the MOEA is driven by it in the next $\tau$ generations. The value function *V* can also be used for determining whether the overall optimization procedure should be terminated or not. To implement the idea, the best and second best points $P_1$ and $P_2$ from the given set of $\eta$ points are firstly identified based on the DM's preference information. The constructed value function can provide information about whether any new point *P* is better than the current best solution $P_1$ with respect to the value function. Thus, a single-objective search is performed along the gradient of the value function $(\nabla V)$ from $P_1$ in order to create better preferred solutions than $P_1$. This principle is used to develop a termination criterion by solving the following ASF problem for $P_1 = z^b$:




**(a)** **(b)**

**Figure 3.4 Local search along the value function gradient direction as a termination criterion: (a) success of the search and (b) failure of the search.**

$$Max \quad \min_{m=1}^{M}\left(\frac{f_m(x)-z_m^b}{\frac{\partial V}{\partial f_m}}\right)+\rho\sum_{m=1}^{M}\frac{f_m(x)-z_m^b}{\frac{\partial V}{\partial f_m}} \tag{3.17}$$

The second term with a small $\rho$ (= $10^{-10}$ used in this work) prevents the solution to converge to a weak Pareto optimal point. Any single-objective optimization method (e.g., the Sequential Quadratic Programming (SQP) method (Wilson 1963)) can be used for solving the above problem and the intermediate solutions $(z^i, i=1,2,...)$ can be recorded. If at any intermediate point, the Euclidean distance between $z^i$ from $P_1$ is larger than a termination parameter $d_s$, the ASF optimization task is stopped and the MOEA search is resumed. In this case, we replace $P_1$ with $z^i$. Figure 3.4(a) depicts this scenario for the bi-objective maximization case. If at the end of the SQP run, the final SQP solution (say, $z^T$) is not greater than $d_s$ distance away from $P_1$, the MOEA is terminated and $z^T$ is declared as the final preferred solution. This situation indicates that based on the current value function, there exists no solution in the search space which will provide a significantly better objective vector than $P_1$. Hence, the optimization run is terminated. Figure 3.4(b) shows such a situation, for the two-objective maximization case, warranting a termination of the PI-EMOA.

The PI-EMOA has shown its effectiveness on two- to five-objective test problems in providing the preferred point corresponding to a DM-emulated utility function. However, the authors have not handled the case in which the DM judges some of the $\eta$ points to be incomparable and the role of the $\delta_V$ parameter is not studied in the value function construction. Moreover, the authors noted that there are some cases which may

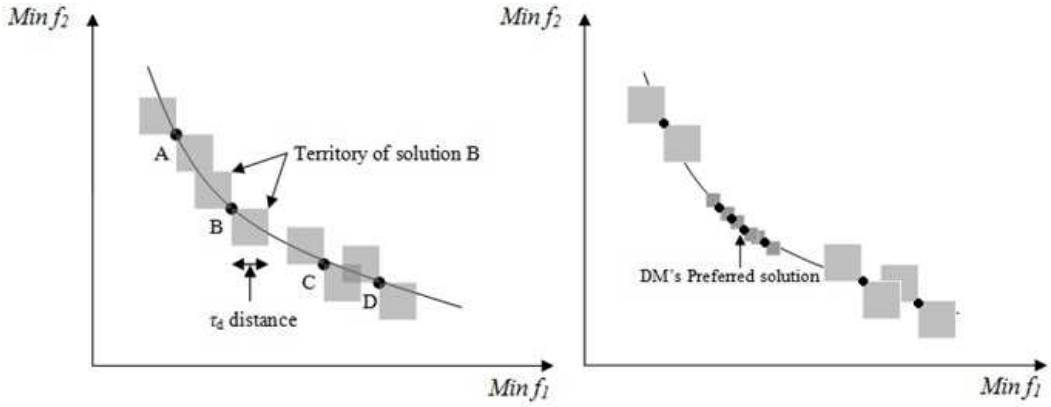
occur in which the building of a value function satisfying all DM's preferences is not possible.

- **Köksalan and Karahan (**2010**) work : the interactive Territory Defining Evolutionary Algorithm (iTDEA)**

The iTDEA is a preference-based interactive version of the TDEA (Karahan and Köksalan 2010). The TDEA is a new MOEA that approximates the whole Pareto front by using the concept of territory. This MOEA is a steady-state algorithm that maintains two populations: (1) the archive population that consists of individuals that are non-dominated relative to the population at hand and (2) the regular population that contains both dominated and non-dominated individuals. When updating the archive population, a territory around the individual closest to the offspring is defined and the offspring is rejected if it violates this territory. The territory defining property of TDEA eliminates the need for an explicit diversity operator, resulting in a fast operation while always keeping a diverse set of individuals in the archive population. The concept of territory is illustrated by figure 3.5(a). The territory region is mentioned with grey colour. The territory of a particular solution $x$ corresponds to the region with a distance $\tau_d$ from $f(x)$ in each objective among the region that neither dominates nor is dominated by $f(x)$. Mathematically, the territory of $f(x)$ is defined as the following hypervolume:

$$T(f(x)) = \{ f(y) \in \Re^M \,\|\, |f_m(x) - f_m(y)| < \tau_d \ \forall m = 1,\ldots,M \text{ and } \exists\, f_m(x) - f_m(y) < 0 \text{ for at leat one } m \text{ and } \exists\, f_m(x) - f_m(y) > 0 \text{ for at leat one } m \} \quad (3.18)$$

The authors proposed a strategy to choose a convenient $\tau_d$ value. The authors modified the TDEA in order to handle DM's preferences and focus the search on the preferred part of the front. The DM is supplied with a sample of diversified solutions and is invited to select the best one from his/her own perspective. In order to concentrate the search towards the preferred solution, the authors suggest shrinking the territories of the individuals falling near the preferred solution. This can be achieved by simply using a smaller $\tau_d$ for such offspring in the archive evaluation stage. This maintains more individuals in this region in the archive population, leading to a higher resolution and better approximation. Meanwhile, individuals located elsewhere are evaluated using a larger $\tau_d$. This leads to less population density in the regions that are less desirable to the DM. An illustration of this mechanism is shown by figure 3.5(b). The iTDEA has demonstrated its effectiveness in providing a biased distribution of the supplied non-dominated solution set where the distribution is denser near the DM's preferred solution. However, the authors noted that filtering the population to provide the potential sample to the DM from which he/she picks his/her preferred solution may




**(a)** **(b)**

**Figure 3.5 The territory effect with: (a) TDEA and (b) iTDEA.**

mislead the algorithm in the first interactions. They suggest increasing the number of solutions to present to the DM; however this number should have a limit.

- **Battiti and Passerini (**2010**) work : the Brain–Computer Evolutionary Multi-objective Optimization Algorithm (BC-EMOA)**

The authors suggested a preference-based MOEA characterized by its ability to learn an arbitrary utility function from a human DM who expresses his/her preferences between couples of selected solutions. The used method to build a flexible preference model, possibly highly non-linear, is based on the concept of Support Vector Machine (SVM) (Cohen et al. 1999). The objective of the learning process is the approximated construction of a utility function *U* to be optimized by the DM, who is also the source of learning signals. The function *U* to be optimized is not completely unknown, such as in a black-box context (Jones 2001), but is to be modelled based on the DM's ranking of candidate solutions. Preference models are built from the DM input by using a SVM-based ranking method. The functional form of the preference function is not fixed a priori by a well-defined ASF, such as in the weighted sum or Chebyshev approaches (Zhang and Li 2007), but is itself learnt during the process in a reactive fashion. The authors noted that SVM-based ranking has a number of desirable properties making it a suitable candidate for learning the DM's preferences. Firstly, it accepts supervision in terms of pairwise preferences, a much more affordable request for a human DM than a quantitative quality score. Secondly, it is well-grounded on learning theory; its trading-off data fitting and complexity of the learned hypothesis allows to effectively dealing with noisy observations, a situation which is quite likely to occur when receiving feedback from a human DM with only partial knowledge on the domain at hand. Thirdly, the ability to implicitly project data onto a higher dimensional feature space

via the kernel trick (Shawe-Taylor and Cristianini 2004) provides the needed flexibility in order to best approximate the underlying preference model of the specific user.

The basic functioning of BC-EMOA can be summarized as follows. Objective vectors are passed to the DM who ranks them and returns the ordered list as feedback. This feedback is converted into *pairwise constraints* for the SVM-based ranking procedure. After training, the predicted utility function *U* is employed to guide the search towards the ROI. From the multi-objective decision making perspective, the main contribution of this method is its ability to function without any a priori assumptions on the shape of the DM's utility function. The methodology of reactive search optimization (Battiti et al. 2008), based on the paradigm of learning while optimizing, is adopted in two directions: (1) the progressive tuning of a preference model following a DM's interactive evaluation and (2) the automated adaptation of the model form to one which is most appropriate, in a cross-validated manner, to the data collected during the interaction. The method is robust as it can potentially withstand incomplete, imprecise and even contradictory feedback by the DM. The BC-EMOA is a generic formulation which can be implemented on top of any MOEA. In this study, the authors adopted the NSGA-II. This latter runs in its original formulation, including the crowded-comparison operator for guaranteeing a sufficiently diversified population, for $gen_1$ generations. After that, the preference model is trained according to the DM's feedback and the ordering of the new population and the selection criterion of the binary tournament selection operator are performed based on the actual utility function value. Additionally, the crowding mechanism is switched-off at this point as the goal is directing the generation of new individuals towards the ROI. The BC-EMOA has demonstrated its effectiveness in guiding the search towards the DM's most preferred solution on some selected DTLZ problems (cf. section 2.4.1) in addition to some instances of the 0/1 multi-objective knapsack problem (Martello and Toth 1990). However, the authors noted that the $gen_1$ parameter value should allow a reasonable coverage of the Pareto front in order not to miss portions possibly containing the DM's preferred solutions; it should thus be of the same order of the number of generations for a plain MOEA run on the same problem. This step seems to be computationally costly. Additionally, after $gen_1$ generations, the crowding mechanism is turned-off and the population is guided towards a certain region of the search space based on the utility function which can reduce population diversity significantly and encourage the premature convergence to occur especially on multimodal MOPs. This problematic is omitted by the authors.

- **Fowler et al. (**2010**) work: the cone-dominance-based approach**

The authors exploited the notion of preference convex polyhedral cones (Ramesh et al. 1988) in order to integrate DM's preferences in a MOEA. Periodically, the DM is provided with a sample of $\eta$ solutions and is invited to specify the best alternative in addition to the worst one from his/her own perspective. These two selected solutions are then used to form a convex polyhedral cone. The cone defines a convex set of solutions that are inferior to the cone vertex in addition to solutions residing within the cone. Since the designed algorithm is interactive, all defined cones are retained during the overall optimization run whether or not the population members from which they are derived are still surviving. Figure 3.6 illustrates graphically the concept of convex polyhedral cone for the bi-objective maximization case for: (a) the case of two solutions (*B* is preferred to *A*) and (b) the case of three solutions (*B* is preferred to *A* and *C* is preferred to *A*). For the case of $\eta$=2 (cf. figure 3.6(a)), the convex cone corresponds to the line segment *AB* and has solution *A* as vertex. Accordingly, every solution that is dominated by the cone (shaded region in figure 3.6(a)) is less preferred to solution *A*, and hence to every solution belonging to the line segment *AB*. These solutions are considered to be inferior and are to be discouraged in the selection process of the MOEA. For the case of $\eta$=3 > 2 (cf. figure 3.6(b)), the preference cone corresponds to a convex polyhedral set. In this case, any population member can have one of the following four possible locations: (1) under the cone (like solution *F*), (2) in the cone (like solution *D*), (3) outside the cone (like solution *E*) or (4) being the cone's vertex (like solution *A*). The DM's preference information is applied by placing solution *v* before solution *u* if *v* is within the cone and *u* is the cone's vertex. The three-point cone in figure 3.6(b) is composed from the union of two two-point cones defined by the two preference relations: (1) solution *B* is preferred to solution *A* (the corresponding shaded area is drawn with thin lines) and (2) solution *C* is preferred to solution *A* (the corresponding shaded area is drawn with bold lines). Solutions under the cone (shaded area) like *F* are considered to be dominated by the cone. We say that they are cone-dominated. Solution *D* is considered to be superior to solution *A* since *D* belongs to the cone and *A* is the cone's vertex. It is important to note that every solution from the grey region defined by points *A*, *B* and *C* is considered to be superior to the cone's vertex *A* and subsequently preferred to every point belonging to the shaded region.

The cone-dominance principle is used in parent selection and replacement mechanisms. The designed preference-based MOEA has been assessed on the multi-objective 0/1 knapsack problem with 2, 3 and 4 objectives. The computational experiments have shown that it is possible to obtain solutions with a reasonable number of DM interactions that are very near or equal to the best found by a similar algorithm that is operating with perfect knowledge of the user's preference function. However, the

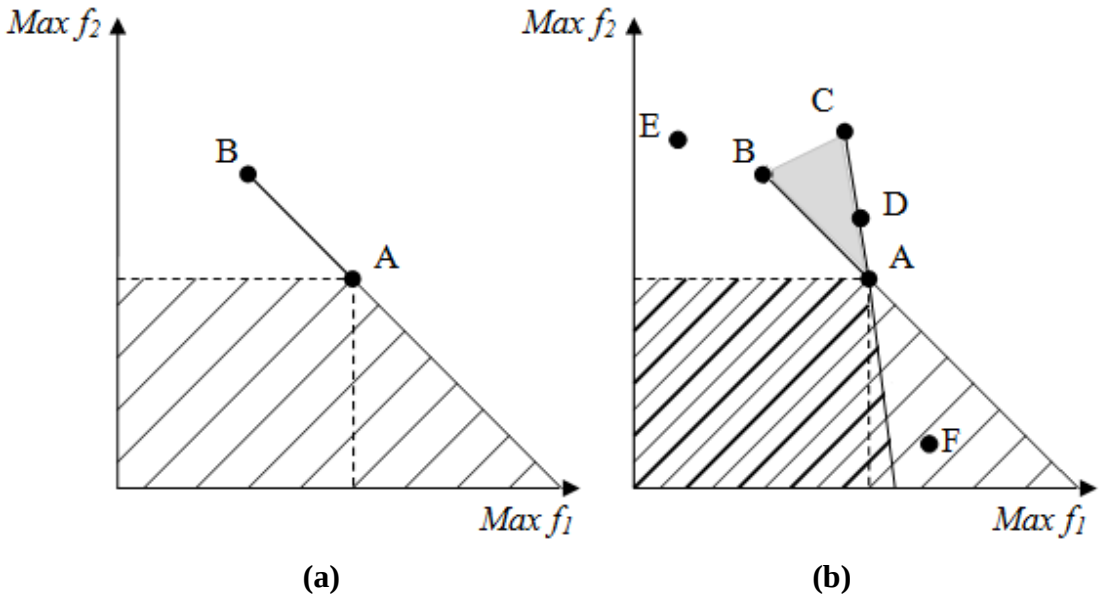



**Figure 3.6 Two preference cones: (a) defined with the two solutions *A* and *B* and (b) defined with the three solutions *A*, *B* and *C* (inspired by** (Ramesh et al. 1988)**).**

authors noted that investigating the effect of DM's preferences inconsistencies is still a direction for future research. Moreover, there is no control over the ROI extent.

♦ **Branke et al. (**2010**) work: the Necessary preference-enhanced Evolutionary Multi-objective Optimizer (NEMO)**

The NEMO algorithm is the result of the combination of NSGA-II and the Robust Ordinal Regression (ROR) (Greco et al. 2010) within an interactive procedure. In ROR, the DM is presented with a small set of alternatives and can express his/her preferences by specifying a holistic preference of one alternative over another or comparing intensities of preferences between pairs of alternatives. ROR then identifies the whole set of additive value functions compatible with the preference information given by the DM. This allows comparing any pair of alternatives, *x* and *y*, in a simple and intuitive way, as follows:

1) *x* is necessarily at least as good as *y*, if this is true for all compatible value functions;
2) *x* is possibly at least as good as *y*, if this is true for at least one compatible value function.

The authors noted that, usually, among the many sets of parameters of a preference model representing the preference information, only one specific set is used to give a recommendation on a set of alternatives. For example, among many value functions representing pairwise comparisons of some alternatives made by the DM, only one value function is finally used to recommend the best choice or to sort or to rank the

alternatives. Since the choice of one among many sets of parameters compatible with the preference information is rather arbitrary, ROR has been recently proposed with the aim of taking into account all the sets of parameters compatible with the preference information given by the DM (Greco et al. 2008; Figueira et al. 2009). The ROR approach extends the simple ordinal regression by taking into account not a single instance of the preference model compatible with DM's preference information, but the whole set of compatible instances of the preference model. As a result of considering the whole set of compatible instances of the preference model, one gets two kinds of results with respect to each pair of alternatives *x* and *y*:

1) *necessary preference relation* ($x \succsim^N y$), if and only if *x* is at least as good as *y* according to all instances of the preference model compatible with the preference information;
2) *possible preference relation* ($x \succsim^P y$), if and only if *x* is at least as good as *y* according to at least one instance of the preference model compatible with the preference information.

Since, NEMO, is a modified version of NSGA-II, we give the modifications performed to this latter as follows:

1) the Pareto dominance relation is replaced by the necessary preference relation in the non-dominated sorting;
2) the crowding distance is substituted by a distance calculated by taking into account the multidimensional scaling given by the most representative value function among the whole set of compatible value functions. The most representative value function corresponds to the value function which maximizes the difference of scores between alternatives related by preference in the necessary preference relation-based ranking (Figueira et al. 2008).

The NEMO algorithm has demonstrated its ability to bias the search towards the ROI interactively. However, the algorithm was tested only on two bi-objective test problems. Consequently, a more thorough empirical analysis on a variety of test problems with more than two objective functions is necessary. Moreover, there is no control over the ROI spread.

### 3.2.3 Objective ranking-based approaches

♦ **Jin and Sendhoff (**2002**) work**

The authors turned fuzzy preferences into weight intervals which were incorporated into a MOEA using Random Weighted Aggregation (RWA) and Dynamic Weighted Aggregation (DWA) techniques (Jin et al. 2001). This was achieved by setting the

upper and lower bounds to the weight perturbations. In fact, the DM is invited to make pairwise comparisons on the set of objectives by using linguistic statements such as "objective $f_2$ is much more important than objective $f_1$", etc. The authors developed a method that converts these fuzzy preferences into interval-based weights where each weight indicates the importance of the relative objective. This approach converts the MOP into a SOP by weighted aggregation, but varies the weights dynamically during the run within the relevant boundaries.

For the RWA, assuming that each individual $i$ has it own weight combination $(w_1^i(t), w_2^i(t))$ in generation $t$ for the bi-objective case, then the MOEA is able to find different Pareto optimal solutions. The weight combinations need to be distributed uniformly and randomly among the individuals in each generation as follows:

$$w_1^i(t) = \text{random}(P)/P \tag{3.19}$$

$$w_2^i(t) = 1 - w_1^i(t) \tag{3.20}$$

where $P$ is the population size and random is function that generates a uniformly distributed random number between 0 and $P$.

For the DWA, all the individuals have the same weight combination which is changed gradually in each generation. The change of the weights is realized as follows assuming a bi-objective case:

$$w_1(t) = \left|\sin(2\pi t/F)\right| \tag{3.21}$$

$$w_2(t) = 1 - w_1(t) \tag{3.22}$$

The weights will change from 0 to 1 periodically from generation to another. The change frequency can be adjusted by the $F$ parameter.

In both RWA- and DWA-based EMO approaches, the weight varies in the interval [0,1] in order to approximate the whole Pareto front. However, in order to take the DM's preferences into account, the weight of each objective $f_m$ is varied in the interval $[w_m^{\min}, w_m^{\max}]$ where the boundaries of the latter interval are obtained from the conversion of the DM's fuzzy preferences. In this way, the search process is guided towards the ROI. The designed weighted sum-based algorithms support incomparability between solutions and provide the user with a control over the focus extent. While DWA facilitates retention of compromise solutions in the non-convex parts of the non-dominated front, the lack of explicit diversity preservation and inferior performance in high-dimensional problems constitute significant drawbacks of the

approach (Jin et al. 2001). Arriving at the upper and lower bounds of weights for higher dimensional problems is also difficult.

♦ **Cvetkovic and Parmee (**2002**) work: the weighted-dominance relation-based approach**

The authors proposed the integration of DM's fuzzy preferences into MOEAs by converting linguistic variables into weights. The DM is invited to make pairwise comparisons between the MOP's objectives by using some linguistic labels such as "more important", "much less important", "do not care", etc. As the number of objectives increases, the number of pairwise comparisons becomes a tedious task for the DM. The use of transitive relations was therefore proposed to reduce the number of pairwise comparisons required from the DM (Cvetkovic and Parmee 2002). The reader is invited to confer to the original paper to explore the details of the mechanism converting the linguistic terms to weights. Based on the obtained weight vector expressing the relative importance for each objective, a new weight-based dominance relation is designed. This relation is called weighted-dominance relation and is expressed by:

$$x \preceq_{\mathrm{w}} y \;\Leftrightarrow \sum_{m=1,\ldots,M,\; f_m(x) \le f_m(y)}^{M} w_m \ge \tau \tag{3.23}$$

with a strict inequality for at least one objective. $x \preceq_{\mathrm{w}} y$ means that solution $x$ is preferred to solution $y$ based on weighted-dominance and $\tau$ is a user-defined parameter expressing the minimum required level of dominance. The main drawback of this dominance relation is that it only considers the number of improvements of one solution with respect to another one and it ignores the amount of each improvement. Additionally, the control of the guidance is difficult and there is no clear interest to use such approach in an interactive way. In a more recent study, Rachmawati (2008) have discussed the effects produced by the obtained values for the weighting coefficients. The weighted-dominance relation preserves Pareto-dominance relation and also allows incomparability when $\tau$ is set such that $\tau \ge \min_{m=1,\ldots,M} w_m$. However, the weighted-dominance has a serious drawback illustrated in the following for a bi-objective MOP. Without loss of generality, we assume that $w_1 > w_2$. For bi-objective problems, three scenarios with respect to different values of $\tau$ exist. The dominated and non-dominated regions of the objective space around a candidate solution in the three scenarios are as follows:

1) $\tau \ge w_1$: In this scenario, the weighted-dominance is equivalent to the Pareto dominance.

2) $\tau \leq w_2$ : In this scenario, the weighted-dominance includes all solutions non-dominated in terms of the Pareto-dominance relation. In an MOEA implementing weighted Pareto dominance with this setting, there will be no non-dominated solution according to the weighted-dominance unless a candidate solution that strongly Pareto dominates all other solutions in the population is present. If this latter does not exist in the population, the MOEA degenerates into a random search.
3) $w_2 \leq \tau \leq w_1$ : In this scenario, the weighted-dominance includes solutions non-dominated in terms of the Pareto dominance relation with inferior $f_2$ values. In an MOEA implementing the weighted-dominance with this setting, the solution in the best non-dominated front that also corresponds to the smallest $f_2$ value dominates all other solutions. Unless the archiving policy allows inclusion of weakly dominated solutions, only the extreme solution is retained in each generation. With an archival policy that allows dominated solutions with respect to the weighted-dominance, the archive includes solutions other than the extreme solution that corresponds to the best secondary niching criterion. Preference is only incorporated in the search as much as the inclusion of the extreme solution corresponding to the smallest attainment $f_1$ into the archive is guaranteed. However, even in general purpose MOEAs using the Pareto dominance relation, the inclusion of extreme solutions in the archive is always guaranteed.

- **Rachmawati and Srinivasan (**2010**) work**

In this approach, the DM is invited to express his/her preferences in the form of relative importance of objectives without using any weighting coefficient. In fact, the DM is invited to specify a total or partial order on the set of objectives. For each pair of objectives ($f_m$, $f_n$), the DM can express one of the following statements: (1) objective $f_m$ is preferred to objective $f_n$ (denoted $f_m \; Pr \; f_n$), (2) $f_m$ and $f_n$ are equally important (denoted $f_m \; I \; f_n$) or (3) $f_m$ and $f_n$ are incomparable (denoted $f_m \; Q \; f_n$). An elicitation algorithm is provided to assist the DM in constructing a coherent overall preference. Besides elicitation of a priori preference, an interactive facility is also furnished to enable modification of overall preference while the search progresses. We note that the default preference relation between pairs of objectives is the incomparability relation. If incomparability is the only preference relation considered by the DM, the whole Pareto front is returned as a general-purpose MOEA.

A way of consistently characterizing the preferred solutions for a given preference profile irrespective of the geometry of the Pareto front is desirable. This consistency is instrumental to an effective articulation of preference by the DM. Even if the geometrical attributes of the actual Pareto front are unknown a priori, a consistent

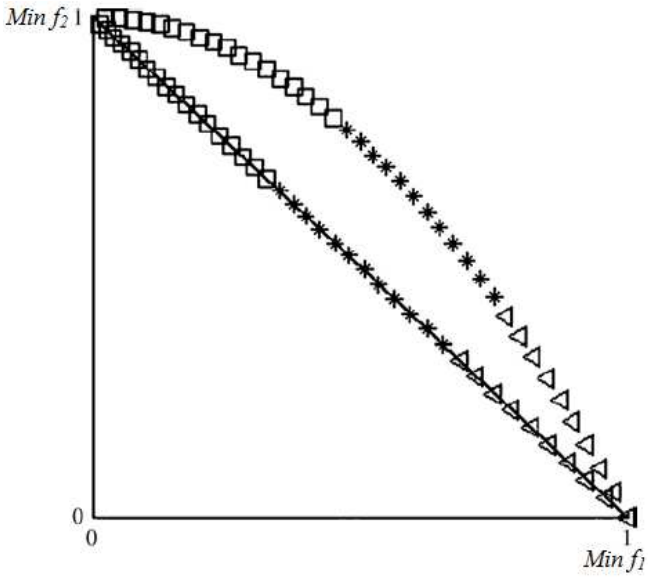



**Figure 3.7 Desired solutions corresponding to $f_1$ *P* $f_2$ (squares), $f_1$ *I* $f_2$ (asterisks) and $f_2$ *P* $f_1$ (triangles) (from** (Rachmawati and Srinivasan 2010)**).**

characterization equips the DM with some information of the solutions he/she may expect for any given preference profile. To achieve this consistency, the functional mapping from preferences expressed in the preference structure $P_F$ (i.e., the group of binary relations that can be defined on the set of objective pairs) to the Pareto front is defined in terms of a prototype non-dominated front. The selected prototype front is linear, continuous and defined in the interval [0,1]. Let the prototype front be described by $\tilde{F} = [\tilde{f}_1, \tilde{f}_2]$, then $\tilde{f}_1 + \tilde{f}_2 = 1$. An illustration of the prototype front and a mapping from the actual front is given in figure 3.7. The choice of the linear front is motivated partly by its simplicity and its scalability with the number of objectives. Simplicity helps the DM in formulating his/her preference in terms of binary relations in $P_F$. The scaling of the preference model and its functional mapping to *M*-objective problems are described next.

To accommodate the three binary preference relations defined in $P_F$, the prototype front is divided into three non-overlapping segments of identical length as depicted by figure 3.7. The linear front in this figure is the prototype front while the curve is a normalized concave Pareto front associated with an actual MOP. The first portion plotted as squares is desired when $f_m$ *Pr* $f_n$ is asserted. The second and third portions, marked by asterisks and triangles respectively, are the desired subsets of the front when $f_m$ *I* $f_n$ and $f_n$ *Pr* $f_m$ are asserted respectively. We recall that the preference assertion $f_m$ *Q* $f_n$ corresponds to the entire span of the prototype Pareto front. Mathematically, the desired subsets of the Pareto front could be characterized by the following inequalities:

$$f_1 \, P \, f_2 \;\Leftrightarrow\; 2\tilde{f}_1 \le \tilde{f}_2 \tag{3.24}$$

$$f_1 \, I \, f_2 \;\Leftrightarrow\; \tilde{f}_1 \le \tilde{f}_2 \le 4\tilde{f}_1 \tag{3.25}$$

where $\tilde{f}_1$ and $\tilde{f}_2$ correspond to the prototype objective space and Pareto front. The choice of the coefficients 2 and 4 in the above inequalities follows from equal division of the prototype front into three non-overlapping subsets. Other values may of course be used if other ways of dividing the prototype front is deemed necessary or desirable. In this approach, equal and non-overlapping division is adopted as it is deemed most intuitive for the general case.

The authors proposed three versions of the NSGA-II where DM's preferences are incorporated as follows:

1) *Inclusion of preference information as constraints*: To incorporate preference, the inequalities (3.18) and (3.19) are applied to the current population and/or archive where normalization is done with respect to the extrema of the best non-dominated front. In this particular strategy, the ROI as defined by inequalities derived from the partial ranking of solutions is considered as the feasible region. The following rule is applied when comparing a pair of solutions ($x$, $y$) in the population and/or the archive:

**If** (($V(x) > 0$) and ($V(y) > 0$)) **Then**
  **If** ($V(x) < V(y)$) **Then**
    $x \prec_{pref} y$
  **Else If** ($V(y) < V(x)$) **Then**
    $y \prec_{pref} x$
  **End If**
**Else**
  **If** ($x \preceq y$) **Then**
    $x \prec_{pref} y$
  **Else If** ($y \preceq x$) **Then**
    $y \prec_{pref} x$
  **End If**
**End If**

The function $V$(x) in the above rule denotes a measure of constraint violation of solution $x$, which is taken to be the maximum magnitude of the violation of all inequalities describing the desired region. The expression $x \prec_{pref} y$ indicates that solution $x$ is preferred to solution $y$.

2) *Inclusion of preference information as rank penalty*: Pareto ranking introduces a complete order to the partially ordered objective space by means of existing dominance relation between solution pairs in the set. Here, incompatibility with the preference-based inequalities incurs a penalty in the Pareto rank of a solution. As Pareto rank is usually defined as integers, the penalty imposed is equal to one. The strategy works only with MOEAs that implement Pareto ranking. Pareto dominance is preserved in NSGA-II by performing the ranking from the best non-dominated layer such that non-dominated solutions satisfying preference-based inequalities are assigned rank 1 (subset 1), non-dominated solutions not satisfying preference inequalities are assigned rank 2 (subset 2) along with solutions which satisfy preference inequalities and are non-dominated with solutions in subset 2. This process is repeated until the population is filled. In this manner, the search is guided based on DM's preferences.
3) *Inclusion of preference in the crowding distance computation:* Satisfaction of the preference inequalities leads to a multiple of the actual crowding distance of a solution to be considered as the crowding distance, i.e., if a solution satisfies the inequalities then *CrowdingDistance* = *Factor* × *ActualCrowdingDistance* with *Factor* is greater than one, whereas dissatisfaction with inequalities corresponds to factor equals one. The multiplication factor is the biasing strength of this approach. This strategy is applicable to any MOEA that implements crowding in the fitness computation.

The three NSGA-II versions were assessed on two- to six-objective test functions. The constraint-based approach and the rank penalty-based approach have demonstrated their abilities to provide a ROI based on DM's preferences. The crowding-based approach has shown its effectiveness in biasing the non-dominated solution distribution towards the preferred Pareto front subset. However, there is no control over the ROI spread.

### 3.2.4 Reference point-based approaches

♦ **Fonseca and Fleming (**1993**) work**

This work is probably the first attempt to incorporate DM's preference information in EMO. The authors model DM's preferences as a goal to be achieved (i.e., a reference point). The main idea, in this study, is to give higher priority to objectives that do not satisfy the goal. Assuming a goal $g = (g_1,...,g_M)$ and two objective vectors $u = (u_1,...,u_M)$ and $v = (v_1,...,v_M)$ to be compared, there exist three cases:

**Case 1:** $u$ meets $M-k$ goals (i.e., $M-k$ of the specified goal components). This can be expressed as follows:

$$\exists k = 1,...,M-1 / \forall i = 1,...,k; \forall j = k+1,...,M \quad (u_i > g_i) \wedge (u_j \leq g_j) \tag{3.26}$$

The expression (3.26) assumes a convenient permutation of the objectives.

**Case 2:** $u$ does not meet any goal. This can be expressed as follows:

$$\forall i = 1,...,M \quad (u_i > g_i) \tag{3.27}$$

**Case 3:** $u$ meet all the goals. This can be expressed as follows:

$$\forall i = 1,...,M \quad (u_i \leq g_i) \tag{3.28}$$

In case 1 (cf. (3.26)), $u$ meets the goals $k+1...M$ and therefore it is considered to be preferred to $v$ if it Pareto dominates $v$ with respect to its $k$ components. For the case where all of the $k$ components of $u$ are equal to those of $v$, $u$ is preferred to $v$ if it Pareto dominates $v$ with respect to the remaining $M-k$ components or if the remaining $M-k$ components of $v$ does not meet their goals. Analytically, $u$ is preferred to $v$ (denoted $u \prec_p v$) if and only if:

$$\begin{array}{l} (u_{(1,...,k)} \preceq v_{(1,...,k)}) \vee \\ \{(u_{(1,...,k)} = v_{(1,...,k)}) \wedge [(u_{(k+1,...,M)} \preceq v_{(k+1,...,M)}) \vee \neg(v_{(k+1,...,M)} \leq g_{(k+1,...,M)})]\} \end{array} \tag{3.29}$$

In case 2 (cf. (3.27)), $u$ does not satisfy any goal, then $u$ is preferred to $v$ if and only if $u$ Pareto dominates $v$, i.e.:

$$u \preceq v \tag{3.30}$$

In case 3 (cf. (3.28)), $u$ meet all the goals which means that it is a satisfactory, though not necessarily optimal solution. Then, $u$ is preferred to $v$ if and only if $u$ Pareto dominates $v$ or $v$ is not a satisfactory solution, i.e.:

$$(u \preceq v) \vee \neg(v \leq g) \tag{3.31}$$

This approach can be used a priori or interactively. The authors also proposed an expert system ensuring the task of supplying goals since setting an appropriate goal is not a trivial task. However, if the goal has been set so ambitious that there is no solution which can reach the goal in even a single objective, the goal has no effect on the search, and simply the whole Pareto front is returned. Consequently, we can say that the obtained results heavily depend on the position of the goal in the objective space. Moreover, the spread of the obtained ROI cannot be controlled and the proposed approach does not consider this issue.

- **Tan et al. (**1999; 2003**) work**

The authors proposed a variant of the Pareto dominance incorporating goal and priority information. In the first stage, the ranking scheme prefers objective vectors fulfilling all criteria and ranks those vectors according to MOGA Pareto dominance-based sorting (Fonseca and Fleming 1993). Among the remaining solutions, the objective vector $u$ dominates the objective vector $v$ if and only if $u$ dominates $v$ with respect to the criteria in which $u$ does not fulfil the goal $g = (g_1,...,g_M)$ (as in (Fonseca and Fleming 1993)), or if $|u-g| \prec |v-g|$ (where $|u-g|$ denotes the vector composed with the absolute values of the differences between the objectives' values of the solution $v$ and the goal $g$). The latter expression corresponds to a *mirroring* of the objective vector along the axis of the fulfilled criteria. Analytically, $u$ is preferred to $v$ (denoted $u \pi_g v$) if and only if:

1) $u$ and $v$ both satisfy all the goals and $u \preceq v$; or
2) $u$ and $v$ both does not satisfy all the goals and $\hat{u} \preceq \hat{v}$ or $|u-g| \preceq |v-g|$, where $\hat{u}$ and $\hat{v}$ corresponds to the vectors composed with the components that does not fulfil the goals of $u$ and $v$ respectively.

The main advantages of this approach is the possibility to consider multiple goals by the use of *AND* and *OR* connectives. The main drawback of this approach is that this kind of dominance is intransitive, i.e., it may lead to the case where $x$ is preferred to $y$ and $y$ is preferred to $z$, but $x$ and $z$ are considered as equivalent.

♦ **Deb et al. (**2006a**) work: the Reference point-based NSGA-II (R-NSGA-II)**

R-NSGA-II is a modified version of NSGA-II that focuses the search on the ROIs according to a user-provided reference point set. The reference points are used to guide the search towards the preferred parts of the Pareto front. In fact, the crowding distance of NSGA-II is modified as follows. For each reference point, the normalized Euclidean distance of each solution of the front is calculated and the solutions are sorted in ascending order of distance. The closest solution from the reference point is assigned a rank of one; the second nearest solution is assigned a rank of two and so on. After such computations are performed for all reference points, the crowding distance of a certain solution is equal to the minimum of its assigned ranks. In this way, solutions closest to all reference points are assigned the smallest crowding distance of one. The solutions having next-to-smallest Euclidean distance to all reference points are assigned the next-to-smallest crowding distance of two, and so on. Thereafter, solutions with smaller crowding distances are preferred in the tournament selection and in forming the new population from the combined population of parents and children. In order to control the extent of the obtained solutions, all solutions having a sum of normalized difference

in objective values of $\varepsilon$ or less between them are grouped. A randomly picked solution from each group is retained and the rest of all group members are assigned a large crowding distance in order to discourage them to remain in the race. The above procedure provides an equal emphasis of solutions closest to each reference point, thereby allowing multiple ROIs to be found simultaneously in a single simulation run.

R-NSGA-II has demonstrated good results on two- to five-objective test problems. However, there were difficulties when using a single reference point since diversity is not well-maintained. Moreover, the $\varepsilon$ clearing parameter setting is not trivial.

♦ **Deb and Kumar (**2007a**) work: the reference direction-based approach**

The authors combined the reference direction method with NSGA-II. The reference direction method allows the DM to set a starting point and a reference point such that the difference of the two defines the reference direction. Firstly, a set of points $(r(t), t \in \aleph)$ are marked on the given reference direction. Then, for each point $r(t)$, we compute the ASF value $s(z, r(t), w)$ for a chosen weight vector $w$ and for each population member $z$. Thereafter, the individual $z^*$ having the smallest value of $s$ is declared to lie on the first non-dominated front by assigning it the rank of one. This procedure is continued for each point $r$ and the corresponding population member for the minimum $s$ is included in the first non-dominated front. Thereafter, these chosen population members are temporarily discarded from the population and the above procedure is repeated. The next set of minimum $s$ solutions is then declared to form the second non-dominated front. This procedure is repeated till all population members are classified into non-dominated frontiers. Thereafter, the crowding distance is computed for each of the classified population members as usual. This hybrid method has the ability to find Pareto optimal solutions corresponding to several reference points along the reference direction. Several preferences could be modelled by various reference directions and the hybrid algorithm found for each reference direction its corresponding ROI. The authors noted that the population size in such a NSGA-II version should be at least two or three times the number of points considered along the reference direction. The multiplicity is needed to ensure that the search is adequately guided towards the corresponding efficient point. The reference direction approach has demonstrated good results in tackling two- to ten-objective MOPs. However, the population diversity degradation that can be yielded when using a single reference direction remains a significant matter since this approach does not include a clearing mechanism such as the R-NSGA-II one.

♦ **Deb and Kumar (**2007b**) work: the light beam search-based approach**

The authors combined the LBS method (cf. section 2.3.1) with NSGA-II. In the original LBS method, the DM has to specify three preference parameters for each objective which is quite demanding on the part of the DM. In order to reduce the DM's load, the authors use only the veto preference parameter. Once, the middle point (cf. figure 2.4) is obtained, the feasible direction of the largest improvement of each objective is determined. The best feasible point in each direction satisfying the outranking criterion is determined. These points are then projected on the Pareto optimal front by solving an augmented form of Wierzbicki's ASF. This results in the best feasible point in each direction satisfying the outranking criterion and Pareto optimality. In this hybrid preference-based MOEA, the DM is asked to supply an aspiration point (i.e., a reference point) and a reservation point. The hybrid algorithm is as follows:

1) Non-dominated sorting is performed for the whole population,
2) For each front, each solution from the front is assigned a crowding rank:
   a) Crowding distance of each solution $x$ (denoted $cd(x)$) is computed as:

$$cd(x) = \max_{m=1,\ldots,M} \left\{ \lambda_m \left( f_m(x) - z_m^a \right) \right\} + \rho \sum_{m=1}^{M} \left( f_m(x) - z_m^a \right) \tag{3.32}$$

   where $z^a = [z_1^a, \ldots, z_M^a]$ is the aspiration point, $\Lambda = [\lambda_1, \ldots, \lambda_M]$ is the weighting coefficient vector ($\lambda_m > 0 \quad \forall m = 1, \ldots M$) and $\rho$ is a sufficient small positive number (called augmentation coefficient which is fixed to $10^{-6}$ here). The weighting vector can be defined by the aspiration point $z^a$ and the reservation one $z^r$ (where $z_m^a < z_m^r \quad \forall m = 1, \ldots, M$) as follows:

$$\lambda_m = \frac{1}{z_m^r - z_m^a} \tag{3.33}$$

   b) Solution with least *cd* value is the middle point $z^c$ and it is assigned the highest crowding rank.
   c) For all solutions $x$ outranking $z^c$, the maximum difference in objective value with $z^c$ is determined:

$$\varphi(x) = \max_{m=1,\ldots,M} \left( f_m(x) - z_m^c \right) \tag{3.34}$$

   Based upon the $\varphi(x)$ value, a crowding rank is assigned to each solution. Solutions with smaller $\varphi(x)$ are assigned higher ranks and vice versa.

d) The remaining solutions, that do not outrank $z^c$, are assigned smaller crowding ranks so they are discouraged during the selection process.

In the case of multiple light beams, a crowding rank corresponding to each light beam is first determined for each solution. Then, the minimum rank for all light beams is assigned as the final crowding rank of the considered solution.

3) In order to obtain a uniform distribution in the lighted regions, no two objective vectors apart by less than an $\varepsilon$ distance are preferred in the same manner as the clearing procedure of R-NSGA-II.

The modified outranking relation used in this work is:

$$\begin{cases} f(x)\ S\ z^c & \text{if } t_v(f(x), z^c) = 0 \\ t_v(f(x), z^c) = card\{m : f_m(x) - z_m^c \geq veto_m, m = 1,..., M\} \end{cases} \quad (3.35)$$

The objective vector $f(x)$ outranks $z^c$ (denoted $f(x)\ S\ z^c$) means that $f(x)$ is as good as $z^c$. As both solutions belong to the same non-dominated front, if $f(x)$ is better than $z^c$ in some objectives, the amount of deterioration of $f(x)$ over $z^c$ must not exceed the corresponding provided veto thresholds ($veto_m$, $m$=1,…,$M$).

This hybrid algorithm has demonstrated its ability to find the part of the Pareto optimal region illuminated by the light beam emanating from the reservation point to the aspiration point with a span controlled by the veto thresholds. The simulation results have shown good results when applying this approach on a suite of benchmarks. However, providing the veto thresholds is not an easy task for the human DM. Further efforts are needed to study how to help the DM to elicit such parameters.

- **Allmendinger et al. (**2008**) work: the Reference point-based Particle Swam Optimization using a Steady State approach (RPSO-SS)**

The authors hybridized the Particle Swarm Optimization (PSO) metaheuristic with the reference point method. The authors used a steady state approach where an offspring is generated one at a time. A replacement strategy is often employed to compare the offspring with its parents. The offspring only replaces a weaker parent. Note that this procedure results in a population size that is constant during the entire run of the algorithm. There is no notion of generation. The velocity is set randomly to be within the variable ranges. The DM is invited to supply one or more reference point(s). The population is divided into equal sized clusters each focusing on one reference point. The main loop is described as follows. Do the following for each particle $x$ in each cluster until a stopping criterion is met:

1) Choose a particle randomly from the current cluster as the best global (or local) best position $p^g = [p_1^g, ..., p_M^g]$,
2) Produce an offspring based on *x*, $p^g$ and $p^b$ where $p^b = [p_1^b, ..., p_M^b]$ is the best position found by the particle *x* so far during the optimization process,
3) If the generated offspring Pareto dominates *x*/$p^g$ then it replaces it, else if the offspring is non-dominated with respect to both *x* and $p^g$ then the two particles closest to the reference point are kept and the farthest particle is deleted.

RPSO-SS uses also an $\varepsilon$-clearing mechanism like R-NSGA-II in order to preserve population diversity and to allow the DM to control the ROI extent. The experimental study has mentioned good results in solving two- and three-objective test problems. However, there were difficulties when solving highly multimodal problems such ZDT4.

In the same study, the authors proposed an extended selection strategy. In fact, the original selection strategy will not keep an offspring in the population that is farther than *x* and $p^g$ from the reference point. The new replacement strategy extends the replacement strategy of the basic RPSO-SS and provides the offspring with the opportunity to replace particles other than *x* or $p^g$. Thus, a randomly selected particle *y* is compared to the offspring and it is replaced if it is dominated by the offspring or it is non-dominated regarding the offspring but it is farther than the offspring from the reference point. Additionally, instead of using a single randomly selected particle as a $p^g$, the new sampling-based selection mechanism selects $Np^g$ particles randomly. Among these particles, the one with the shortest Euclidean distance to the reference point is chosen as $p^g$. The obtained results have demonstrated the larger the $Np^g$ value is, the better the convergence is.

♦ **Wickramasinghe and Li (**2008**) work**

The authors hybridized the Non-dominated Sorting PSO (NSPSO) (Li 2003) and the MaxiMin PSO (MMPSO) (Li 2004) with the reference point method. The NSPSO has the same sorting and diversity preserving mechanisms of NSGA-II. In fact, from a population of size *N*, an offspring population of size *N* is created. After that, a non-dominated sorting is performed on the merged population of size 2*N* in order to classify it into several non-dominated fronts. Once non-domination ranks are assigned, the crowding distance assignment is performed frontwise. The leaders $p^g$ are then selected from the set of top 10% least crowded solutions from the first non-dominated front. In MMPSO, the fitness of particle *x* is given by the maximum of the minimum values between *x* and all other particles *y* from the population *P* and is given by:

$$fitness(x) = \max_{m=1,...,M;\, y \in P \setminus \{x\}} \{\min\{f_m(x) - f_m(y)\}\} \tag{3.36}$$

The particles having a fitness values less than zero are considered as non-dominated and constitute the first non-dominated front. The total number of individuals that move to the next generation consists of all individuals in the first non-dominated set. If the non-dominated set size is less than desired, then dominated individuals are chosen randomly to fill the vacant positions. A particle chooses a leader (global best), randomly from the top 10% of least crowded particles in the non-dominated front. In both reference point-based PSO algorithms, the leader selection strategy is modified. In NSPSO, a particle chooses its leader from the potential set corresponding to the top 10% closest particles to the reference point picked from the first non-dominated front. In MMPSO, a particle will choose random values from all dimensions of the set of potential leaders and produce a leader. An $\varepsilon$-clearing mechanism is included in both NSPSO and MMPSO in the same way as R-NSGA-II in order to control the ROI spread. The two PSO algorithms have demonstrated their ability to provide several ROIs near the user-provided reference points with a controlled spread defined by the user-specified $\varepsilon$ parameter. However, NSPSO and MMPSO effectiveness depends on the population size since the potential set size depends on the population size.

♦ **Molina et al. (**2009**) work : the g-dominance**

The authors suggested a new kind of dominance, called *g-dominance*, where solutions satisfying all aspiration levels and solutions fulfilling none of the aspiration levels are preferred over solutions satisfying some aspiration levels. The g-dominance has three merits: (1) it can be incorporated in several metaheuristics, (2) although the preferences are modelled as a reference point, the search process works without using any ASF and (3) it can be used in an interactive way. Formally, assuming $g = (g_1,...,g_M)$ to be a reference point, each solution $x$ is assigned a flag as follows:

$$Flag_g(x) = \begin{cases} 1 & \text{if} \quad f_m(x) \le g_m \ \ \forall\, m = 1,...,M \\ 1 & \text{if} \quad f_m(x) \ge g_m \ \ \forall\, m = 1,...,M \\ 0 & \text{otherwise} \end{cases} \tag{3.37}$$

Given two distinct solutions $x$ and $y$, $x$ is said to g-dominate $y$ if and only if:

$$Flag_g(x) > Flag_g(y) \text{ or } Flag_g(x) = Flag_g(y) \text{ and } f(x) \preceq f(y) \tag{3.38}$$

The main disadvantage of this approach is that it does not preserve the order induced by the Pareto dominance relation. Hence, a dominated solution which satisfies none of the goals may be preferred to a solution that dominates it and which fulfills some of the

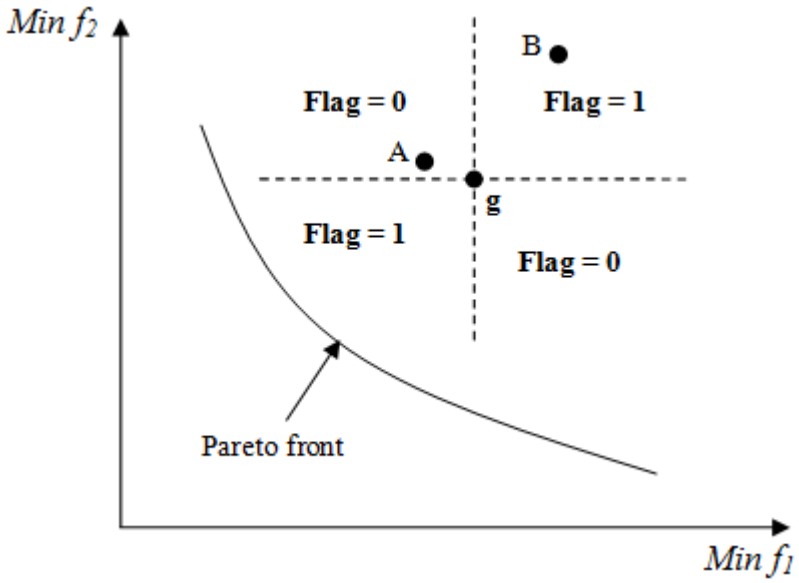



**Figure 3.8 The g-dominance: non-preservation of the Pareto dominance order.**

goals. Figure 3.8 illustrates this case. Solution *A* Pareto dominates *B*, however solution *B* g-dominates solution *A*. This fact discourages convergence towards the Pareto optimal front.

- **Thiele et al. (**2009**) work : the Preference Based-Evolutionary Algorithm (PBEA)**

The authors combined IBEA with the reference point method. The obtained method is called the Preference-based IBEA and denoted PBEA. In IBEA the fitness function of a particular solution *x* is given by:

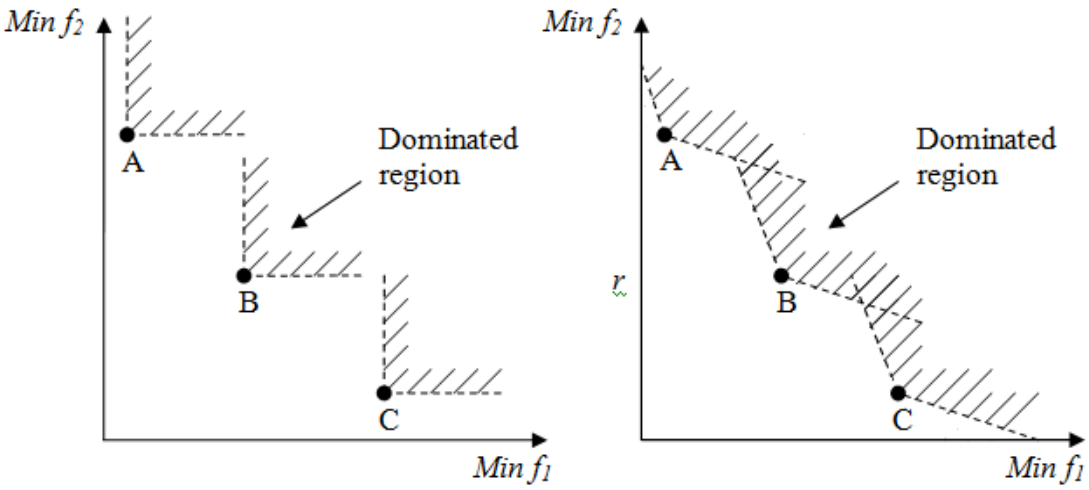
$$fitness(x) = \sum_{y \in P \setminus \{x\}} \left( -e^{-I(y,x)/\kappa} \right) \tag{3.39}$$

where $\kappa$ is a scaling factor (Zitzler and Künzli 2004). The fitness of *x* expresses the loss in quality if solution *x* is removed from the population, i.e., the marginal contribution of *x* in terms of approximation quality. The quality indicator used in IBEA is the additive binary ε-indicator $I_{\varepsilon^+}$ which is a Pareto compliant indicator. The following expression gives another formulation of $I_{\varepsilon^+}$ :

$$I_{\varepsilon^+}(x, y) = \min_{\varepsilon} \{ f_m(x) - \varepsilon \leq f_m(y) \; \forall \, m = 1, \ldots, M \} \tag{3.40}$$

In order to take DM's preferences into account, the authors modified (3.40) as follows:

$$I_p(x, y) = I_{\varepsilon^+}(x, y) / s(g, f(x), \delta_{PBEA}) \tag{3.41}$$




**(a)** **(b)**

**Figure 3.9 Dominated region for: (a) Pareto dominance and (b) trade-off-based dominance.**

where $$s(g, f(x), \delta_{PBEA}) = s(g, f(x)) + \delta_{PBEA} - \min_{y \in P}\{s(g, f(y))\} \quad (3.42)$$

$\delta_{PBEA}$ is called the specificity parameter and allows the DM to control the spread of the obtained ROI and $s(g, f(x))$ is the ASF of solution $x$. The main advantages of this approach are that the used $I_p(x, y)$ binary quality indicator is Pareto dominance preserving and the approach can be used interactively with more than one reference point. However, the authors noted that adjusting the specificity parameter is not an easy task and such topic is for further investigation (Thiele et al. 2009).

### 3.2.5 Trade-off-based approaches

♦ **Branke et al. (**2001**) work: the Guided MOEA (G-MOEA)**

The authors proposed a variant of the Pareto dominance relation that focuses the search towards the preferred part of the front based on trade-off information provided by the DM. In fact, the DM is invited to provide, for each pair of objectives, maximally acceptable trade-offs. For example, for the bi-objective case, the DM could specify that an improvement by one unit in objective $f_2$ is worth a degradation of objective $f_1$ by at most $c_{12}$ unit. Similarly, a gain in objective $f_1$ by one unit is worth a degradation of objective $f_2$ by at most $c_{21}$ unit. This trade-off information is then used to modify the dominance relation as follows:

$$x \preceq_t y \Leftrightarrow \big(f_1(x) + c_{12} f_2(x) \le f_1(y) + c_{12} f_2(y)\big) \wedge \big(c_{21} f_1(x) + f_2(x) \le c_{21} f_1(y) + f_2(y)\big) \quad (3.43)$$

where $x \preceq_t y$ means that $x$ is preferred to $y$ based on the designed trade-off-based

dominance relation. Figure 3.9 shows the effect produced by the new dominance relation in the objective space. In fact, when compared to the original dominance relation, a particular solution now dominates a larger region. With this dominance scheme, only a portion of the Pareto optimal front remains non-dominated. This portion corresponds to the ROI. However, since this approach implicitly assumes linear utility functions, it may not be possible for G-MOEA to focus on all parts of concave Pareto optimal front (Branke et al. 2001). Moreover, this approach is restricted to the bi-objective case.

### 3.2.6 Outranking-based approaches

- **Fernandez et al. (**2010**) work: the Non-Outranking Sorting Genetic Algorithm (NOSGA)**

The authors exploited the outranking concept (Roy 1996) in order to integrate DM's preferences in NSGA-II. For each objective function $f_m$, a relational system of preferences ($Pr_m$, $I_m$) is designed where *Pr* means preference and *I* means indifference. For each objective vector component pair ($f_m(x)$, $f_m(y)$), one and only one of the three following statements holds:

- $f_m(x)\ Pr\ f_m(y)$,
- $f_m(y)\ Pr\ f_m(x)$, or
- $f_m(x)\ I\ f_m(y)$.

This formulation allows indifference thresholds in order to model some kinds of imprecise one-dimensional preferences. It should be noticed that the considered relational system of preferences is more general than the usual formulations which consider only true criteria (i.e., $f_m(x) \neq f_m(y)$ implies non-indifference). Without loss of generality, the following is supposed:

$$f_m(x)\ Pr\ f_m(y) \Rightarrow f_m(x) > f_m(y) \tag{3.44}$$

For each pair ($f(x)$, $f(y)$), the DM, assisted by the decision analyst, creates a fuzzy predicate modelling the truth degree of the predicate "$f(x)$ is at least as good as $f(y)$". The authors adopted the outranking approach based on ELECTRE methods (Roy 1990, Mousseau and Dias 2004). Assuming $u = f(x)$ and $v = f(y)$, the proposition "*u* outranks *v*" which means "*u* seems at least as good as *v*" holds if and only if the coalition of criteria in agreement with this proposition is strong enough and there is no important coalition discordant with it. It can be expressed by the following logical equivalence:

$$u\ S\ v \ \Leftrightarrow\ C(u,v) \wedge \neg D(u,v) \tag{3.45}$$

where $C(u, v)$ is the predicate about the strength of the concordance coalition and

$D(u, v)$ is the predicate about the strength of the discordance coalition.

The NOSGA works with non-strictly outranked solutions instead of non-dominated ones. The selection process is similar to NSGA-II one but it extracts non-strictly outranked individuals in order to classify the population into different non-strictly outranked fronts. Since the MOEA searches for the ROI and not an approximation of the whole Pareto front, the crowding distance criterion is replaced by a weakness measure *W* (Fernandez et al. 2010). The NOSGA performance assessment was done on several instances of four- to nine-objective knapsack problem. The NOSGA has demonstrated its superiority over NSGA-II in providing non-outranked solutions in a privileged zone of the Pareto front. Moreover, the NOSGA is shown to be less sensitive to the increase of the number of objectives than NSGA-II. The main crux of this study is that the NOSGA was not confronted to any preference-based MOEA. Such confrontation is required for more validation.

- **Fernandez et al. (**2011**) work: the Non-Outranking Sorting Genetic Algorithm II (NOSGA-II)**

In a more recent study (Fernandez et al. 2011), the same authors proposed an enhanced version of NOSGA, called NOSGA-II, which increases the selective pressure towards the preferred solutions. This is achieved by considering other binary preference relations (Roy 1996) in the preferential system in addition to the strict preference relation and the indifference one. Assuming $u = f(x)$ and $v = f(y)$, these relations are the following:

1) *Weak preference*: It corresponds to the existence of clear and positive reasons in favor of *u* over *v*, but that are not sufficient to justify strict preference.
2) *Incomparability*: None of the situations of indifference, strict preference nor weak preference predominates. That is, the absence of clear and positive reasons that justify any of these relations.
3) *K-preference*: It corresponds to the existence of clear and positive reasons that justify strict preference in favor of one (identified) of the two solutions or incomparability between the two solutions, but with no significant division established between the situations of strict preference and incomparability.

The authors reported that the NOSGA-II outperforms the NSGA-II and the NOSGA on a real world instance of the multi-objective knapsack problem.

### 3.2.7 Desirability function-based approaches

- **Wagner and Trautmann (**2010**) work: the Desirability Function-based SMS-EMOA (DF-SMS-EMOA)**

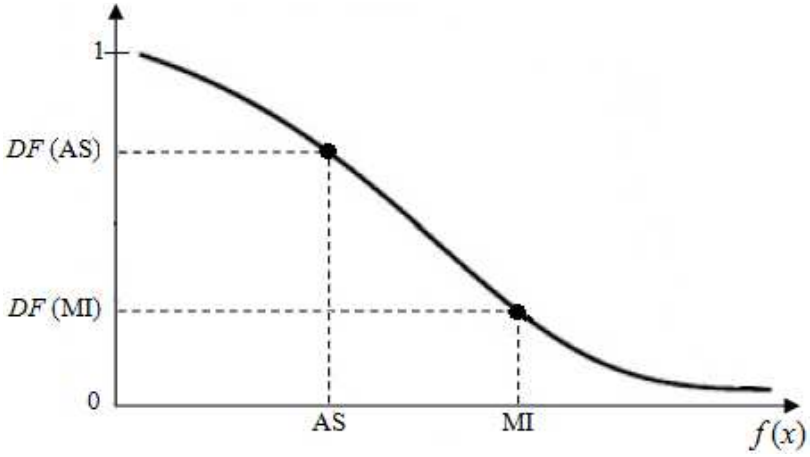



**Figure 3.10 A realization of Harrington's DF based on DM's preferences: (AS, DF(AS)) and (MI, DF(MI)).**

The authors proposed a preference-based version of the SMS-EMOA. We recall that SMS-EMOA belongs to the third generation of MOEAs since it combines non-dominated sorting with hypervolume-based selection. In fact, in each generation, the last (worst) considered non-dominated front is pruned by removing the individual having the least contribution in terms of hypervolume. For each objective function, the DM is invited to express his/her preferences by supplying two thresholds which represent: (1) an Absolutely Satisfying objective value (AS) and (2) a Marginally Infeasible objective value (MI). These thresholds serve as parameters to the DF of the corresponding objective function. The concept of desirability was introduced by Harrington (1965) in the context of multi-objective industrial quality control. DFs map the values of the objectives to disirablities, i.e., values on a unitless scale in the domain [0,1]. The mapping is based on preference information regarding exemplary objective values (i.e., the two thresholds AS and MI). We note that in addition to supplying AS and MI, the DM specifies a desirability value for each threshold in the domain [0,1]. The preferences are specified under the assumption that the smaller the difference between the actual desirability and the maximum value of one is, the better the quality of the solution in the corresponding objective is. In general, any *DF*: $f(x) \rightarrow [0,1]$ describing the desirability of different regions in the objective space can be defined as a DF. Figure 3.10 presents an example of a realization of Harrington's DF (Harrington 1965). The main idea of DF-SMS-EMOA is to convert the objective function of the original MOP into DFs and then optimizing these DFs instead of the original objectives. We recall that disirablities are to be maximized. The DF-SMS-EMOA has demonstrated its ability to bias the search towards the DM's preferred region on the bi-objective ZDT test functions and the five-objective turning process problem (Biermann et al. 2008). However, the authors noted that, when the number of objectives increases,

the number of border solutions outside the specifications limits of DFs increases and the hypervolume computational complexity also increases. Consequently, further research effort is required to study the combinations between MOEAs and DFs for many-objective optimization problems (i.e., MOPs involving more than 3 objective functions (Hughes 2005)).

## 3.3 Discussion

### 3.3.1 From MOEAs to preference-based MOEAs

If a single solution is to be selected from the Pareto front of a MOP at some point during the optimization process, the DM has to reveal his/her preferences. Specifying these preferences a priori, i.e., before alternatives are known, often means to ask too much from the DM. On the other hand, searching for all non-dominated solutions, as most MOEAs do, may result in: (1) wasting computational efforts to find solutions that are clearly undesired by the DM and (2) complicating the DM's task when selecting the final preferred alternative from a huge set of solutions. In the previous section, we presented a review of the most prominent preference-based MOEAs which allow avoiding the two above cited shortcomings of general-purpose MOEAs. Table 3.1 presents a synthetic comparison of Preference-based MOEAs. The works are classified based on the type of the DM's preferences and are listed in a chronological order for each category in order to illustrate the evolution scheme of each class of algorithms. We remark that most of these works are published after 2006. Additionally, we see that from 2008, there is an increasing emphasis on the topic of including DM's preferences in EMO. Table 3.1 lists several comparison criteria that are classified into two main classes: (1) general criteria that are discussed in this subsection and (2) pros and cons criteria that are discussed in section next. The general criteria are: (1) *modification* which indicates the modified part of the MOEA and *influence* which indicates whether the result is a bounded region of the Pareto optimal front or a just a biased distribution.

According to the algorithmic details of the different search methods provided in the second section, we remark that most preference-based MOEAs are modified versions of general-purpose MOEAs. This observation is emphasized by the column *modification* in table 3.1, where we see that the most frequently modified part of the MOEA is the dominance relation. In fact, several preference-based dominance relations were proposed such as the g-dominance, the trade-off-based dominance, and so fourth. When the Pareto dominance is replaced by such dominance relations, the search process is guided towards the ROI according to the DM's preferences. Based on the column *influence*, we see that most preference-based MOEAs aim at providing a bounded ROI

**Table 3.1 Comparison of preference-based MOEAs (inspired by (**Branke 2008**)): MROI means Multiple ROIs, SC means Spread Control, PDP means Pareto Dominance Preservation, DP means Diversity Problems and SCAD means SCAlability Demonstration with respect to the number of objectives. For pros and cons criteria, "Y" means Yes and "N" means No.**

| | **General criteria** | | **Pros & Cons criteria** | | | | |
|---|---|---|---|---|---|---|---|
| **Reference** | **Modification** | **Influence** | **MROI** | **SC** | **PDP** | **DP** | **SCAD** |
| **Weights** | | | | | | | |
| (Deb 1999) | Crowding operator | Distribution | N | N | Y | N | N |
| (Branke and Deb 2004) | Crowding operator | Distribution | N | Y | Y | N | Y |
| (Zitzler et al. 2007) | Quality indicator | Distribution | N | N | Y | N | Y |
| **Ranking some candidate solutions** | | | | | | | |
| (Greenwood et al. 1997) | Dominance | Region | N | N | Y | N | N |
| (Deb et al. 2010) | Dominance | Region | N | N | Y | N | Y |
| (Köksalan and Karahan 2010a) | Dominance | Distribution | N | Y | Y | N | N |
| (Battiti and Passerini 2010) | Crowding operator | Region | N | N | Y | Y | Y |
| (Fowler et al. 2010) | Dominance | Region | N | N | Y | N | Y |
| (Branke et al. 2010) | Dominance + crowding distance | Distribution | N | N | Y | N | N |
| **Ranking objectives** | | | | | | | |
| (Jin and Sendhoff 2002) | Objectives aggregation | Distribution | N | N | N | Y | N |
| (Cvetkovic and Parmee 2002) | Dominance | Distribution | N | Y | N | N | N |
| (Rachmawati and Srinivasan 2010) | 1) Dominance<br>2) Solution sorting mechanism<br>3) Crowding operator | Region<br>Region<br>Distribution | N<br>N<br>N | N<br>N<br>N | N<br>Y<br>Y | N<br>N<br>N | Y<br>Y<br>Y |
| **Reference point** | | | | | | | |
| (Fonseca and Fleming 1993) | Dominance | Region | N | N | Y | N | N |
| (Tan et al. 1999, 2003) | Dominance | Region | Y | N | N | N | Y |
| (Deb et al. 2006) | Crowding operator | Region | Y | Y | Y | Y | Y |
| (Allmendinger et al. 2008) | Leader selection strategy | Region | Y | Y | Y | Y | N |
| (Wickramasinghe and Li 2008) | Leader selection strategy | Region | Y | Y | Y | Y | N |
| (Molina et al. 2009) | Dominance | Region | Y | N | N | Y | N |
| (Thiele et al. 2009) | Quality indicator | Region | Y | Y | Y | N | Y |
| **Reference direction (reference point + reservation point)** | | | | | | | |
| (Deb et al. 2007a) | Solution sorting mechanism | Region | Y | Y | Y | Y | Y |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Reference direction and some preference thresholds** | | | | | | | |
| (Deb et al. 2007b) | Crowding operator | Region | Y | Y | Y | Y | Y |
| **Trade-offs between objectives** | | | | | | | |
| (Branke et al. 2001) | Objective functions | Region | N | N | Y | Y | N |
| **Outranking parameters** | | | | | | | |
| (Fernandez et al. 2010) | Dominance | Region | N | N | Y | N | Y |
| (Fernandez et al. 2011) | Dominance | Region | N | N | Y | N | Y |
| **Desirability thresholds** | | | | | | | |
| (Wagner and Trautmann 2010) | Objective functions | Region | Y | N | Y | N | Y |

rather than a biased distribution of non-dominated solutions. This fact provides the DM only with preferred solutions.

### 3.3.2 Preference modeling tools

From the preference modelling tools cited in this table, it is difficult for the DM to precisely state his/her preferences in a priori way, e.g., how could the DM specify the aspiration/reservation levels while he/she ignores the range of each objective function? A simple way to handle this difficulty is to run the MOEA for some small number of generations and then provide the DM with some solutions such as the ideal point and the nadir one. In this way, the DM builds an idea about the ranges of the different objectives which facilitates the task of supplying aspiration/reservation levels. This fact has motivated researchers to design some EMO-based techniques for estimating the nadir point which plays a crucial role in the discovery of objective ranges (Deb and Miettinen 2008; Bechikh et al. 2010b). Such techniques have demonstrated their ability in finding near nadir point quickly and reliably on high dimension MOPs. When modelling preferences as weights, it is difficult to control the guidance of the search towards the ROI. In fact, with the increase of the number of objectives, it is difficult to verify whether the MOEA's provided approximation really replies to the DM's specified weights. For this reason, using weights in an interactive manner is not really so attractive. Modelling preferences as trade-offs between objectives is a complicated task especially when the number of objectives increases. Consequently, using such approach interactively augments the demanded effort from the DM. Ranking a sample of solutions seems to be an interesting way to elicit DM's preferences. However, how to select solutions to build such a sample is still an open question for further research. For example, in the BC-EMOA (Battiti and Passerini 2010), the authors noted that the evolutionary process should be run for a certain number of generations, that is of the same order of a plain MOEA run on the same problem, in order to ensure a reasonable coverage of the whole Pareto front and eventually not to miss portions containing some

possibly preferred solutions. This fact makes BC-EMOA behave like a general purpose MOEA where the entire Pareto front is firstly approximated and then the DM's preferences are used to select the final alternative to realize. Consequently, we do not really see the advantages of articulating DM's preferences within the MOEA. Furthermore, the computational complexity is increased. Using outranking relations in a priori way seems to be interesting. However, the DM should be assisted by the decision analyst (an expert) to set appropriate outranking parameters. Moreover, updating such parameters interactively during the MOEA run is not a trivial task and may augment the DM's burden. Objective ranking is also an interesting way to model DM's preferences. However, the preference update mechanism should be controlled in order to ensure preference consistency as noted by (Rachmawati and Srinivasan 2010). Desirability thresholds seem to allow a straightforward specification of the DM's preferences. We note that the concepts of desirability thresholds and aspiration/reservation levels seem to be so similar.

Among all the used preference modelling tools, it seems to be that the most natural and precise way to express DM's preferences is the reference point (Bechikh et al. 2012a) (e.g., the DM would like to achieve 20 units in the first objective and 15 units in the second objective). In this way, DM's preferences could guide the search towards the ROI precisely and interactively without demanding a great effort from the DM even if the number of objectives $M$ increases. Indeed, the reference point could be drawn on the same plot of the MOEA's population whatever is the number of objectives $M$ (by using the 2D/3D plots for the two-/three-dimensional cases and the parallel coordinates plots for higher dimension cases). This fact facilitates not only the verification of the guidance of the population but also the update of the reference point. These statements are emphasized by the results presented in table 3.1 since the reference point is the most used DM's preference information structure.

### 3.3.3 Pros and cons

Table 3.1 lists a set of criteria that allow illustrating the pros and cons of the different approaches:

—*MROI*: "Y" value means that the algorithm offers the DM the ability to obtain more than one ROI. Usually, at the beginning of the evolutionary process, the DM does not have any idea about the search space. This fact can make the DM doubtful when expressing his/her preferences. Hence, with the option MROI, the DM can guide the search towards multiple ROIs and then he/she focuses the MOEA's population on the final desired ROI during the interactive run. The MROI option represents an advantage for the MOEA since it allows the DM to learn about the search space and consequently

about his/her preferences during the interactive optimization process which facilitates the task of preference updating and adjusting.

—*SC*: "Y" value means that the algorithm allows the DM to control spread of the obtained ROI. The option SC represents an advantage for the MOEA because if the algorithm does allow controlling the ROI breadth, the obtained results can be ambiguous for the DM. Since MOEAs are stochastic search algorithms, if the ROI breadth is not controlled, the DM can obtain different ROIs' spreads in each run of the algorithm. This fact represents a difficulty to the DM when selecting his/her final alternative to realize. Additionally, in this case, the focus of the population on the desired region heavily depends on the termination criterion since the population is guided gradually towards a particular region in the search space from generation to another which may cause a diversity problem especially with the increase of the number of objectives.

—*PDP*: "Y" value means that the preference-based guidance mechanism of the related MOEA preserves the order induced by the Pareto dominance relation, i.e., a dominated solution with respect to the Pareto dominance relation cannot be preferred to a solution that dominates it. This fact preserves elitism (Deb 2001). In fact, the PDP criterion has been mentioned in the survey of Coello (2000). If the preference-based MOEA allows contradicting the Pareto dominance order, then serious convergence problems can occur.

—*DP*: "Y" value means that the algorithm has some diversity problems; which represents an inconvenient for the MOEA. In fact, guiding the search process towards a particular region of the search space at the beginning of the search process may cause a reduction in the population phenotypic diversity. For this reason, when designing a preference-based MOEA, it is of particular interest to conceive a diversity mechanism that allows preserving the population diversity.

—*SCAD*: "Y" value means that the scalability of the proposed approach with respect to the number of objectives is demonstrated in the original paper of the algorithm. The emerging field of many-objective optimization (Hughes 2005) has recently attracted a lot of researchers. One of the proposed approaches to handle such type of problems is to incorporate DM's preferences in the evolutionary process in order to explore only the desired portion of the Pareto front. With the increase of the number of objectives, the Pareto dominance becomes ineffective when comparing between solutions. Consequently, researchers have used DM's preferences as an additional criterion to distinguish between the population individuals and focus the search towards the optimal ROI of a many-objective problem.

## 3.4 Group preference handling

### 3.4.1 Group preference handling in EMO

In the EMO community, the problematic of incorporating the preferences of a group of DMs in MOEAs is ignored by most researchers. (Pfeiffer et al. 2008) is probably the unique work that has considered such a problematic by proposing four variants of the previously discussed algorithm R-NSGA-II. According to the authors, while the *OR* operator evolves the population towards either of the reference points, the *AND* operator tries to minimize the deviation from all goals concurrently and therefore attempts to find a consensus. The four variants are described and classified into two categories as follows:

- **Ranking-based approach**

Instead of using the minimum of the ranks (in reference to the OR operator), the maximum of the ranks is taken as crowding distance (in reference to the AND operator). Since in contrast to the original NSGA-II, the crowding distance is supposed to be minimized, this variant is called the *MinMaxRanking* approach. The second idea is to assign the average rank as crowding distance. This variant is named *AvgRanking* approach. For example, if one solution has the ranks 1, 4 and 10 to each of the three reference points respectively, the crowding distance would be 5.

- **Distance-based approach**

In this approach, only the normalized Euclidean distances directly are used as crowding distances. The crowding distance is assigned as either the maximal normalized Euclidean distance they have to a reference point (*MinMaxDistance* approach) or the average distance (*AvgDistance* approach).

These four variants of the R-NSGA-II algorithm were assessed on three ZDT test problems (i.e., ZDT1, ZDT2 and ZDT3 (cf. appendix A)) and two realistic instances of the flow shop scheduling problem. The obtained results show that the proposed approaches are able to provide the desired average ROI which is considered to be composed with consensus solutions according to the authors. However, from our point of view, the fact of providing an average ROI does not resolve the problem since several DMs are still dissatisfied with the obtained results and the conflict is still existing. In fact, consensus reaching is a complex process and requires more elaborated mechanisms for preference aggregation (Herrera-Viedma et al. 2007). For this reason, we present in the next section a brief presentation of *social choice theory* (Arrow 1951) and the main difficulties encountered when aggregating the preferences of a set of DMs especially: (1) the impossibility to achieve all kinds of fairness and (2) manipulation.

### 3.4.2 Social choice theory: A brief review

Social choice theory is a theoretical framework for measuring individual interests, values or welfares as an aggregate towards a collective decision (Arrow 1951). The problematic of social choice can be described as follows. Assuming that we have a set of agents having preferences over a set of alternatives, the issue is how to design a mechanism that outputs a social preference or a single winner over the set of alternatives. Voting methods, such as plurality method, Borda method and Condorcet method are considered as examples of social mechanisms (Bouyssou et al. 2009). The most known and influential result within the social choice community is *Arrow's impossibility theorem* which states that *fairness* is *multifaceted* and that it is impossible to achieve all of these kinds of fairness simultaneously (Shoham and Leyton-Brown 2009). According to Arrow (1951), the multifaceted fairness criteria are the following:

1) *Unrestricted domain (or the universality criterion)*: Each agent is allowed to rank the set of alternatives in any order without any a priori constraint and the social choice function must generate a collective preference order from any logically possible set of individual preference orders;
2) *Transitivity*: Assuming three alternatives *a*, *b* and *c*, if each agent prefers *a* to *b* and *b* to *c* then *a* should be preferred to *c* in the collective preferences;
3) *Pareto efficiency*: Whenever all individuals prefer an alternative *a* to another *b*, *a* must be preferred to *b* in the collective preferences;
4) *Independence of irrelevant Alternatives*: the collective preference order of any pair of alternatives *a* and *b* should depend solely on the individuals' preferences between these alternatives and not on their preferences for other (irrelevant) alternatives; and
5) *Non-dictatorship*: the collective preferences should not invariably correspond to the preferences of any single individual, regardless of the preferences of the others.

The Arrow's Impossibility Theorem is given as follows:

**Definition 3.3: Arrow's impossibility theorem**

When there are more than two alternatives, any social choice function that satisfies Pareto efficiency and independence of alternatives necessarily violates non-dictatorship and is therefore *dictatorial*.

Another important issue in social choice theory, that has received considerable attention, is *manipulation* (Conitzer 2006). By definition, a manipulation if a way of misreporting one's true preferences that leads to a better result for oneself. Another impossibility result in social choice theory is stated by the *Gibbard's impossibility theorem* which is given as follows (Gibbard 1973):

**Definition 3.4: Gibbard's impossibility theorem**
When there are more than two alternatives, any social choice function that: (1) satisfies non-dictatorship and (2) is *onto* (onto means that every alternative can be a winner under some preference profile) is *manipulable*.

We can say that the two aforementioned impossibility results represent the main challenges when designing a social choice mechanism which makes social choice a very active research area till today (Chevaleyre et al. 2007).

## 3.5 Conclusion

This chapter provided a survey of preference-based MOEAs. These algorithms are mostly modified versions of general-purpose MOEAs. Different parts of the MOEAs can be modified in order to direct the search towards the preferred part(s) of the Pareto front. The DM's preference information structure can have different forms. The reference point seems to be a promising way to include DM's preferences within MOEAs. Reference point-based MOEAs are characterized by their abilities to: (1) provide a Pareto optimal bounded ROI, (2) control the breadth of the obtained ROI and (3) provide more than one ROIs. Additionally, such algorithms can be executed interactively without increasing the DM's burden and permits the DM to verify in a straightforward manner if the obtained ROI really corresponds to his/her preferences, i.e., whether the obtained Pareto-optimal solutions are concentrated around the DM's reference point. Most of the proposed preference-based MOEAs consider the case of a single DM. For this reason, in section 3.4, we have presented the work of Pfeiffer et al. (2008) that is probably the unique work in the EMO community that has considered the case of multiple DMs. Additionally, we have introduced the main difficulties faced when aggregating a set of DMs' preferences by providing a brief review of social choice theory.

# Chapter 4

# Implicit Preference-based Evolutionary Multi-objective Optimization

## 4.1 Introduction

In the absence of explicit DM's preference information, there exist special points of the Pareto front that could represent implicitly preferred parts of the Pareto front for the DM. These parts are: (1) knee regions and (2) nadir point. Knee regions are potential parts of the Pareto front presenting the maximal trade-offs between the MOP's objectives. Solutions residing in knee regions are characterized by the fact that a small improvement in either objective will cause a large deterioration in at least another one which makes moving in either direction not attractive. Such characteristic renders almost always knee solutions of particular interest in practical context. Nadir point is the vector composed with the worst objective values at the Pareto optimality stage. For this reason, this special point represents very important information for the DM. Figure 4.1 illustrates the concepts of knee regions and nadir point. This chapter is devoted to review the different approaches for integrating implicit DM's preferences in MOEAs. The next section presents the existing approaches for knee region approximation. The third section reviews the proposed approaches for nadir point estimation. The last section concludes the chapter.

## 4.2 Implicit DM's preferences as knee regions

### 4.2.1 Motivations for knee region approximation

Das (1999) noticed that the trade-off level is varying across the Pareto front and there are solutions corresponding to maximal trade-off levels. In fact, Das noted that "*from practical experience . . . the user or designer usually picks a point in the middle of the surface . . . where the Pareto surface bulges out the most.*". Such solutions are called "*knee*" solutions and they are very interesting to the DM since they constitute the optima in terms of trade-off. Knee regions are constituted with solutions having the

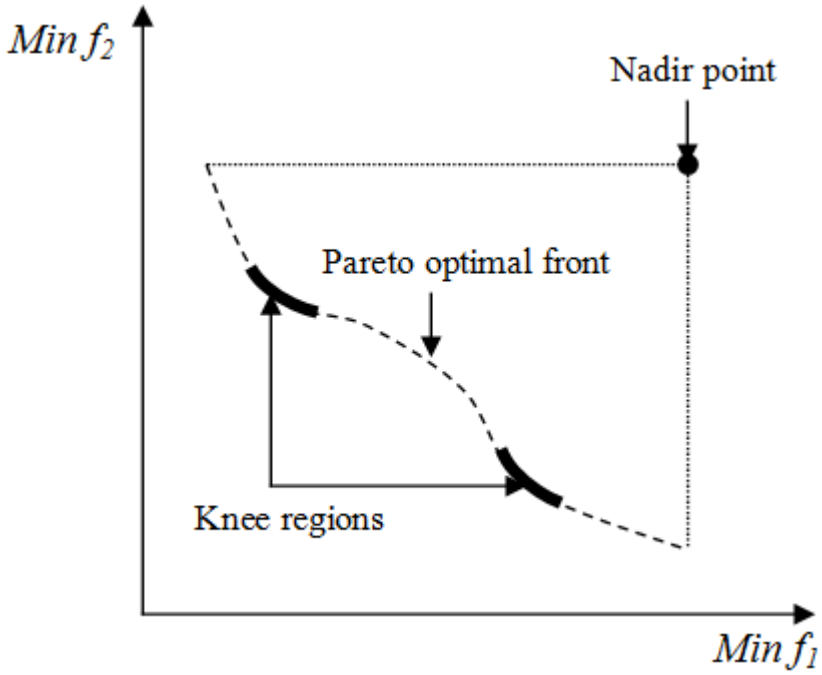



**Figure 4.1 Illustration of implicit DM's preferences: (1) knee regions and (2) nadir point.**

highest values in terms of the marginal rate of return. Substitution of a given non-knee Pareto optimal solution with another solution on the knee region yields the largest improvement per unit degradation. Such characteristic renders knee regions so important to the DM. Few studies were interested in discovering knee regions compared to the general-purpose MOEAs' literature. In the next subsection, we give a brief review of these works.

### 4.2.2 Existing methods for knee region approximation

The focus on knee regions is not new. Das (1999) proposed a method based on the Normal-Boundary Intersection (NBI) (Das and Dennis 1998) to locate the knee of the Pareto front. Das characterized the knee solution in terms of the Convex Hull of Individual Minima (CHIM), which was defined as the set of points in $\Re^n$ that are convex combinations of $F^* - F(X^*)$, where $F^*$ is the utopia vector of the global minima of objectives $F = [f_1, f_2, ..., f_M]$ and $F(X_m^*)$ is the objective vector of the global minimizer of $f_m$, $X_m^*$. The knee solution corresponds to the farthest solution from the CHIM. The non-linear programming method of Das maximizes the distance to the CHIM in order to get the knee solution. The technique requires an a priori estimate of $F(X_m^*)$ for each objective (i.e., extreme solutions) and the accuracy of the obtained knee solution is directly related to these estimates. In general, providing an accurate a

priori estimate of the CHIM is not an easy task. It is even more difficult when the Pareto front contains multiple knee regions. As each knee solution is associated with its corresponding CHIM, finding all knee solutions necessitates the estimation of several CHIMs and the discovery of each farthest point in terms of the distance to each CHIM. Extensions to cater the problem with multiple knee solutions with non-uniform geometry were not provided either in Das's work (Das 1999). Additionally, the NBI furnishes only one unique knee solution. However, in practice, it would be desirable to obtain a set of solutions in the vicinity of each knee center which facilitates the DM's task when selecting the final preferred alternative. Branke et al. (2004) designed a MOEA which does not require a priori information about the extreme solutions. In fact, they proposed two modification strategies to the NSGA-II (Deb et al. 2002a) to make it focusing on knee regions. The authors modify the diversity selection criterion of NSGA-II. In the first strategy, the algorithm utilizes a geometrical property of knee regions, i.e., the external angle formed by a certain solution and its neighboring non-dominated solutions is larger for solutions lying on the knee region than for those otherwise located. Hence, the NSGA-II crowding factor was replaced by an angle-based measure and the focus on knee regions were achieved by maximization of this measure. This approach has been shown to be effective for the bi-objective case. However, it is not amenable to higher dimension problems. In the second strategy, the authors suggested a marginal utility function to approximate the angle-based measure in the case of more than two objectives. In fact, the larger the external angle between a solution and its neighbors is, the larger the gain in terms of linear utility obtained from substituting the neighbors with the solution of interest is. The substitution of the NSGA-II diversity factor by this linear utility measure forces the algorithm to concentrate the search on knee regions and hence discovering knee regions for problems involving more than two objectives. However, the reliance of the proposed utility function on weighted sums precludes the convergence of the algorithm to the non-convex parts of the Pareto front which may lead to the loss of less pronounced knee regions. It is worth noting that the issue of controlling the extent of knee regions was not addressed in both methods (Branke et al. 2004). Rachmawati and Srinivasan (2006a, 2006b) developed a MOEA focusing on knees. The proposed MOEA computes a transformation of the original objectives based on weighted sums. The transformed functions identify niches which correspond to knees in the objective space. The extent and the density of coverage of the knee regions are controllable by the niche strength and pool size parameters respectively. Although based on weighted sums, the algorithm is capable to find out solutions in the non-convex regions of the Pareto-front. In fact, $Q$ weighted sums of objectives are computed with a set of $Q$ uniformly distributed weight

sets. These weighted sums are then sorted according to magnitude to yield $Q$ rank figures for each individual. The highest $P$ rank figures constitute objective values to be optimized for an individual. $P$ and $Q$ are user-defined positive integer parameters where $P < Q$. A biasing selection criterion formulated on this strategy encourages the survival of the global optima of $P$ subsets of the $Q$ weighted sums. The strategy is susceptible to the loss of less pronounced knee regions which constitute local Pareto optima in the aggregation computed with the weight sets describing their respective CHIMs but are not part of the global optima in the very same weighted sums. The authors argued that while a judicious choice of the parameter $P$ and a large $Q$ minimizes the loss of less pronounced knee regions, there are special cases where some knee solutions cannot be found. Another disadvantage of this approach (Rachmawati and Srinivasan 2006b) is that the efficacy of controlling the spread of the knee regions depends heavily on the geometrical shape of the Pareto front which may mislead the DM. Schütze et al. (2008) suggested two updating strategies which can be integrated in stochastic search algorithms for the approximation of knee regions. The advantage of the proposed strategies is that they can be used either as standalone algorithms together with any stochastic search procedure or integrated into any other archiving strategy without causing additional function calls. Additionally, the knee region extent can be controlled by a user-provided parameter. However, this study was restricted to the bi-objective case. Recently, Rachmawati and Srinivasan (2009) proposed a fitness scheme that applies preference-based selection pressure in a MOEA to obtain solutions concentrated on knee regions. This scheme can be considered as an extension of their preceding work (Rachmawati and Srinivasan 2006b). The strategy may be easily incorporated into any MOEA framework with Pareto-based ranking in the selection of solutions. The proposed method is a two-step algorithm. In the first step, the MOEA seeks a rough approximation of the Pareto front, and in the second step, the linear weighted sums of the original objective functions are optimized to guide solutions towards the knee regions. A heuristic was introduced to compute the appropriate weights for each potential knee region in the front approximation. A mechanism to control the extent of focus on the knee region was also provided via the user-supplied parameter $\delta'$. Although the approach relies on weighted sums, solutions on the non-convex region of the Pareto front can also be retained once discovered given an enough large value of $\delta'$. The preference-based fitness introduces little added computational cost, as the number of objective functions to be optimized in the second stage is the same as the number of original objective functions. The approach has been successfully applied on several two- and three-objective knee-based test problems. The main disadvantages of this approach are: (1) the obtained results heavily depend on the quality of the weight sets

found during the first step of the optimization process and (2) the control of the extent of the knee regions depends on the geometrical form of the optimal knee which may mislead the DM.

## 4.3 Implicit DM's preferences as nadir point

### 4.3.1 Motivations for nadir point estimation

Nadir point estimation is an age-old important task in multi-objective optimization (Deb and Miettinen 2008). Unlike the ideal point which can be found by minimizing each objective individually over the feasible search space, the nadir point is constructed from worst objective values in the Pareto optimal front. Hence, the Pareto optimality is a necessary condition to the exact determination of the nadir criterion values which makes the estimation of the nadir point a difficult task especially when the number of objectives increases. Researchers have been interested in finding the nadir point since the early seventies. However, there is still a lack of methods to estimate the nadir objective vector with the desired accuracy for problems involving more than three objectives. Hence, the question that arises here is "*why estimating the nadir point?*". The nadir point is reported to represent important information not only for the DM but also for multi-objective optimization researchers. In fact, along with the ideal point, the nadir point can be used to normalize the objective space which helps multi-objective methods to be applied more reliably to problems involving non commensurable objective functions. It was reported that normalizing the objective space may help in reducing the computational effort by solving the problem faster (Miettinen et al. 2006). Besides, the nadir point and the ideal point help the DM to know the range of the objective functions at the Pareto optimality stage which facilitates the task of preference expression for him/her. Additionally, the nadir point is a pre-requisite for different interactive algorithms such as the GUESS method (Buchanan 1997), the STEP method (Benayoun et al. 1971) and the NIMBUS method (Miettinen and Mäkelä 2006). Moreover, together with the ideal point, the nadir point is crucial for visualizing the optimal Pareto front which facilitates the comparison between solutions especially when using visualization techniques for high dimension problems (e.g., parallel coordinates plot, petal diagrams, bar charts, etc.). Finally, it was reported in (Deb and Miettinen 2008) that the accurate estimation of the nadir point for problems involving more than three criteria is still an open research challenging topic till today. For all these reasons, the nadir point could be considered as a form of implicit DM's preferences.

### 4.3.2 Existing methods for nadir point estimation

In this subsection, we review the proposed approaches in the specialized literature to estimate the nadir point in Multi-Objective Linear Programming (MOLP) and Multi-Objective Non-Linear Programming (MONLP). We enhance this review by drawing a classification scheme for these approaches.

♦ **Exact methods**

Benayoun et al. (1971) used the payoff table to compute the nadir objective values. This involves the minimization of each objective individually over the search space. Then, the payoff table is constructed in such a way the $i^{th}$ row corresponds to values of all other objective functions evaluated at the point where the $i^{th}$ objective obtained its minimum. After that, the maximum (worst) value of the $j^{th}$ column is considered as an estimate of the $j^{th}$ component of the nadir objective vector. The main drawback of this approach is that the payoff table may provide a wrong estimation (overestimation or underestimation) of the nadir point when there is more than one solution having the minimum value for a certain objective and different values for the others. Moreover, this approach faces difficulties when the number of objectives increases. Iserman and Steuer (1988) have demonstrated the difficulties of finding the nadir point when using the payoff table method even for linear problems and have emphasized the need of using a better method. Hence, they suggested three exact approaches to do this task more reliably. The first uses a vector-maximum code to compute all Pareto optimal extreme points. The second resolves a large primal-dual program with non-linear constraints. The third is a simplex-based approach that uses the fact that all Pareto optimal extreme points are connected with paths of efficient edges. The authors concluded that the third approach is the only deterministic approach that can be used in practice. However, they noticed that this approach is computationally expensive. Erghott and Tenfelde-Podehl (2003) suggested an algorithm to compute the nadir objective values exactly in MOLP. This approach is practical only for bi- and tri-objective problems. In fact, its application to an *M*-objective MOP necessitates the discovery of the whole efficient set of each sub-problem optimizing (*M-1*) objectives. Hence, the efficacy of such an approach decreases dramatically with the increase of the number of objectives to optimize. Alves and Costa (2009a) proposed a new method to determine the exact nadir criterion values over the Pareto optimal set in MOLP. The basic idea is to determine for each criterion, assuming the maximization case, the region of the weight space associated with the efficient solutions that have a value in that criterion below the minimum already known (by default, the minimum in the payoff table). If this region is empty, the nadir value is found. Otherwise, a new efficient solution is computed using a weight vector picked from the delimited region and a new iteration is performed. The method is able to find the nadir values in MOLP

problems with any number of objective functions. However the authors argued that the computational cost increases significantly with the increase of the number of objectives. It is worth noting that all these exact methods are restricted to the linear case and thus they cannot handle non-linear problems.

♦ **Heuristic methods**

- **Non-evolutionary approaches**

Dessouky et al. (1986) suggested three heuristic methods for approximating the nadir values. The authors argued that none of these methods guarantee the exact estimation of the nadir point. Moreover, this work is restricted for only linear problems where all objectives and constraints are linear functions of the decision variables. Korhonen et al. (1997) proposed another heuristic method which is based on reference directions. In each iteration, a reference direction is chosen that maximally minimizes the objective under consideration. This process is iterated until the considered objective reaches a local minimum over the non-dominated set. Then, a cutting plane is inserted into the problem and another direction, if one can be found, that maximally minimizes the objective under consideration is employed. Although the method is heuristic, computational experience shows that much better estimation of the nadir criterion values can be obtained than with the use of the payoff table approach. However, this approach is restricted for MOLP problems. Metev and Vassilev (2003) designed a heuristic approach using reference points. The obtained results are better than those obtained by using the payoff table. However, the approach cannot be applied for the non-linear case.

- **Evolutionary approaches**

Several evolutionary approaches for nadir objective values estimation were proposed in the specialized literature. According to the study of Deb and Miettinen (2008), these approaches can be classified into three classes:

  - **Surface-to-nadir class**

Since MOEAs are shown to be effective black-box tools to approximate the whole Pareto front of various MOPs, the simplest-minded idea comes from finding a representative approximation of the optimal Pareto front and then the nadir point is computed from the extreme solution values. This approach was tested by Deb et al. (2006b) using the NSGA-II. It faces essentially two difficulties. On one hand, the algorithm must find the extreme Pareto optimal points accurately. Otherwise, it may provide a wrong estimation of the nadir point. On the other hand, MOEAs have shown to not work well in discovering a well-distributed set of solutions on the whole Pareto

optimal front for many-objective problems (Hughes 2005), thereby making MOEAs difficult to apply in such scenario.

- **Edge-to-nadir class**

Since the nadir point is constructed from only extreme points, it is unnecessary to get intermediate Pareto optimal solutions. The idea is to find only critical edges of the Pareto surface. Critical edges are boundaries responsible for the true estimation of the nadir point. This approach was implemented in Szczepanski and Wierzbicki (2003) by solving $\binom{M}{2}$ bi-objective sub-problems and the nadir point is then constructed from the extreme non-dominated points from all obtained solutions. This approach seems to be less computationally expensive than the surface-to-nadir approach. However, as discussed by Deb et al. (2006b), this approach may present at least three difficulties: (1) the algorithm may not provide the true extreme solutions if the solutions are not well-distributed on the critical edges, (2) solving many bi-objective sub-problems may provide the same boundary (or a part of it) repeatedly, thereby wasting computational effort and (3) such an approach may require to find multi-modal Pareto optimal solutions and may need to employ a lexicographic procedure to find the true extreme Pareto optimal points.

- **Extreme-point-to-nadir class**

It is intuitive to say that finding intermediate solutions on the critical edges does not help to compute the nadir point since the estimation of the nadir values requires only the discovery of the true extreme Pareto optimal points. Recent studies suggested finding only these extreme points. Several recently proposed methods fall into this class. Deb et al. (2006b) proposed two modified versions of the NSGA-II that focus the search on extreme solutions by modifying the diversity criterion. In the first approach, termed *Worst-Crowded NSGA-II* (*WC-NSGA-II*), in every generation, the population members in every non-dominated front of size $N_F$ are first sorted in an ascending order based on each objective (assuming the minimization case) and a rank equal to the position of the solution in the sorted front is assigned. Hence, each individual *i* from the front gets a rank $R_i^m$ from the sorting based on the $m^{th}$ objective. After assigning all ranks based on each one of the *M* objectives, the crowding of individual *i* is equal to the maximum of its assigned ranks, i.e., $d_i = \max\left\{R_i^1, R_i^2, ..., R_i^M\right\}$. The authors reported that this approach faces difficulties in maintaining the population diversity which may not only slow down the search but also encourage the premature convergence phenomenon to occur. Additionally, this approach may provide spurious solutions which are non-dominated with respect to the current extreme solutions but non Pareto

optimal. This type of solutions may cause a wrong estimation of the nadir point. The second approach, termed *Extremized-Crowded NSGA-II* (*EC-NSGA-II*), represents a slight modification of the first one by modifying the rank assignment as follows: $R_i^m = \max\left\{R_i^m, N_F - R_i^m + 1\right\}$. In this way, solutions having best objective values and solutions having worst ones are emphasized. The authors reported that the second approach avoids the difficulties faced by the first approach and provides better performance. More recently, Deb et al. (2009a, 2009b) enhanced the extremized-crowded approach by hybridizing it with a bi-level local search based on the reference point method. The upper-level optimization uses a reference point and a weight vector as decision variables and optimizes the critical objective. The lower-level optimization projects solutions onto the Pareto front by minimising an augmented ASF using the reference point and the weight vector obtained in the upper-level optimization. The authors reported that this approach presents better performance compared to the two other extreme-point-to-nadir approaches. Another evolutionary approach was proposed by Alves and Costa (2009b). This approach uses a population of weight vectors with particular characteristics, which are then used as parameters in the optimization of weighted sums of the objective functions. The population evolves through a process of selection, recombination and mutation. The algorithm has been tested on a number of problems for which the nadir point is known and good results where reported. However, the application of such an approach is restricted to linear problems.

## 4.4 Conclusion

In this chapter, we have provided a review of the most prominent works in incorporating implicit DMs preferences in EMO. Actually, there are two forms of implicit preferences. In one hand, we have knee regions corresponding to the portions of the Pareto front composed with the worthiest solutions in terms of compromise between the different conflicting objectives. The breadth control and the sensitivity to the objective functions' shapes represent difficult challenges for all discussed methods. In the other hand, we have the nadir point which represents important information for the DM since it is composed by the worst values picked from the Pareto optimal front and not from the whole search space. As discussed previously, there are several motivations to estimate such vector. One of them, which is directly related to our thesis topic, is the assistance of the DMs in expressing their preferences in terms of reference points. The main challenges for the discussed methods for nadir point estimation are: (1) the accuracy of the estimation and (2) the computational efficiency.

# Part II
# Contributions

# Chapter 5

# Incorporating Explicit DM's Preferences in Evolutionary Multi-objective Optimization

## 5.1 Introduction

In this chapter, we propose a new dominance relation for preference-based EMO. This dominance relation is called *r-dominance* and is inspired from the reference point method and the classical Pareto dominance relation (Ben Said et al. 2010). The originality of the r-dominance is its ability to create a strict partial order among Pareto-equivalent (non-dominated) solutions. This fact makes our new proposed dominance relation able de guide the search towards the ROI based on DM's explicit preference information expressed as a set of aspiration levels (i.e., a reference point). After integrating the r-dominance in the NSGA-II, the efficacy and the usefulness of the modified algorithm (called "*r-NSGA-II*") are assessed through two- to ten-objective test problems a priori and interactively. Moreover, the proposed approach provides competitive and better results when compared to other recently proposed preference-based EMO approaches. This chapter is structured as follows. The second section details the proposed approach. The third section is devoted to the experimental study. The fourth section concludes this chapter.

## 5.2 Proposed approach

### 5.2.1 The r-dominance: definition and properties

The r-dominance (reference solution-based dominance) takes its origins from the hybridization between the Pareto dominance principle and the reference point method. The key feature of this new dominance relation is to prefer solutions that are closer to the reference point (DM's preferences) while preserving the order induced by the Pareto dominance. In order to determine the closeness of a certain solution to the reference point, an ASF is required. There exist various ASFs in the MCDM literature

(cf. (Miettinen 1999) for a review). We choose to use the weighted Euclidean distance employed by Deb et al. (2006a):

$$Dist(x,g)=\sqrt{\sum_{m=1}^{M} w_m\left(\frac{f_m(x)-g_m}{f_m^{\max}-f_m^{\min}}\right)^2},\ w_m\in[0,1],\ \sum_{m=1}^{M} w_m=1 \tag{5.1}$$

where $x$ is the considered solution, $g_m$ is the $m^{th}$ component of the user-specified reference point $g$, $f_m^{\max}$ is the upper bound of the $m^{th}$ objective values, $f_m^{\min}$ is the lower bound of the $m^{th}$ objective values and $w_m$ is the weight associated with the $m^{th}$ objective. It should be noted that the ASF (5.1) can be used when solving non-convex MOPs (Deb et al. 2006a). Our choice is justified by the fact that the weighted Euclidean distance gathers more information about the closeness of a certain solution to the reference point than the ASF proposed by Wierzbicki (cf. equation (2.7)) especially when the number of objectives increases. For example, given two solutions $X$ = (10, 8, 9, 7) and $Y$ = (10, 2, 3, 4) for a four-objective minimization problem and assuming the reference point (0, 0, 0, 0) and a uniform weight vector, when using the Wierzbicki ASF, $X$ and $Y$ are considered to be equivalent. However, when using the ASF (5.1), the solution $Y$ is closer to the fixed reference point than $X$ which is seen from the objectives values of the two solutions. In the following, we give the definition of the r-dominance relation and we study its main properties.

**Definition 5.1: The r-dominance**

Assuming a population of individuals $P$, a reference point $g$ and a weight vector $w$, a solution $x$ is said to r-dominate a solution $y$ (denoted $x \prec_r y$) if one of the following statements holds true:

1) $x$ dominates $y$ in the Pareto sense; or
2) $x$ and $y$ are Pareto-equivalent and $D(x,y,g)<-\delta$, where $\delta\in[0,1]$ and:

$$D(x,y,g)=\frac{Dist(x,g)-Dist(y,g)}{Dist_{\max}-Dist_{\min}} \tag{5.2}$$

$$Dist_{\max}=Max_{z\in P}\ Dist(z,g) \tag{5.3}$$

$$Dist_{\min}=Min_{z\in P}\ Dist(z,g) \tag{5.4}$$

$\delta$ is termed *the non-r-dominance threshold*.

The main idea behind the r-dominance relation is to create a strict partial order between Pareto-equivalent solutions. Hence, the r-dominance has the ability to differentiate between non-dominated solutions in a partial manner based on the user-supplied

aspiration level vector. This fact not only makes the r-dominance selection pressure "*stronger*" than the Pareto dominance one but also integrates the DM's preferences in the selection process. In order to prove the strict Partial order induced by the r-dominance on the set of Pareto-equivalent solutions, we study the properties of the r-dominance relation as follows. Let *A* be a set containing only Pareto-equivalent solutions and let $\{x, y, z\} \subseteq A$.

**Property 5.1:** The r-dominance is an irreflexive relation on the set of alternatives *A*.

**Proof:** We would like to show that $\forall x \in A, x \not\prec_r x$.

$D(x, x, g) = \dfrac{Dist(x, g) - Dist(x, g)}{Dist_{\max} - Dist_{\min}} = 0$. Thus, the condition "$D(x, y, g) < -\delta$" is not satisfied since $\delta \in [0,1]$. For this reason, $(\prec_r, A)$ is irreflexive.

**Property 5.2:** The r-dominance is an asymmetric relation on the set of alternatives *A*.

**Proof:** We would like to show that if $x \prec_r y$ then $y \not\prec_r x$.

$$x \prec_r y \Leftrightarrow D(x, y, g) = \frac{Dist(x, g) - Dist(y, g)}{Dist_{\max} - Dist_{\min}} < -\delta.$$

$$D(y, x, g) = \frac{Dist(y, g) - Dist(x, g)}{Dist_{\max} - Dist_{\min}} = -D(x, y, g).$$

Since $D(x, y, g) < -\delta$, then $D(y, x, g) > \delta$. Hence, if $x \prec_r y$ then $y \not\prec_r x$. For this reason, $(\prec_r, A)$ is asymmetric.

**Property 5.3:** The r-dominance is a transitive relation on the set of alternatives *A*.

**Proof:** We would like to show that if $x \prec_r y$ and $y \prec_r z$, than $x \prec_r z$.

$$D(x, z, g) = \frac{Dist(x, g) - Dist(z, g)}{Dist_{\max} - Dist_{\min}} \quad \Leftrightarrow$$

$$D(x,z,g) = \frac{Dist(x,g) - Dist(y,g) + Dist(y,g) - Dist(z,g)}{Dist_{\max} - Dist_{\min}} \quad \Leftrightarrow$$

$$D(x,z,g) = \frac{Dist(x,g) - Dist(y,g)}{Dist_{\max} - Dist_{\min}} + \frac{Dist(y,g) - Dist(z,g)}{Dist_{\max} - Dist_{\min}} \quad \Leftrightarrow$$

$$D(x, z, g) = D(x, y, g) + D(y, z, g)$$

Given that $x \prec_r y$ and $y \prec_r z$, we have $D(x, y, g) < -\delta$ and $D(y, z, g) < -\delta$. Consequently, $D(x, z, g) < -\delta$ (i.e., $x \prec_r z$). For this reason, $(\prec_r, A)$ is transitive.

**Property 5.4:** The r-dominance defines a strict partial order on the set of alternatives *A*.

**Proof:** Since the r-dominance is an irreflexive, asymmetric and transitive relation on the set of alternatives *A*, then it defines a strict partial order on *A*.

One of the main issues identified by Coello (2000) when discussing preference-based MOEAs is the preservation of the Pareto dominance. Hence, it is interesting to study the compatibility and the completeness of the r-dominance with the Pareto dominance relation. Thus, we first define these two terms. Following the study of Zitzler et al. (2003), the compatibility and the completeness with the Pareto dominance are defined as follows:

**Definition 5.2: Compatibility and completeness with the Pareto dominance**
Let $\rhd$ be an arbitrary binary relation where the expression $x \rhd y$ means that solution $x$ is preferred to solution $y$. The relation $\rhd$ is said to be compatible with the Pareto dominance if and only if:

$$x \rhd y \;\Rightarrow\; x \prec y \tag{5.5}$$

The relation $\rhd$ is said to be complete with the Pareto dominance if and only if:

$$x \prec y \;\Rightarrow\; x \rhd y \tag{5.6}$$

Now, we can announce the following theorem.

**Theorem 5.1:** Given a population of individuals *P*, the r-dominance is (1) *complete* with the Pareto dominance relation and (2) *compatible* with the non Pareto dominance relation.

**Proof:**

1) From the r-dominance definition, if $x$ Pareto-dominates $y$ ( $x \preceq y$ ) than automatically $x$ r-dominates $y$ ( $x \prec_r y$ ). Consequently, the r-dominance is said to be complete with the Pareto dominance.
2) From the r-dominance definition, if $x$ r-dominates $y$ ( $x \prec_r y$ ) then one of the two following assumptions may hold: (a) $x$ Pareto-dominates $y$ ( $x \preceq y$ ) or (b) $x$ and $y$ are Pareto-equivalent. Hence, if $x \prec_r y$ then $y \not\preceq x$. Consequently, the r-dominance is said to be compatible with the non Pareto dominance relation.

From the above mentioned theorem, the r-dominance respects the issue announced by Coello (2000). The r-dominance is said to be *Pareto dominance compliant*, i.e., it does not contradict the order induced by the Pareto dominance. The conclusion to draw from this theorem is that the r-dominance encapsulates DM's preferences expressed as a

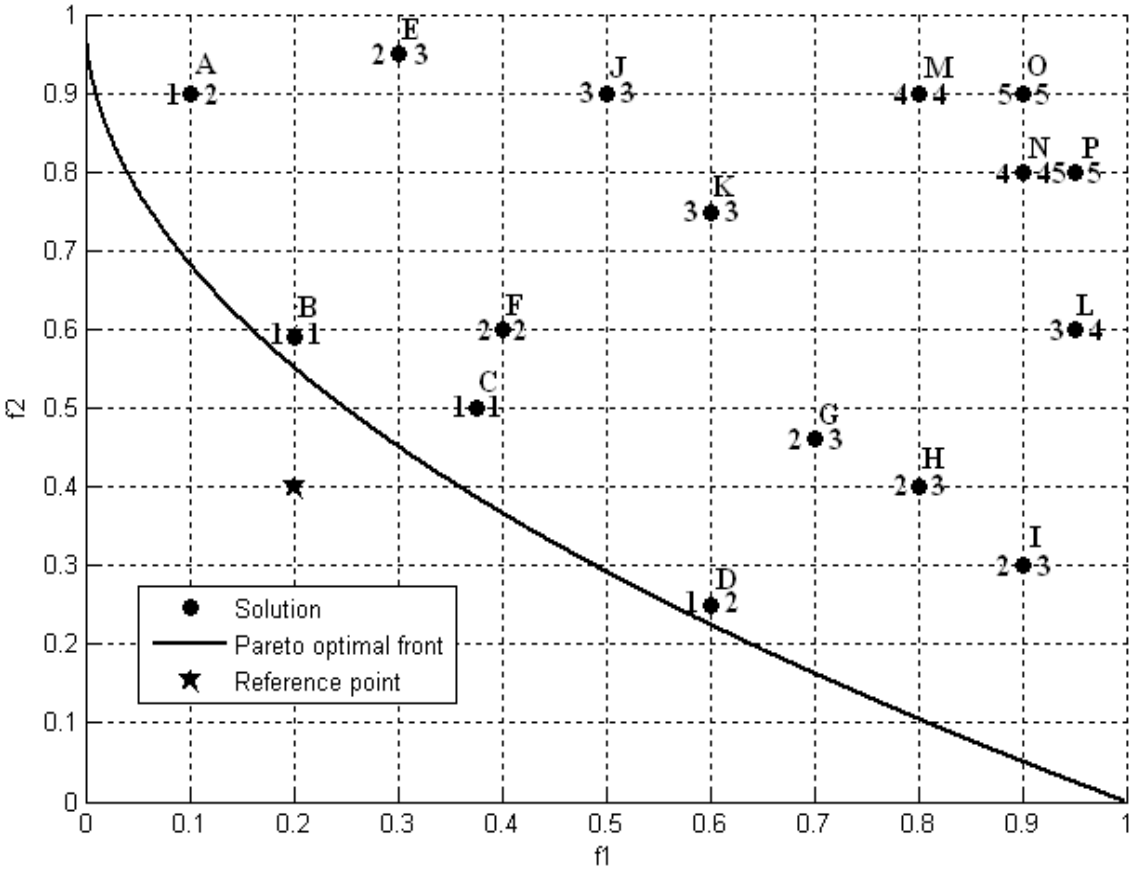



**Figure 5.1 The non-r-dominated sorting ($\delta$= 0.3).**

reference point while *preserving* the Pareto dominance which makes the r-dominance able to focus on Pareto optimal solutions that matches at most the DM's preferences.

### 5.2.2 The non-r-dominated sorting

Inspired from the non-dominated sorting algorithm used in NSGA-II (Deb et al. 2002a), we propose to modify this sorting algorithm by substituting the Pareto dominance relation by the r-dominance relation. We recall that the original non-dominated sorting algorithm classifies a population of individuals into several fronts. First, non-dominated solutions are assigned a rank of one. Then, the individuals belonging to the first front are discarded temporarily and non-dominated solutions from the rest of the truncated population are assigned a rank of two, and so on. This process is performed until the classification of all population individuals. The idea behind the replacement of Pareto dominance by the r-dominance is to classify a population of solutions based on DM's preferences (expressed as a reference point) while preserving the order induced by Pareto dominance. Figure 5.1 illustrates an example of classification of a population of 16 individuals using the r-dominance principle, the reference point (0.2, 0.4) shown with a filled black star and the ZDT1 problem (cf. appendix A). For each population

**Table 5.1 Effect of varying the parameter $\delta$.**

| $\delta$ **value** | 0 | 0.2 | 0.4 | 0.6 | 0.8 | 1 |
|---|---|---|---|---|---|---|
| **Number of classes** | 100 | 6 | 4 | 3 | 2 | 1 |

individual (designed by a filled black circle), we mark the non Pareto domination rank on the left of the individual and the non r-domination rank on the right of the individual. From this figure, it is clear that the Pareto dominance selection pressure is weaker than the r-dominance one. For example, if we take a look around the first non-dominated front composed by solutions (*A, B, C, D*), we remark that this front is subdivided into two different classes when using the non-r-dominated sorting. The two classes are: $C1 = \{B, C\}$ containing individuals that are closer to the reference point and $C2 = \{A, D\}$ containing solutions that are farther from the reference point. We conclude that solutions *A* and *D* are each r-dominated by one of the solutions *B* and *C*. Hence, the selection pressure of the r-dominance relation is stronger than the Pareto dominance one. This selection pressure is guided by DM's preferences and controlled with the non r-dominance threshold $\delta$ (cf. section 5.2.3). Besides, the same phenomenon is observed for the second and third non-dominated fronts. These observations prove what has been demonstrated theoretically in section 5.2.1.

### 5.2.3 Effect of varying the non-r-dominance threshold $\delta$

In this subsection, we study the effect of varying the non-r-dominance threshold value. When observing the definition of the r-dominance, we remark that this relation is equivalent to Pareto dominance in the case where $\delta = 1$. Besides, when $\delta = 0$, the r-dominance induces a total order between *non-equidistant* Pareto-equivalent solutions from the reference point. In other words, when $\delta = 0$, we prefer solutions that are closer to the reference point between such Pareto-equivalent solutions. Table 5.1 illustrates the number of classes obtained when performing a non-r-dominated sorting on a population of 100 Pareto-equivalent (non-equidistant) individuals using different $\delta$ values. We used the ZDT1 problem for this experimentation. The reference point is set to (0.4, 0.6). From table 5.1, we notice that the total number of obtained classes increases when decreasing the non-r-dominance threshold $\delta$ from 1 to 0, and vice versa. The above mentioned remarks concerning $\delta = 1$ and $\delta = 0$ are also observed from this table. Consequently, we conclude that the $\delta$ parameter enables the DM to control the selection pressure of the r-dominance relation.

### 5.2.4 The r-NSGA-II procedure

In this subsection, we propose a variant of the NSGA-II which incorporates the r-dominance concept. We call this variant *reference solution-based NSGA-II* and we

denote it as *r-NSGA-II*. This latter is based on the non-r-dominated sorting and on crowding distance assignment (cf. figure 2.9). The *interactive* scenario of r-NSGA-II is as follows:

- **Step 1:** Ask the DM to provide the population size, the stopping criterion, the reference solution, the weight vector and the $\delta$ value;
- **Step 2:** Run the r-NSGA-II procedure until the stopping criterion is met;
- **Step 3:** Supply the DM with the set of obtained preferred solutions. If the DM is satisfied with the provided set of solutions then stop the optimization process, else ask the DM if he/she would like to update the reference solution, the weight vector, the $\delta$ value and/or the stopping criterion then return to **Step 2**.

It should be noted that **Step 2** is equivalent to running the NSGA-II using the r-dominance instead of the Pareto dominance. It is important to note that r-NSGA-II could be extended to guide the search towards multiple user-supplied reference points and hence it explores multiple ROIs. This fact is achieved by sorting the population based on each user-supplied reference solution in each generation. After that, the rank of the individual is equal to the *minimum* of its assigned ranks. This extension is interesting when the DM is not sure about his/her preferences and he/she would like to explore several ROIs.

### 5.2.5 Managing the non-r-dominance threshold $\delta$

Driving the search towards a certain region of the search space at the beginning of the optimization process may cause a lack of solution diversity since the population individuals will have similar phenotypes. This fact has been treated in (Deb et al. 2006a, Allmendinger et al. 2008; Wickramasinghe and Li 2008) by using some clearing mechanisms. These mechanisms ensure that similar solutions are grouped and a randomly picked solution from each group is retained. All the rest of the group members are discouraged to remain in the race of the evolution process. In order to avoid such additional computations and keep population diversity, we propose to manage the non-r-dominance threshold $\delta$ adaptively during the evolutionary process. It is important to note that the idea of adaptive management of MOEA parameters is not new (Tan et al. 2001). Assuming *nb_gen* to be the number of generations indicated by the DM as a stopping criterion of a certain evolutionary cycle, *gen* to be the current generation index of the current evolutionary cycle, and $\delta$*_user* to be the $\delta$ value provided by the DM, we suggest managing the $\delta$ parameter as shown in figure 5.2. At the initialization stage (*gen* = 0), r-NSGA-II sorts the population based on Pareto dominance (since $\delta = 1$). From the first generation onwards, the $\delta$ value is truncated by a quantity equal to $1-((1-\delta\_user)/nb\_gen)$ in each generation. In this manner, in the

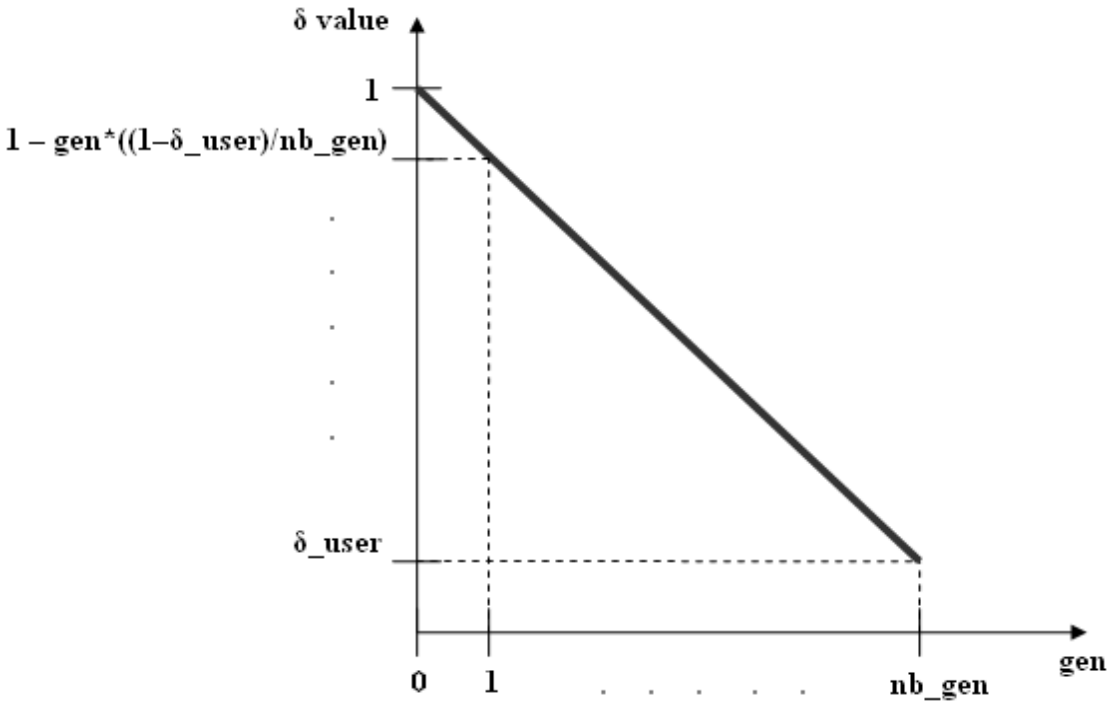



**Figure 5.2 Managing the parameter $\delta$.**

last generation, the $\delta$ value is equal to the $\delta$ value provided by the DM (i.e., $\delta\_user$). The aim of such adaptive management of the $\delta$ parameter is to guide the search gradually during the evolution process towards the ROI which avoids the premature convergence phenomenon to occur.

## 5.3 Experimental study

This section is devoted to demonstrate simulation results on two- to ten-objective test problems using the r-NSGA-II. All experiments are made with MATLAB software[1]. Firstly, we show some interesting a priori simulation results. Secondly, we demonstrate experimentally the positive effect of managing the non-r-dominance threshold $\delta$ adaptively. Then, we describe an interactive scenario. In the last subsection, the r-dominance is compared to three other recently proposed preference-based EMO approaches: (1) the g-dominance (Molina et al. 2009), (2) PBEA (Thiele et al. 2009) and (3) R-NSGA-II (Deb et al. 2006a). For each test problem, we show the mean of the obtained results over 20 independent simulation runs. In all simulations, we use the SBX operator with a distribution index of 10 and polynomial mutation with a distribution index of 20 (Deb and Agrawal 1995). The crossover and mutation probabilities are set to 0.9 and $1/n$ (where $n$ is the number of decision variables)

[1] The used version is MATLAB 7.4 (http://www.mathworks.com).

respectively. For all bi-objective experiments, we use a population of 50 individuals unless otherwise specified.

### 5.3.1 A priori simulation results

First, we consider the bi-objective 30-variable Fonseca test problem (Fonseca 1995). This problem has a concave Pareto optimal front. Figure 5.3 shows the effect of different $\delta$ values on the distribution of the obtained solutions after performing 500 generations (i.e., 25000 Function Evaluations (FEs) since r-NSGA-II evaluates 50 offspring individuals per generation). We use the reference point (0.6, 0.5) designed with filled star and the weight vector (0.5, 0.5). For $\delta$ = 0, all population individuals have converged to the closest Pareto optimal point to the reference solution. In figure 5.3, solutions with other $\delta$ values are shown with an offset to the true Pareto optimal front for ease of visualization. It is obvious that the range of the obtained solutions increases with the increase of the parameter $\delta$. Thus, if the DM would like to obtain a large neighborhood of solutions near the desired region, a large value of $\delta$ should be chosen. We conclude that the DM could control the spread of the obtained ROI by means of the parameter $\delta$. For $\delta$ = 1, the r-NSGA-II provides an approximation of the whole Pareto front. This observation emphasizes our claim discussed in section 5.2.3 (i.e., the r-dominance is equivalent to the Pareto dominance when $\delta$ = 1).

Next, we investigate the effect of changing the weight vector of the weighted Euclidean distance on the distribution of the obtained preferred solutions. We use the same problem as previously. Figure 5.4 shows the obtained solutions after 500 generations (i.e., 25000 FEs) with the three following weight vectors: (0.5, 0.5), (0.25, 0.75) and (0.75, 0.25). The reference point and the $\delta$ parameter are settled to (0.6, 0.5) and 0.4 respectively. The solutions obtained for each one of the weight vectors are superimposed on another in figure 5.4 for ease of visualization. From this figure, we remark that modifying the weight vector introduces a bias on the distribution of the ROI solutions. For the first weight vector (0.5, 0.5), there is no bias among the obtained solutions. As expected for the second weight vector (0.25, 0.75), there is more emphasis on $f_2$, thereby obtaining solutions that optimize $f_2$ more than $f_1$. For the third weight vector (0.75, 0.25), the opposite phenomenon is observed. Thus, the ASF weights express a *second level of preferences*, i.e., if the DM would like to bias some objectives more than others, a biased distribution of near user-supplied reference point solutions could be obtained by r-NSGA-II. In all subsequent simulations, we use a uniform weight vector but the user can modify this vector freely if he/she is interested to bias some objectives over others.

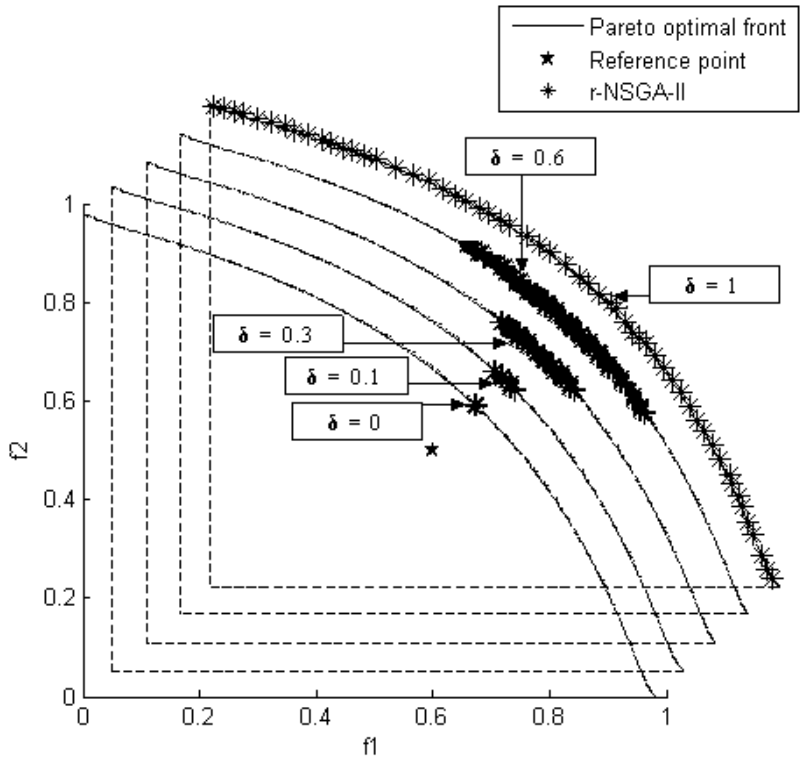



**Figure 5.3 Effect of varying the $\delta$ value on Fonseca test problem.**

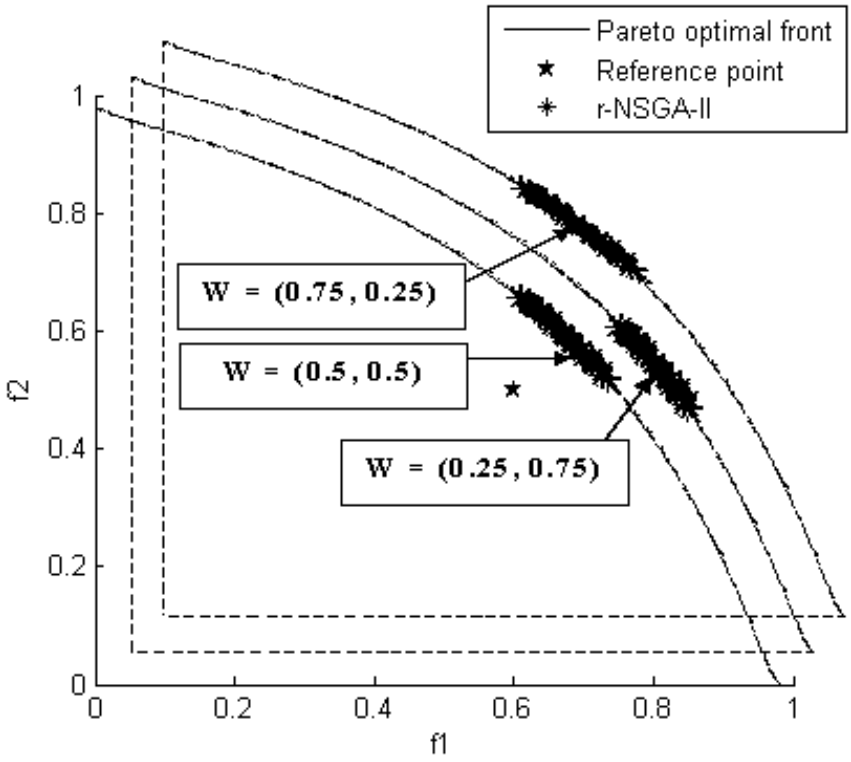



**Figure 5.4 Effect of varying weights on Fonseca test problem.**

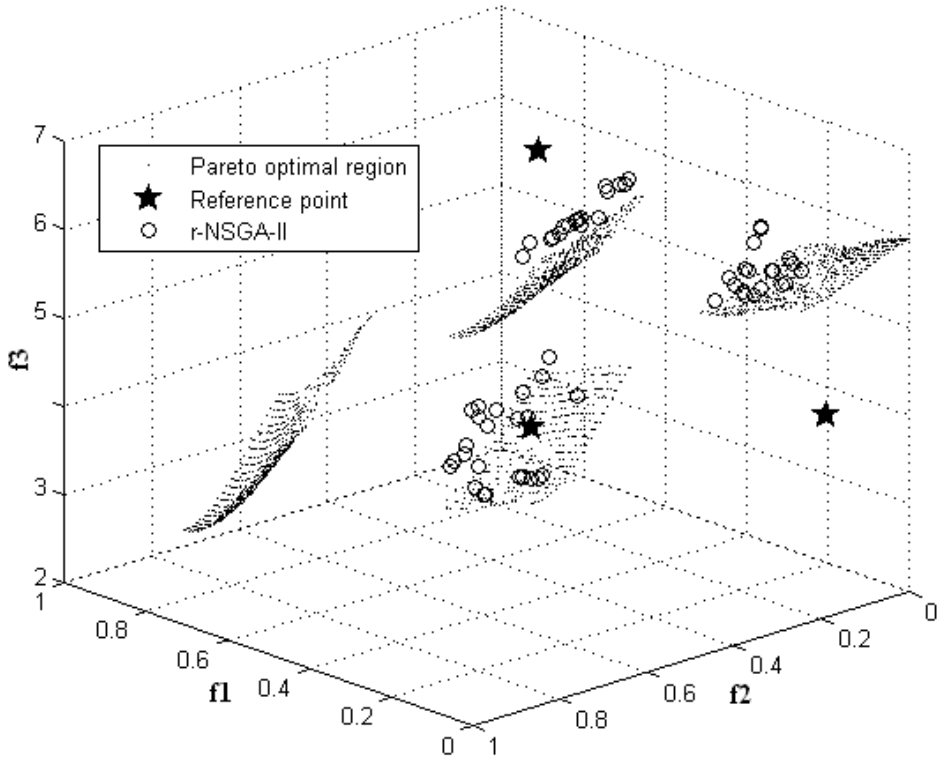



**Figure 5.5 Handling three reference points on the tri-objective DTLZ7 problem.**

Now, we assess the ability of our algorithm to search for multiple ROIs near multiple reference points on the tri-objective DTLZ7 problem. We recall that this problem has four disjoint Pareto optimal regions (cf. section 2.4.1). We use three reference points: (1) a feasible Pareto optimal point $A$(0.1650, 0.7100, 5.6780), (2) a feasible non Pareto optimal point $B$(0.75, 0.15, 6.00) and (3) an infeasible point $C$(0.10, 0.10, 5.00). The population size is set to 60 and the parameter $\delta$ is set to 0.2. Figure 5.5 shows the preferred obtained solutions after 500 generations (i.e., 30000 FEs). Through this figure, we notice that our proposed approach could be easily extended to handle multiple reference points and thus it discovers various ROIs. This is an important feature when the DM is not sure about his/her preferences and he/she would like to explore several ROIs. In addition, it should be noted that the r-NSGA-II is *insensitive* to the feasibility of the reference point.

In the following, we solve 19-variable 10-objective DTLZ2 problem with the reference point: (0.30, 0.30, 0.30, 0.10, 0.30, 0.55, 0.35, 0.35, 0.25, 0.45). The $\delta$ parameter is set to 0.2. We use a population of 200 individuals. Figure 5.6 shows the parallel coordinates plot of the obtained preferred solutions after performing 500 generations (i.e., 100000 FEs). The higher number of objectives makes the problem harder.

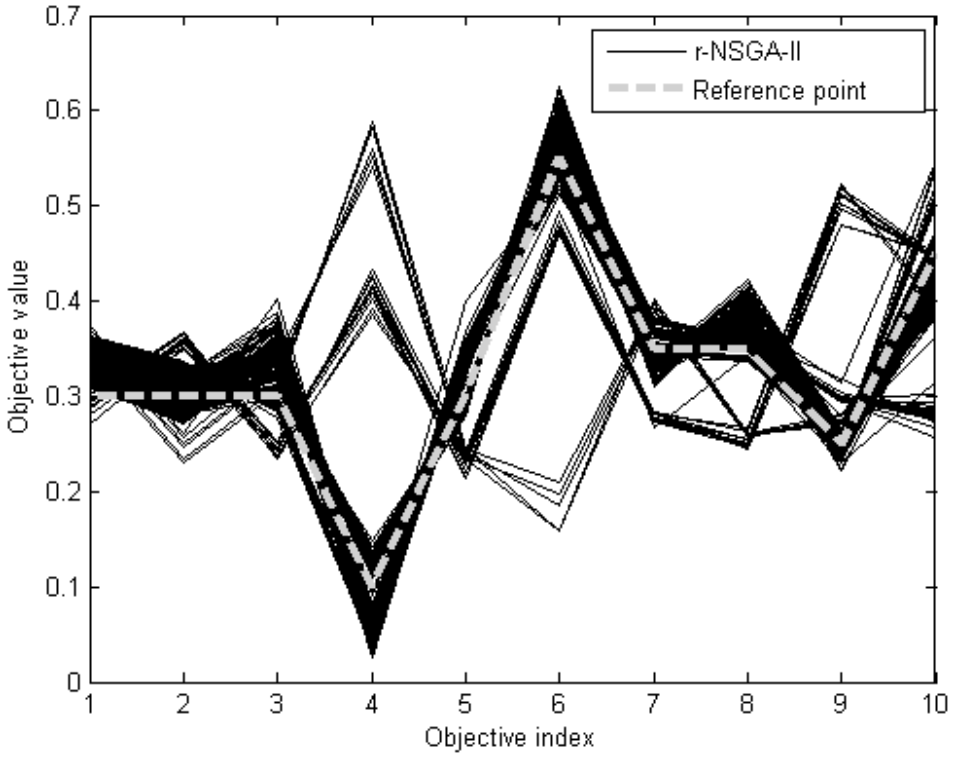



**Figure 5.6 The ROI for the 10-objective DTLZ2.**

Therefore, the population size and the number of generations are increased. From figure 5.6, we remark that, although the objective values lie in [0,1], most obtained solutions are concentrated near the reference point designed with a dashed bold grey line which would be the region closest to the chosen reference point. When computing $\sum_{i=1}^{10} f_i^{\,2}$ for all obtained solutions, the values are found to lie within [1.131, 1.324], thereby meaning that all solutions are near the true Pareto region (since Pareto optimal solutions of the 10-objective DTLZ2 satisfy $\sum_{i=1}^{10} f_i^{\,2} = 1$).

### 5.3.2 Adaptive management effect

In order to show the positive effect of managing the $\delta$ parameter value adaptively, we compared two versions of our algorithm: (1) a first version of r-NSGA-II without adaptive management, denoted r-NSGA-II$_W$, where the value of $\delta$ is constant during the overall run and (2) a second version of r-NSGA-II with adaptive management, denoted r-NSGA-II$_A$ (i.e., the original r-NSGA-II version described in section 5.2), where the value of $\delta$ is updated during the run according to the strategy described by figure 5.2. For this experiment, we use the 30-variable bi-objective ZDT3 test problem. Its Pareto front consists of several non-contiguous convex parts. The reference point and the weight vector are set to (0.4, 0.0) and (0.5, 0.5) respectively. The $\delta$ value is settled to

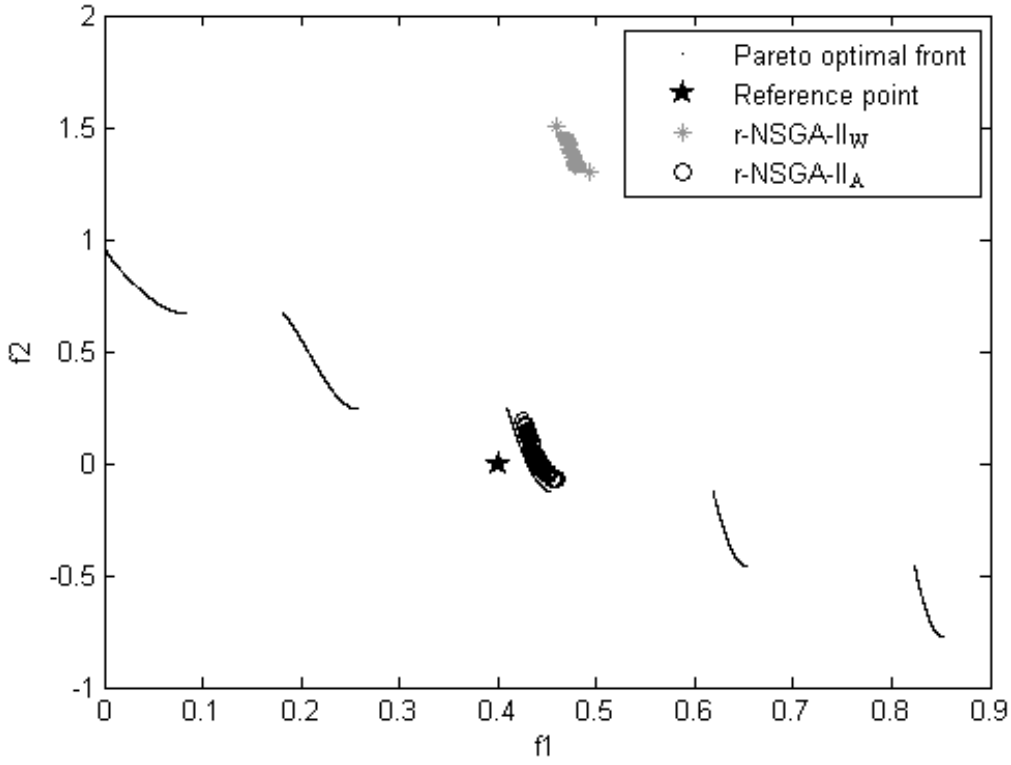



**Figure 5.7 Showing the positive effect of the adaptive management of the parameter $\delta$ on ZDT3.**

0.2. Figure 5.7 demonstrates the obtained results after performing 500 generations (i.e., 25000 FEs) for the two versions. From this figure, we remark that r-NSGA-II$_W$ is unable to reach the Pareto optimal front. However, r-NSGA-II$_A$ provides solutions lying on the optimal Pareto front. As discussed previously in section 5.2.5, focusing the search towards a certain region of the search space at the beginning of the evolutionary process may reduce the diversity and hence preventing the population individuals to progress towards the Pareto optimal front. This phenomenon explains the obtained results by r-NSGA-II$_W$. The conclusion to draw from this experimentation is that the adaptive management of the $\delta$ parameter is a bonus feature for r-NSGA-II since it permits reaching the Pareto optimal front while focusing on the desired ROI.

### 5.3.3 Interactive scenario

In this subsection, we describe an interactive run of the r-NSGA-II algorithm on the ZDT1 problem. The overall interactive scenario is illustrated by figure 5.8 and it is composed of four cycles. We call *cycle* a run of the r-NSGA-II algorithm for a certain number of generations freely fixed by the DM. Firstly, the DM performs a run of 50 generations (first cycle) without preferences (i.e., $\delta = 1$) in order to have an idea about

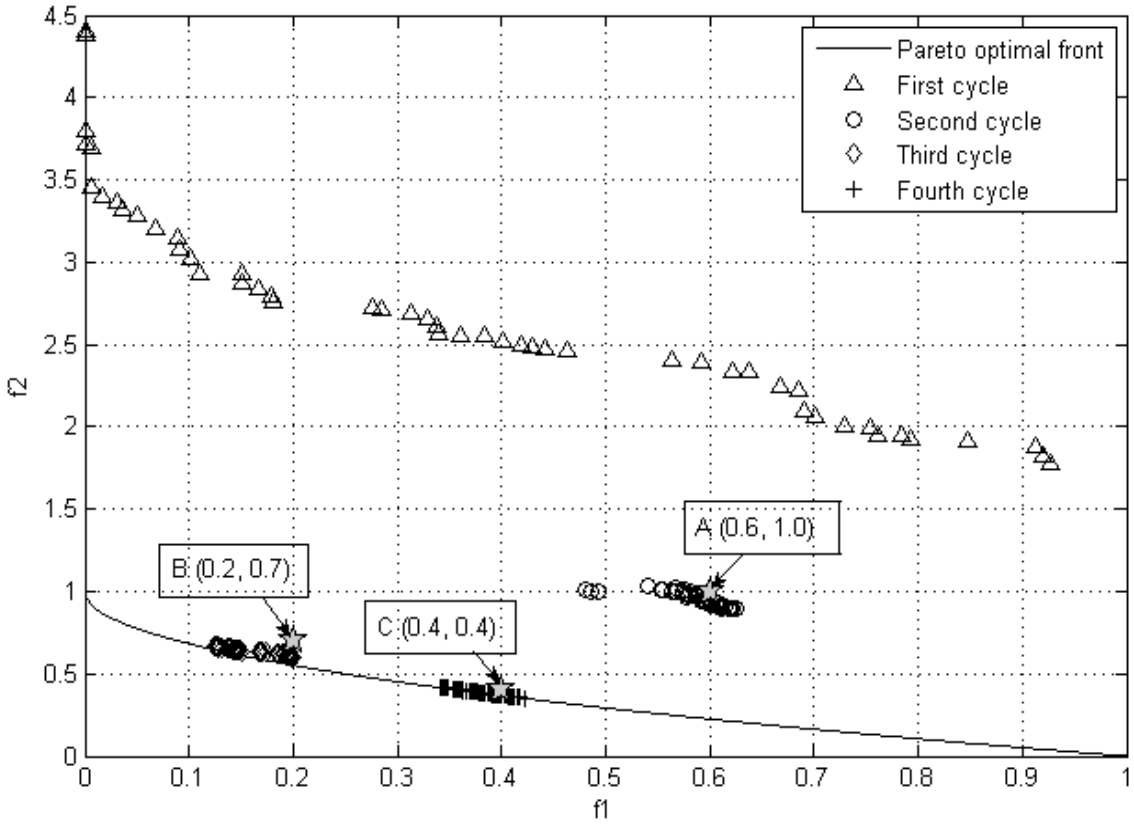



**Figure 5.8 Interactive scenario on ZDT1.**

the objective ranges. The solutions designed with triangles are obtained. These solutions are well diversified as shown in figure 5.8. After that, the DM puts the reference point to $A(0.6, 1.0)$ and performs a run of 100 generations (second cycle) with $\delta = 0.25$. Near user-supplied reference point solutions are obtained (designed with circles). We suppose now that the DM is dissatisfied. Thus, he/she resets the reference point to $B(0.2, 0.7)$ and he/she reruns the search process for 150 generations (third cycle) with $\delta = 0.2$ and a bias expressed by using a non uniform weighting coefficient vector (0.75, 0.25). Pareto optimal solutions drawn with diamonds are obtained near the reference solution. These solutions show a bias emphasizing the minimization of $f_1$ more than $f_2$. The DM is always dissatisfied. He/She performs a run of 100 generations (fourth cycle) with the reference point $C(0.4, 0.4)$, a uniform weight vector and the same $\delta$ value. The solutions shown with the symbol "+" are obtained. They belong to the Pareto optimal front while matching DM's preferences. It is important to notice that, after performing the third cycle, the r-NSGA-II gets Pareto optimal solutions in the vicinity of the reference point $B(0.2, 0.7)$. Nevertheless, the DM is not interested with such solutions. Hence, he/she resets the reference point to $C(0.4, 0.4)$ and performs a fourth cycle. The Pareto optimal solutions seem to slide along the Pareto optimal front towards the chosen reference point. We conclude that: (1) the r-NSGA-II solutions can

move along the Pareto front based on DM's preferences and (2) the reference point plays the role of a "*magnet*" to the candidate solutions. It is worth noting that several parameters were varied from one cycle to another just to demonstrate that this is possible but it is not necessary. Hence, the DM can modify several parameters after each performed evolutionary cycle which makes him/her free when specifying his/her preferences interactively. Consequently, the population individuals seem to "*walk*" in the objective space according to these preferences. This fact develops the DM's acquired knowledge about the exploration of the search space.

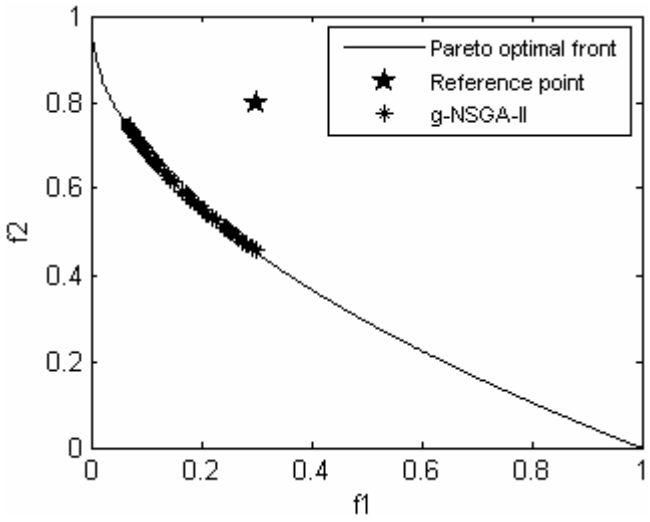
### 5.3.4 Comparative experiments

In this subsection, we conduct three sets of experiments. In the first set, we confront our approach to the g-dominance of Molina et al. (2009). In the second set, we compare the r-NSGA-II to PBEA of Thiele et al. (2009). Finally, in the third set, we make a comparison between r-NSGA-II and R-NSGA-II of Deb et al. (2006a). It should be noted that (Molina et al. 2009) and (Thiele et al. 2009) are the most recent reference point-based EMO methodologies cited in table 3.1 (cf. section 3.3).

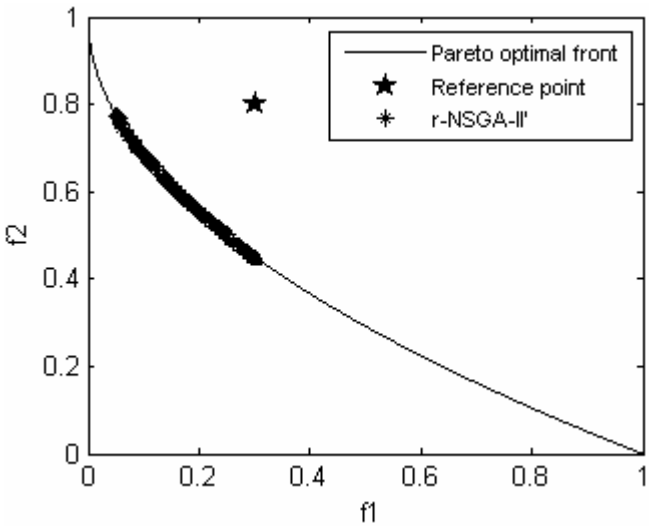
- **r-dominance versus g-dominance**

In this subsection, we compare the r-dominance relation to the recently proposed g-dominance relation on two- to ten-objective optimization problems. Differently speaking, we compare the r-NSGA-II to the g-NSGA-II (i.e., the NSGA-II version incorporating the g-dominance (Molina et al. 2009)). For fairness of comparison, we use two instances of r-NSGA-II with two $\delta$ values: (1) a first instance with $\delta$ = 0.3 (denoted r-NSGA-II') and (2) a second instance with $\delta$ = 0.1 (denoted r-NSGA-II"). This comparison is made by: (1) visualizing the plots of the obtained solutions and (2) using the additive binary ε-indicator as a performance indicator (cf. section 2.5.2).

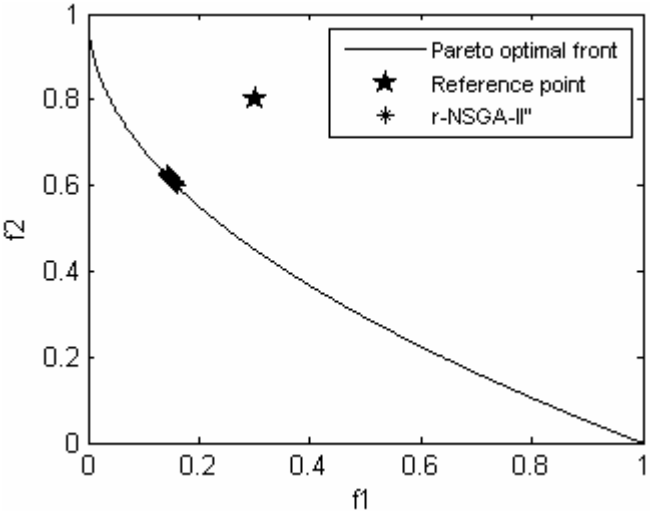
Firstly, we consider the 30-variable ZDT1 problem. We perform two runs of 500 generations each one. For the first run, we use a feasible reference point which is far from the Pareto front (0.3, 0.8). For the second run, we use a feasible reference point which is very near to the Pareto front (0.5, 0.3). Figure 5.9 shows the obtained preferred solutions of the first run (i.e., for the reference point (0.3, 0.8)). We observe that g-NSGA-II and r-NSGA-II' provide similar results. The two ROIs are very similar viewpoint convergence and spread. However, the spread of the ROI of r-NSGA-II" is so small since $\delta$ = 0.1 in this instance. Table 5.2 presents the additive binary epsilon indicator values for ZDT1 problem (we suppose that $G$ is the g-NSGA-II solution set, $R'$ is the r-NSGA-II' solution set and $R''$ is the r-NSGA-II" one). From this table, we remark that for the reference point (0.3, 0.8):




**(a)**




**(b)**




**(c)**

**Figure 5.9 Preferred solutions on ZDT1 with the reference point (0.3, 0.8): (a) g-NSGA-II, (b) r-NSGA-II' and (c) r-NSGA-II".**

**Table 5.2 Binary ε-indicator values for ZDT1.**

| **Reference point** | (0.3, 0.8) | (0.5, 0.3) |
|---|---|---|
| **(I(R', G), I(G, R'))** | (0.0495, 0.0800) | (-0.0176, 0.1093) |
| **(I(R", G), I(G, R"))** | (0.1433, 0.0090) | (-0.0117, 0.0389) |

- I(*R*', *G*)*>0*, I(*G*, *R*')>0 and I(*R*', *G*)<I(*G*, *R*'). The set *R*' is said to be better than *G* in a weaker sense,
- I(*G*, *R*")>0 and I(*R*", *G*)>0 and I(*G*, *R*")< I(*R*", *G*). The set *G* is said to be better than *R*'' in a weaker sense.

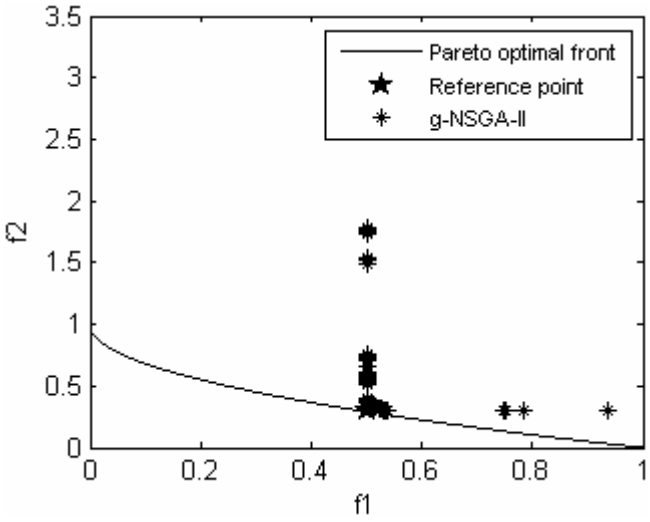
This result may be explained by the fact that the spread of the ROI of r-NSGA-II" is smaller than the spread of g-NSGA-II ROI since the two ROIs have approximately the same convergence rate. Hence, the ROI having greater spread is better viewpoint the ε-indicator value.

Figure 5.10(a-c) illustrates the obtained preferred solutions of the second run (i.e., for the reference point (0.5, 0.3)). We observe from the plots that the two r-NSGA-II instances perform better than g-NSGA-II viewpoint convergence and diversity. The ε-indicator values emphasize these observations since:

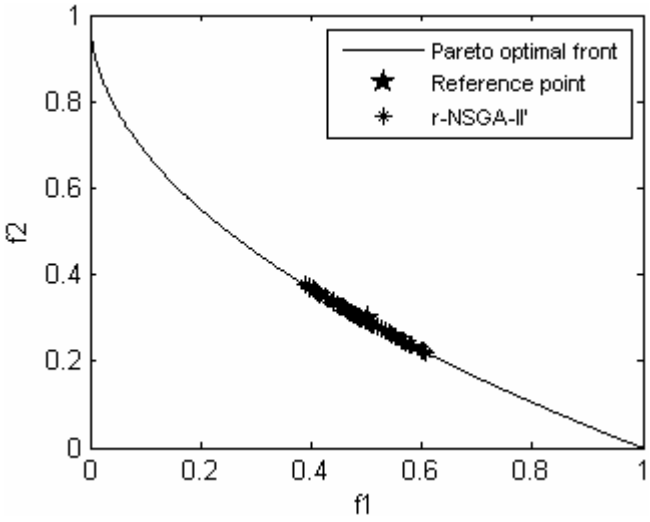
- I(*R*', *G*) <0 and I(*G*, *R*')>0, and
- I(*R*", *G*) <0 and I(*G*, *R*")>0.

The conclusion to draw from all these observations is that the g-dominance performs worse when the reference point is *very close* to the Pareto optimal front. This phenomenon could be explained by the fact that the g-dominance excludes solutions fulfilling *partially* the goals. Hence, the solutions being Pareto-equivalent to the reference point (which are important in the case where the reference point is near the Pareto front) are discouraged to remain in the race. However, the r-dominance is insensitive to such a problem since it preserves this kind of solutions. This characteristic seems to be the main inconvenience of the g-dominance.

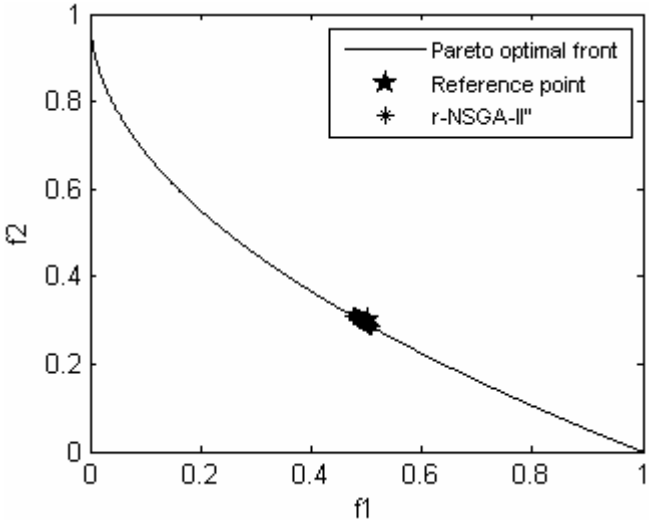
It should be noted here that the DM can control the spread of the obtained ROI when using r-NSGA-II. However, this is not possible when using g-NSGA-II. This fact represents another advantage of the r-dominance over the g-dominance. Additionally, the g-dominance is very dependent on the position of the reference point in the search space. For example, if we use one of the reference points (0, 0) or (1, 1) when solving ZDT1 problem, g-NSGA-II will provide an approximation of the whole Pareto front instead of a bounded ROI. More generally, when using a reference point (*x*, *y*) such that




**(a)**




**(b)**




**(c)**

**Figure 5.10 Preferred solutions on ZDT1 with the reference point (0.5, 0.3): (a) g-NSGA-II, (b) r-NSGA-II' and (c) r-NSGA-II".**

**Table 5.3 Binary ε-indicator values for tri-objective DTLZ2.**

| **Reference point** | (0.4, 0.8, 0.8) | (0.3, 0.5, 0.8) |
|---|---|---|
| **(I(R', G), I(G, R'))** | (-0.0975, 0.1570) | (-0.0218, 0.0696) |
| **(I(R", G), I(G, R"))** | (-0.0492, 0.1314) | (-0.0319, 0.0578) |

"$x \leq 0$ and $y \leq 0$" or "$x \geq 1$ and $y \geq 1$", g-NSGA-II approximates the whole Pareto front of ZDT1 problem instead of a bounded ROI. This shortcoming is independent of the used multi-criteria problem and can be generalized to any MOP. This fact is due to the g-dominance principle which prefers solutions satisfying all aspiration levels and solutions fulfilling none of the aspiration levels over solutions satisfying some of the aspiration levels. This last mentioned defect makes the g-dominance inefficient when used interactively since it can mislead the DM. Moreover, we note that the r-dominance is insensitive to the reference point position which makes it superior to the g-dominance when facing such situations.

In the following, we make a comparison between the two r-NSGA-II instances and g-NSGA-II on the 12-variable tri-objective DTLZ2 test problem. We recall that this problem has a non-convex three-dimensional Pareto optimal region. We perform two simulation runs: (1) a first run with a feasible reference point which is far from the Pareto front (0.4, 0.8, 0.8) and (2) a second run with a feasible reference point which is very near to the Pareto front (0.3, 0.5, 0.8). The population size and the number of generations are set to 50 and 500 respectively. Unlike the bi-objective case, we observe from figures 5.11(a-c) and 5.12(a-c) that the two instances of r-NSGA-II perform better than g-NSGA-II in the tri-objective case for the two used reference points. This claim is emphasized by the binary ε-indicator values presented in table 5.3 (assuming the same notation as table 5.2). It should be noted that the same remark, discussed for the bi-objective case with ZDT1 test problem, remains valid since the reference point (0.3, 0.5, 0.8) is very close to the Pareto optimal region. This disadvantage of the g-dominance relation is dangerous when the DM runs the algorithm interactively especially when he/she puts a reference point near the Pareto front. It should be noted that the probability of the event "*setting a reference point near the Pareto front*" increases during the interactive run of the algorithm since the search is guided towards the optimal region progressively. This phenomenon makes the g-dominance inefficient when used in an interactive way.

Next, we solve the 10-objective DTLZ2 problem using g-NSGA-II and compare the obtained approximation to r-NSGA-II' and r-NSGA-II" ones. The population size and the number of generations are set to 200 and 500 respectively. The used reference point

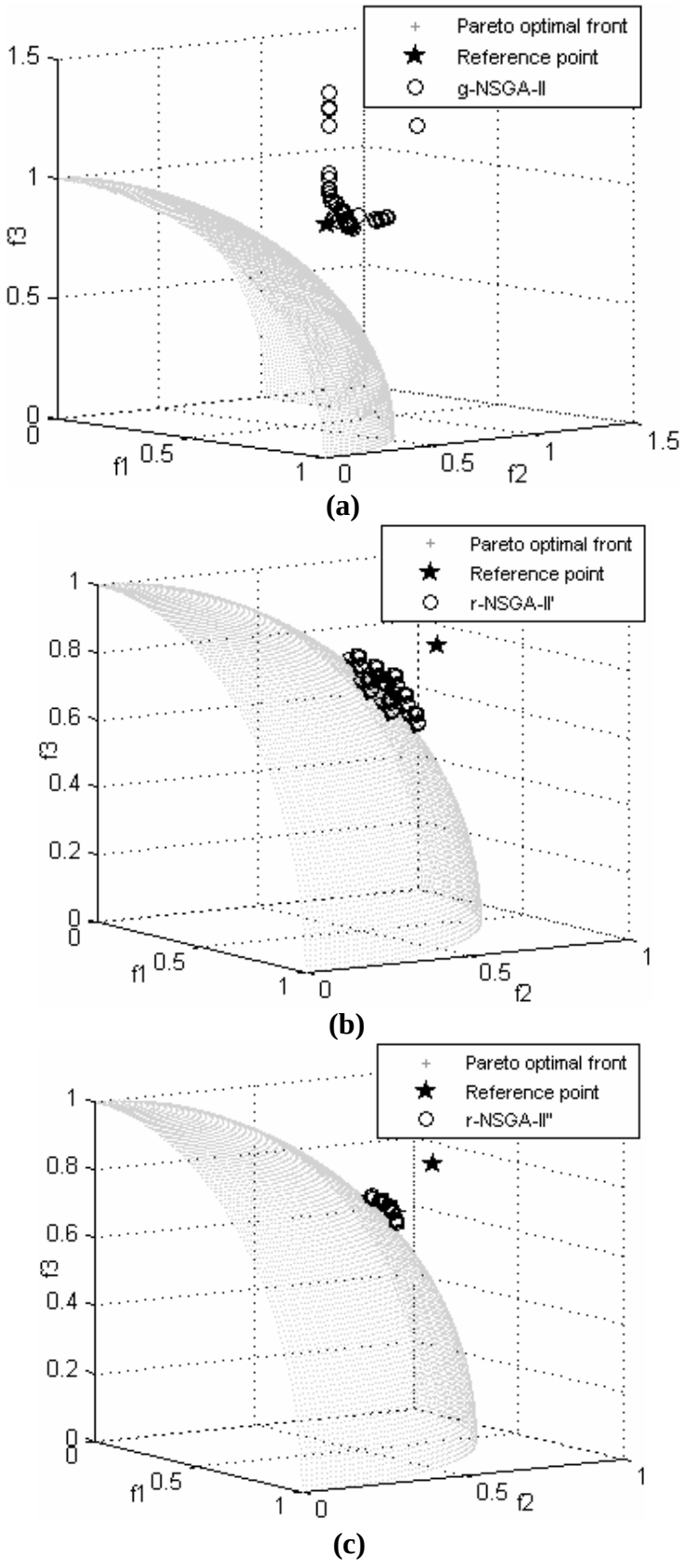



**Figure 5.11 Preferred solutions on tri-objective DTLZ2 with the reference point (0.4, 0.8, 0.8): (a) g-NSGA-II, (b) r-NSGA-II' and (c) r-NSGA-II".**

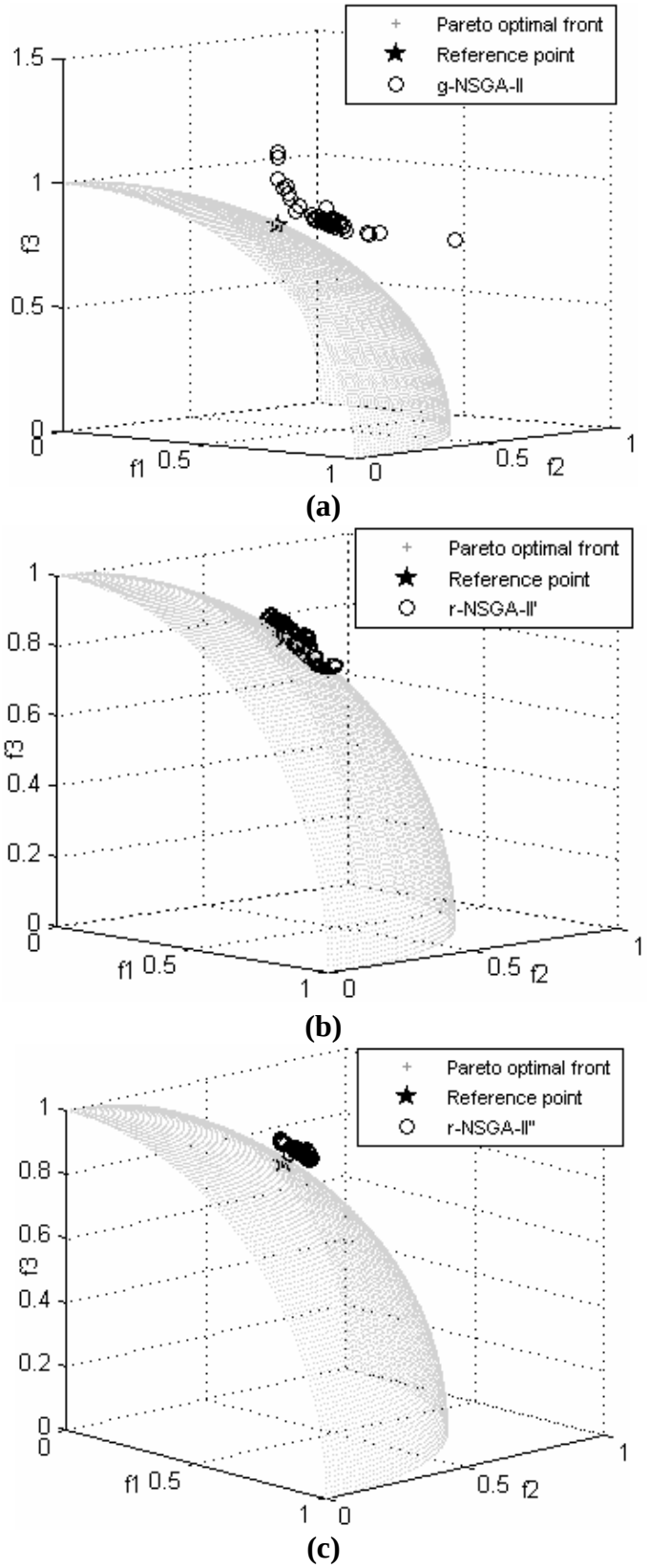



**Figure 5.12 Preferred solutions on tri-objective DTLZ2 with the reference point (0.3, 0.5, 0.8): (a) g-NSGA-II, (b) r-NSGA-II' and (c) r-NSGA-II".**

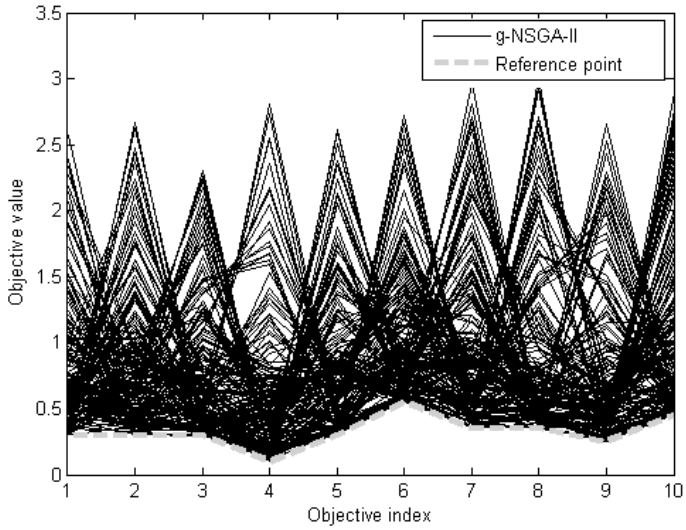
**Table 5.4 Binary ε-indicator values for 10-objective DTLZ2.**

| **Reference point** | (0.30, 0.30, 0.30, 0.10, 0.30, 0.55, 0.35, 0.35, 0.25, 0.45) |
|---|---|
| **(I(R', G), I(G, R'))** | (0.0242, 1.043) |
| **(I(R", G), I(G, R"))** | (0.0544, 1.0428) |

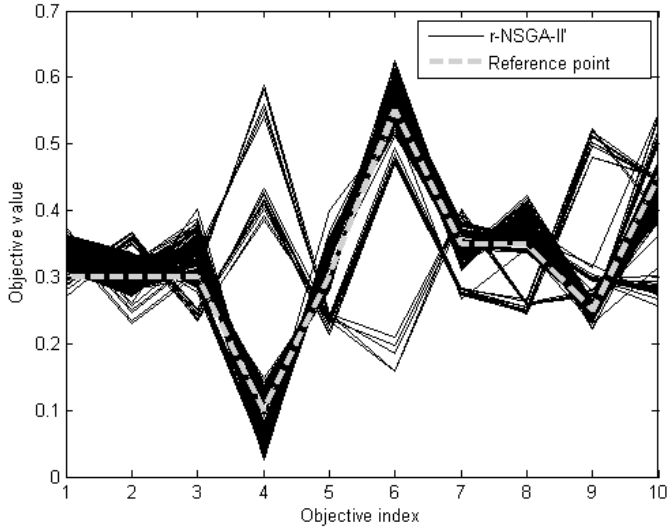
is (0.30, 0.30, 0.30, 0.10, 0.30, 0.55, 0.35, 0.35, 0.25, 0.45). From figure 5.13(a-c), we remark that the performance of g-NSGA-II degrades dramatically with the increase of the number of objectives to optimize. In order to validate our claim, we compute the quantity $\sum_{i=1}^{10} f_i^2$ for all obtained solutions of each algorithm. The values are found to lie within [6.6884, 11.5308] for g-NSGA-II, within [1.131, 1.324] for r-NSGA-II' and within [1.127, 1.884] for r-NSGA-II". Unlike the two instances of r-NSGA-II, g-NSGA-II provides solutions that are so far from the true Pareto region (since Pareto optimal solutions of DTLZ2 satisfy $\sum_{i=1}^{10} f_i^2 = 1$). This phenomenon could be explained by the fact that the probability of obtaining solutions fulfilling all the goals (eventually none of the goals) decreases largely when the problem dimension increases. Table 5.4 demonstrates the ε-indicator values for this problem (assuming the same notation as table 5.3). The obtained values show the performance of r-NSGA-II' and r-NSGA-II" over g-NSGA-II. We conclude that the r-dominance could be not only a way to tackle *many-objective* optimization problems (i.e., MOPs with $M > 3$ (Hughes 2005; López Jaimes et al. 2009)) but also a tool to search for ROIs in such type of problems. In fact, as discussed by Farina and Amato (2004), the Pareto dominance could be unsatisfactory in the many-criteria decision making task due to two reasons: (1) the number of improved or equal objective values is not taken into account and (2) the (normalized) size of improvements is not considered. The r-dominance could be a way to overcome the two aforementioned defects since the used ASF encapsulates *implicitly* the number of improvements and the size of these improvements. These observations emphasize the obtained results on the 10-objective DTLZ2 problem.

- **r-NSGA-II versus PBEA**

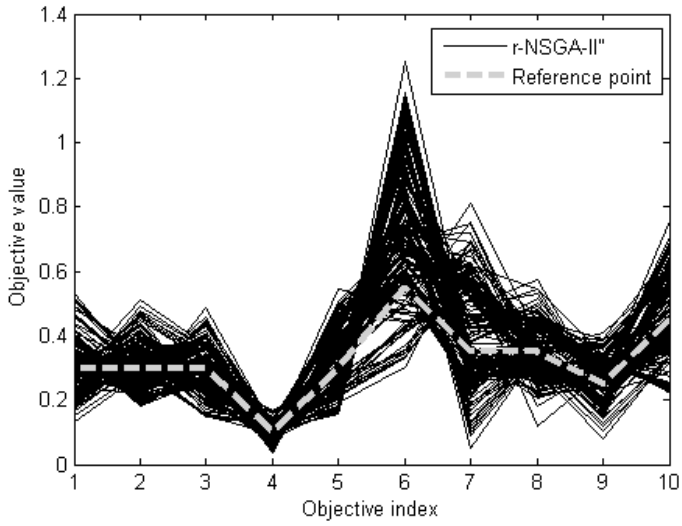
PBEA and r-NSGA-II are similar in the fact that they require reference point(s) and a parameter controlling the spread of the obtained ROI(s) supplied by the DM. However, their internal behaviors are different since r-NSGA-II is based on non-r-dominated sorting and crowding distance assignment and not on indicator based selection. In this subsection, we make a comparison between r-NSGA-II and PBEA on two test problems. We use the same experimental design presented in (Thiele et al. 2009) and described in table 5.5. We note that the comparative experiments with PBEA are done




**(a)**




**(b)**




**(c)**

**Figure 5.13 Preferred solutions on 10-objective DTLZ2 with the reference point (0.30, 0.30, 0.30, 0.10, 0.30, 0.55, 0.35, 0.35, 0.25, 0.45): (a) g-NSGA-II, (b) r-NSGA-II' and (c) r-NSGA-II".**

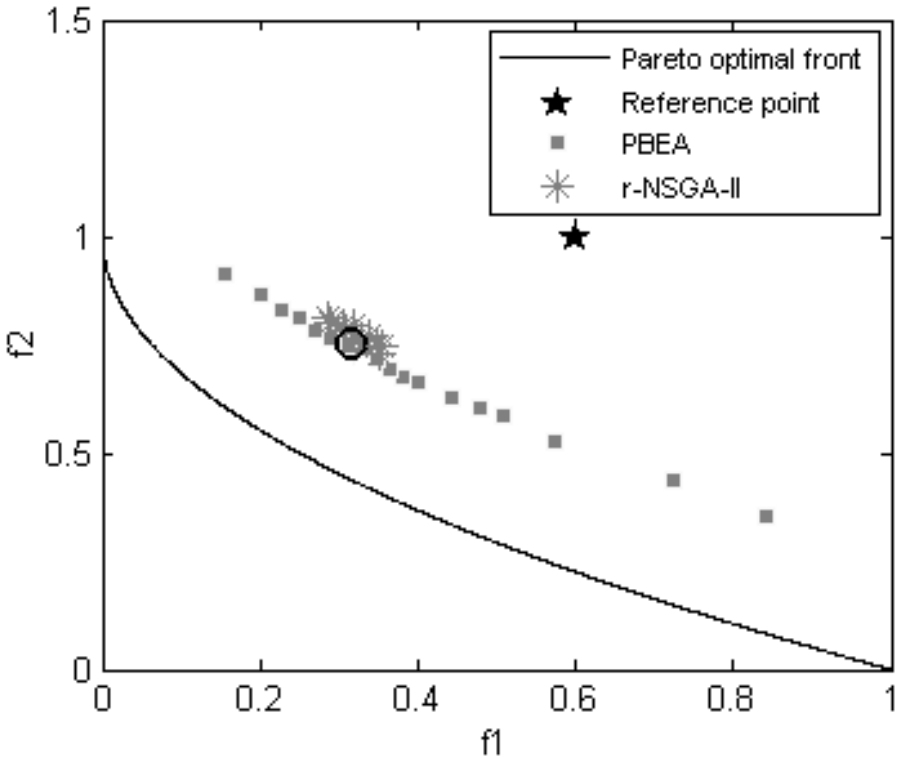



**Figure 5.14 r-NSGA-II versus PBEA on ZDT1 with the reference point (0.6, 1.0).**

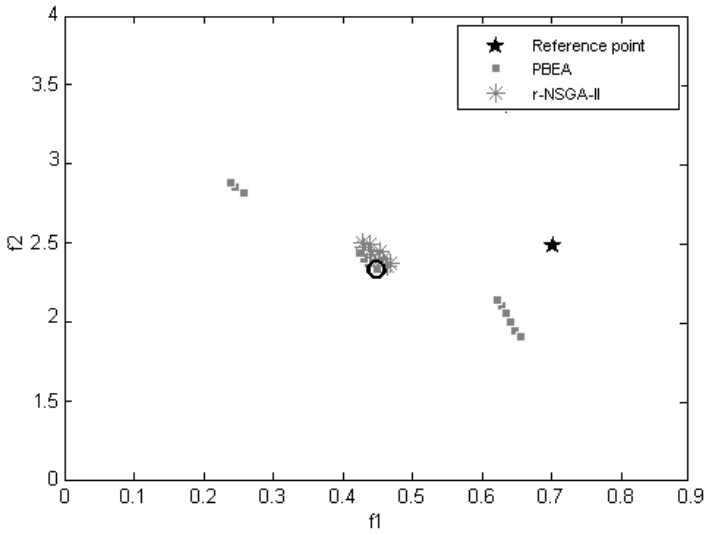
**Table 5.5 Parameter settings: r-NSGA-II versus PBEA.**

| **Parameters** | **r-NSGA-II** | **PBEA** |
|---|---|---|
| Population size | 20 | 20 |
| Number of generations | 500 for ZDT1<br>100 for ZDT3 | 500 for ZDT1<br>100 for ZDT3 |
| Fitness scaling factor (κ) | - | 0.05 |

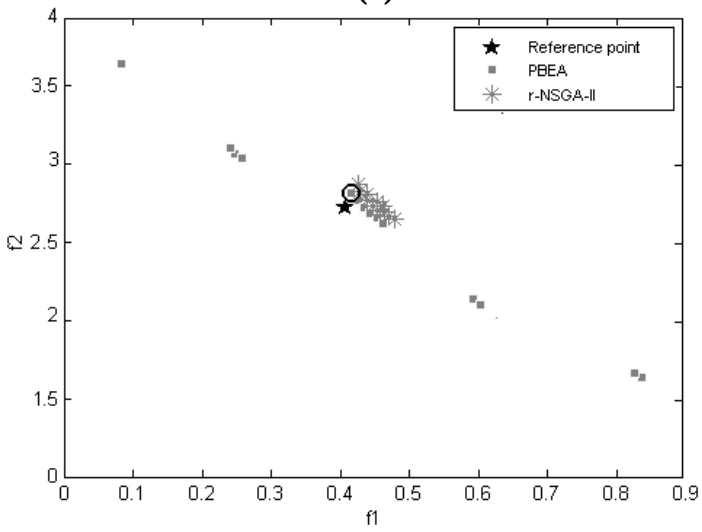
under the framework of PISA (Bleuler et al. 2003). The first experiment is made with the bi-objective ZDT1 problem using the reference point (0.6, 1.0). For both algorithms, we use the same spread, i.e., $\delta = \delta_{PBEA} = 0.1$. The obtained ROI approximations of the two algorithms are shown by figure 5.15(a-c). The two approximations have similar convergence rates. However, although we use the same spread value (termed specificity in (Thiele et al. 2009)) for the two algorithms, we observe that r-NSGA-II approximation is more concentrated around *the projected*

*reference point* shown by a black circle. The projected reference point corresponds to the solution having the smallest ASF value among PBEA approximation members. We conclude that the mechanisms designed in each one of the algorithms to control the spread of the ROI are different and have different effects. It is worth noting that the spread control is easier when using r-NSGA-II than with the use of PBEA. In fact, the upper/lower bounds of the spread parameter $\delta$ of r- NSGA-II are known since $\delta \in [0,1]$ (cf. definition 5.1). Contrariwise, the upper/lower bounds of $\delta_{PBEA}$ are unknown which represents a great difficulty to the DM to control the ROI spread especially when solving real world MOPs with PBEA.

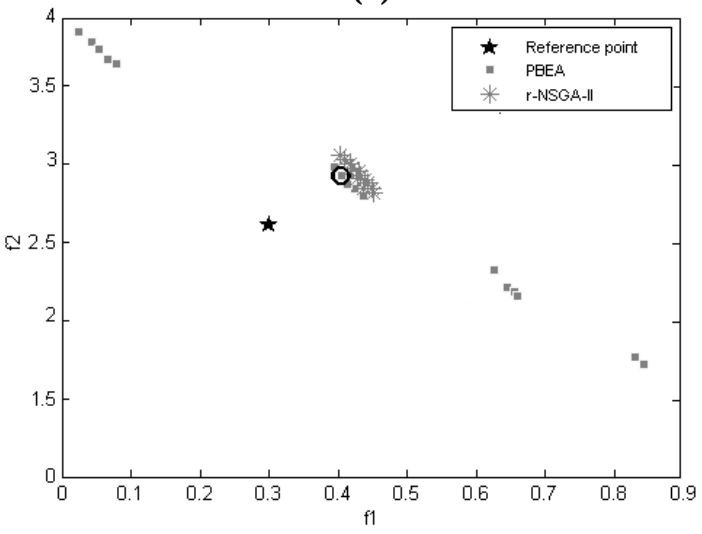
The second experiment is performed with the bi-objective ZDT3 test problem. We use three reference points: (0.7, 2.5), (0.4, 2.7) and (0.3, 2.6). Figure 5.15 shows the approximations obtained by PBEA ($\delta_{PBEA}$ = 0.03 for the first reference point, $\delta_{PBEA}$ = 0.02 for the second one and $\delta_{PBEA}$ = 0.01 for the third one (Thiele et al. 2009)) and those obtained by r-NSGA-II ($\delta$ = 0.2 for the three reference points). We observe, from this figure, that for each reference point r-NSGA-II produces a bounded ROI in the vicinity of the projected reference point designed by a black circle. Nevertheless, for each reference point, PBEA provides a biased distribution of solutions concentrated in the proximity of the projected reference point. We remark that the larger the distance between the reference point and PBEA approximation is, the smaller the effect of concentring the search around the projection of the reference point is. This phenomenon can be easily seen from figure 5.15(c) where the *optimistic* reference point (0.3, 2.6) is used. Additionally, these observations were highlighted in the work of Thiele et al. (2009). This fact represents the main inconvenience of PBEA since the probability of the event "*setting an optimistic reference point"* increases during the interactive run of PBEA, since the population is converging progressively towards the Pareto optimal front, which can mislead the DM when specifying his/her preferences. However, the r-NSGA-II is insensitive to such problem. In summary, we can resume the advantages of r-NSGA-II over PBEA as follows: (1) the control of the spread of the ROI in r-NSGA-II is easy in contrast to PBEA where this control is very rough, (2) contrarily to PBEA, r-NSGA-II is independent of the position of the reference point in the search space and (3) finally, the last point which has not been mentioned previously is the sensitivity of IBEA, and eventually PBEA, to the fitness scaling factor κ which depends on the used quality indicator and on the MOP under consideration (Zitzler and Künzli 2004). Thus, setting an appropriate value to κ represents a great difficulty to the DM when solving real world multi-criteria problems.




**(a)**




**(b)**




**(c)**

**Figure 5.15 r-NSGA-II versus PBEA on ZDT3 with the reference points: (a) (0.7, 2.5), (b) (0.4, 2.7) and (c) (0.3, 2.6).**

♦ **r-NSGA-II versus R-NSGA-II**

In this subsection, we compare r-NSGA-II to R-NSGA-II approach of Deb et al. (2006). Contrary to r-NSGA-II where preference-based selection is based on the r-dominance relation, R-NSGA-II preference-based selection is made by modifying the crowding distance assignment strategy by emphasizing solutions situated near reference points to survive for the next generations. Additionally, an $\varepsilon$-clearing procedure is used to control the spread of the obtained ROIs by restricting the distance in objective space between two neighboring solutions in the same front to be greater than a user-defined small quantity $\varepsilon$ (cf. section 3.2.4). R-NSGA-II has demonstrated its ability to guide the search towards multiple reference points; however there were difficulties when handling only one unique reference point. For this reason, we make an experiment using a single reference point and compare the obtained results of the two algorithms: (1) r-NSGA-II and (2) R-NSGA-II. We use the 30-variable bi-objective test problem ZDT1 and the reference point (0.2, 0.2). The population size and the number of generations are set to 100 and 500 respectively. The R-NSGA-II clearing parameter and the r-NSGA-II $\delta$ threshold are settled, by trial and error, to 0.001 and 0.1 respectively in order to obtain similar ROI spreads for both algorithms. Figure 5.16 shows the obtained results for the two different preference-based EMO methodologies. We remark that r-NSGA-II ROI lies on the optimal Pareto front. However, R-NSGA-II is unable to reach the Pareto optimal front. This phenomenon may be explicated by two reasons. Firstly, R-NSGA-II guides the search towards the single reference point at the beginning of the search process which reduces the population diversity. The diversity reduction slows down the search process and may cause a premature convergence which makes the MOEA unable to reach the Pareto optimal front. The used $\varepsilon$-clearing procedure seems to be unable to keep the population diversity sufficiently. In fact, Deb et al. (2006a) have mentioned this problem and have suggested the use of extreme points as additional reference points in order to keep population diversity and avoid getting stuck in local optima. The conclusion to draw is that R-NSGA-II must handle more than one reference point to achieve satisfactory results. Secondly, contrary to R-NSGA-II, r-NSGA-II does not concentrate the search in a certain region of the search space at an early stage of the evolutionary process. In fact, the population is guided towards the reference point *gradually* during the MOEA execution by means of the *adaptive* management of the non-r-dominance threshold $\delta$ (cf. section 5.2.5) which preserves population diversity and allows the convergence towards the Pareto optimal front. It is important to note that, unlike R-NSGA-II, the crowding distance assignment used in the original NSGA-II is kept in r-NSGA-II which emphasizes even more the population diversity. In summary, the main advantages of r-NSGA-II over R-NSGA-II

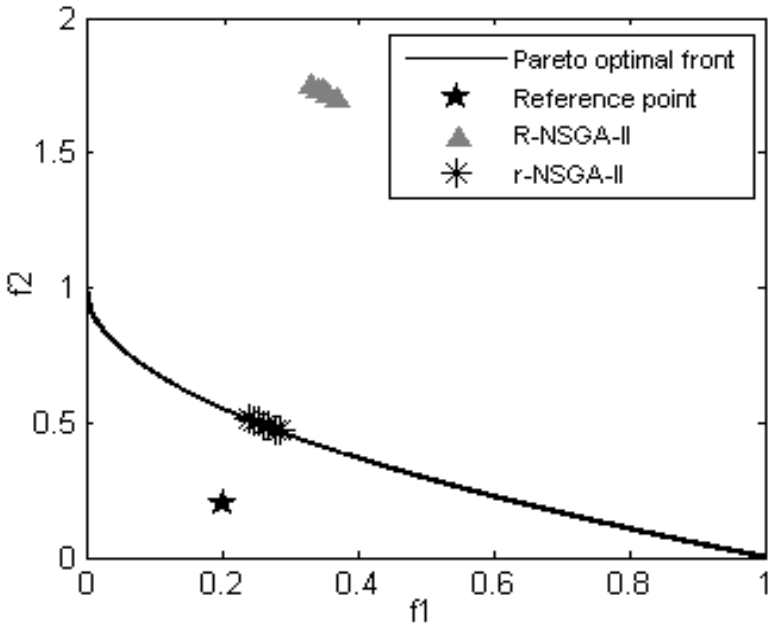



**Figure 5.16 r-NSGA-II versus R-NSGA-II on ZDT1 with the reference point (0.2, 0.2).**

are: (1) the ability of r-NSGA-II to handle a unique reference point where R-NSGA-II faces difficulties in such situation and (2) the easiness of the ROI spread control when using r-NSGA-II since the upper/lower bounds of the non-r-dominance threshold $\delta$ are known ($\delta \in [0,1]$, cf. definition 5.1) which is not the case for R-NSGA-II since setting an appropriate $\varepsilon$-clearing factor is not a trivial task when solving real world problems.

## 5.4 Conclusion

In this chapter, we have suggested a new dominance relation inspired from the Pareto dominance concept and the reference point approach. When incorporated in NSGA-II, the r-dominance has shown its ability to guide the search based on DM's preferences towards the preferred parts of the Pareto optimal front on a reasonable number of FEs. The spread of the obtained ROI could be easily controlled by the non-r-dominance threshold $\delta$. Bias on certain objectives could be achieved by varying the ASF weight vector. The r-NSGA-II has also the ability to handle multiple reference points. The usefulness of our new proposed approach has been demonstrated a priori and interactively on a set of two- to ten-objective test problems. Moreover, our approach has shown competitive and better results when confronted to three recently proposed reference point-based EMO approaches. In the next chapter, we focus on searching for special points of the Pareto front which could be considered as implicit DM's preferences, i.e., DM's preferences when the DM does not specify explicit preferences.

# Chapter 6

# Incorporating Implicit DM's Preferences in Evolutionary Multi-objective Optimization

## 6.1 Introduction

As noted in the fourth chapter, in the absence of explicit DM's preference information, there exist special points of the Pareto front that could represent implicitly preferred parts of the optimal frontier for the DM, which are: (1) *knee regions* and (2) *nadir point*. This chapter is devoted to present two contributions. The first one corresponds to a preference-based MOEA which approximates knee regions. The second one corresponds to a preference-based MOEA which approximates nadir objective values. The rest of this chapter is structured as follows. The second section describes our proposed approach for knee region approximation. The third section presents our proposed algorithm for nadir point estimation. In each of these two sections, we describe the proposed methods and we provide comparative experiments that allow validating them regarding the most prominent works in the corresponding specialized literature. The last section concludes the chapter.

## 6.2 Proposed methods for knee region approximation

### 6.2.1 Knee-based R-NSGA-II (KR-NSGA-II)

- **Algorithmic description**

Before presenting the algorithmic details of KR-NSGA-II (Bechikh et al. 2010a), we describe the knee point characterization adopted in our algorithm. In fact, we choose the characterization of Das (1999) which is illustrated in figure 6.1 for the convex case and the concave one. In the bi-objective case, the knee of the Pareto front corresponds to the farthest solution from the extreme line $L^*$. The extreme line is the line defined by the extreme solutions $s_1^*$ and $s_2^*$ (i.e., solutions having minimal objectives values). According to Das, the knees correspond to the maximum bulges of the convex/concave parts of the Pareto front of a MOP. However, the DM is interested only to knees

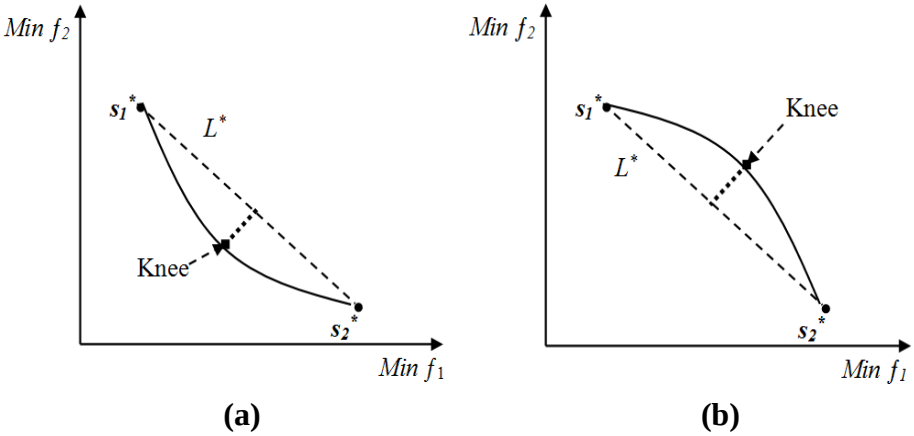



**Figure 6.1 Das characterization of the concept of knee for: (a) the convex case and (b) the concave case.**

situated in convex parts since these latter have the maximal values in terms of the marginal rate of return (Das 1999). Hence, knees situated in concave parts are to be discarded. Consequently, we are interested only to convex bulges (figure 6.1(a)) and not to concave ones (figure 6.1(b)).

KR-NSGA-II is an extension of the R-NSGA-II (Deb et al. 2006a) using *mobile* reference points. We call *mobile reference point* a reference point that is updated automatically in each generation of the MOEA. In the original version of R-NSGA-II, the reference points are to be supplied by the DM a priori or interactively. However, in KR-NSGA-II, the reference points are picked from the *first* non-dominated front automatically via an updating strategy in each generation of the MOEA. We call this selection strategy *Mobile Reference Points Updating Strategy (MRPUS)*. The MRPUS algorithm is presented in figure 6.2. First, the MRPUS considers extreme solutions of the first non-dominated front as reference points since these solutions define the extreme line $L^*$. Hence, we assign infinite distances to the extreme solutions (lines 09-12). Besides, we compute the Cartesian coordinates of the extreme line $L^*$ defined by the extreme solutions. These coordinates serve to compute the distance of a certain solution to the extreme line (line 13). After assigning each solution its distance from $L^*$ (lines 15-17), the MRPUS procedure searches for the *KN* farthest solutions from the extreme line $L^*$ situated in the convex parts of the Pareto front (where *KN* is a user-supplied parameter indicating the number of knees that the DM is searching for). Thus, the cardinality of the mobile reference points set MRP is *KN+2*. The MRPUS updates the reference points in such a way they are not pairwise $\xi$-duplicates (lines 18-28). Two solutions are $\xi$-duplicates if they have a normalized difference in the objective space less than or equal to $\xi$. This is an important feature of the updating strategy since it ensures that only a unique reference point resides in each knee region. This fact avoids

**MRPUS algorithm**

```
01. Input
02. M: the objectives number
03. KN: the knees number
04. FF: the first non-dominated front
05. OMRP: the old mobile reference point set
06. Output
07. MRP: the updated mobile reference point set
08. Begin
09.   ES ← extreme_solutions (FF, M);
10.   For i = 1 to size(ES) do
11.       ES(i).distance ← Inf;
12.   End For
13.   L* ← cartesian_coordinates (ES);
15.   For i = 1 to size(FF) do
16.       FF(i).distance ← distance_to_L* (FF(i), L*);
17.   End For
18.   Sorted_FF ← Sort (FF, 'descend');
19.   j ← 1;
20.   k ← 1;
21.   While (k <= KN+2) and (j <= size (FF)) do
22.       If (NOT(is_ξ_duplicate (sorted_FF(j), OMRP)) Then
23.               MRP(k) ← sorted_FF(j);
24.               OMRP(k) ← MRP(k);
25.               k ← k+1;
26.       End If
27.       j ← j +1;
28.   End While
29.   If (size (MRP) < KN+2) Then
30.       For i = (size (MRP) + 1) to KN+2 do
31.               MRP(i) ← sorted_FF(i);
32.       End For
33.   End If
34. End
```

**Figure 6.2 The MRPUS algorithm.**

the premature convergence of the mobile reference points towards the same knee to occur. The control of $\xi$-duplicates reference points provides an equal emphasis of solutions closest to each reference point situated in each *distinct* knee, thereby allowing multiple knee regions to be found simultaneously in a single simulation run. Figure 6.3 illustrates the functioning principle of the MRPUS algorithm for the bi-objective case with two knee regions. First, extreme solutions *A* and *B* are selected in order to preserve population diversity. After that, the algorithm selects the point *C* as a

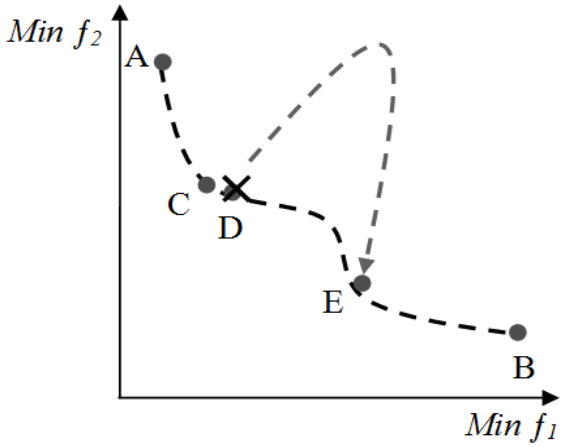



**Figure 6.3 Illustration of the selection of the mobile reference points with the MRPUS for the bi-objective case with two knees.**

reference point since it has the maximal distance from the extreme line $L^*$. In order to not to converge to the same knee region, the algorithm ignores the point *D* since it is a $\xi$-duplicate with the already selected point *C* and jumps directly to the point *E* which permits the discovery of a new knee region. The control of $\xi$-duplicates reference points provides an equal emphasis of solutions closest to each reference point situated in each distinct knee, thereby allowing multiple knee regions to be found. For the case where the DM supplies a knees number (*KN*) greater than the existing knees in the Pareto front of the MOP under consideration, the MRPUS procedure permits to $\xi$-duplicates reference points to occur (i.e., reference points residing in the same knee) (lines 29-33). Besides, for the case where the DM specifies the parameter *KN* in such a way *KN* is less than the existing knees in the Pareto front of the MOP, we remediate this problem, as will be shown later, by making the KR-NSGA-II algorithm *interactive*. Hence, the DM may exploit its acquired knowledge about the search space during the interactive run of the algorithm in order to explore all the existing knee regions. It should be noted that the extent of the knee regions is controlled by means of the clearing parameter $\varepsilon$ (used in R-NSGA-II) since a knee region in KR-NSGA-II corresponds to a ROI in R-NSGA-II. Additionally, it is worth noting that the MRPUS is scalable with the number of objectives. The extreme line $L^*$ can be replaced by: (1) the *extreme plane* $P^*$ defined by *three* extreme solutions for the *tri-objective* case and (2) the *extreme M-dimensional hyper-plane* $HP^*$ defined by *M* extreme points for the *M-objective* case.

In the following, we explain how we assign for each solution its distance to the extreme line $L^*$. Mathematically, the distance from a given point $A(x_A, y_A)$ to a given line

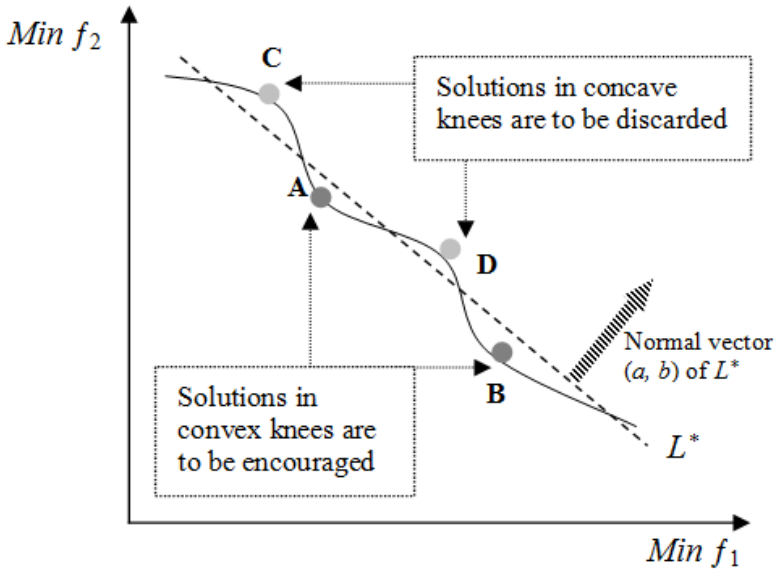



**Figure 6.4 Encouraging the discovery of convex knee regions.**

$D: ax + by + c = 0$ is expressed by the following formula (Smedley and Wiseman 2001):

$$d(A, D) = \frac{|ax_A + by_A + c|}{\sqrt{a^2 + b^2}} \tag{6.1}$$

The sign of $s = ax + by + c$ determines on which side the point $(x, y)$ lies with respect to the line. If $s > 0$ then the point lies on the same side as the normal $(a, b)$. If $s < 0$ then it lies on the opposite side. Finally, if $s = 0$ then the point $(x, y)$ lies on the line. Since in KR-NSGA-II we are interested only to convex knee regions, the distance from a solution $Z(x_z, y_z)$ to the extreme line $L^*: ax + by + c = 0$ is computed as follows:

$$d(Z, D) = \begin{cases} \dfrac{|ax_Z + by_Z + c|}{\sqrt{a^2 + b^2}} & \text{if} \quad ax_Z + by_Z + c < 0 \\ -\dfrac{|ax_Z + by_Z + c|}{\sqrt{a^2 + b^2}} & \text{otherwise} \end{cases} \tag{6.2}$$

According to equation (6.2), solutions residing in concave knees (for the case of minimization problems) are discouraged to remain in the race by assigning them the opposite of their distances to the extreme line. This phenomenon is illustrated by figure 6.4. From this figure, solutions *C* and *D* are residing in concave knee regions. Hence, they are to be discarded by assigning them the opposite of their distances (i.e., negative distances). However, solutions *A* and *B* residing in convex knees are to be preserved to the next generations. Analogously, such computations can be extended for the

*M*-dimensional (*M*-objective) case (Smedley and Wiseman 2001) which makes the KR-NSGA-II able to focus on knees for problem with more than two objectives. It is important to note that, based on the knee characterization adopted in KR-NSGA-II, concave knees do not always occur in a Pareto front containing knees.

♦ **Experimental results**

This subsection is devoted to demonstrate simulation results on two- and three-objective knee-based test problems using the KR-NSGA-II algorithm. All experiments are made with MATLAB[1] software. For each test problem, we show the mean of the obtained results over 20 independent simulations runs. In all simulations, we use the SBX operator with a distribution index of 10 and polynomial mutation with a distribution index of 20 (Deb and Agrawal 1995). The crossover and mutation probabilities are set to 0.9 and 1/*n* (where *n* is the number of decision variables) respectively. The population size is set to 100 for the bi-objective case and 200 for the tri-objective case. For all experiments, the $\xi$ parameter is set to $10^{-2}$. The knee-based test problems used in this chapter are described in appendix B. Inspired from the DTLZ problems (Deb et al. 2002b), Branke et al. (2004) designed three knee-based test problems named DO2DK, DEB2DK and DEB3DK in order to assess their knee-based MOEAs. DO2DK and DEB2DK are two bi-objective problems with *n* decision variables. They have a settable parameter *K* controlling the number of knees in the front. DO2DK has an additional parameter *s* controlling the skew of the front. DEB3DK is a tri-objective problem where, like DEB2DK, the parameter *K* expresses the number of knee regions in the Pareto surface. Based on these three test problems, Rachmawati and Srinivasan (2009) created other benchmarks: DO2DK-1, DEB2DK-1 and DEB3DK-1. These problems are modified versions of DO2DK, DEB2DK and DEB3DK respectively. In fact, the $g(x)$ function is updated in order to impose a greater density of solutions away from the optimal front and vice versa. This non-uniformity of the search space challenges the MOEA's ability to progress towards the Pareto front. DEB2DK-2 is a modified DEB2DK where the Pareto front is discontinuous and contains a bias in the solution distribution along the front in addition to knees in concave parts. All bi-objective test problems are implemented with $K = 1$ to 4. The parameter *s* is set to 0 for $K = 1$ and to 1 for $K > 1$. Tri-objective problems are implemented with $K = 1$ to 2. In total, 24 problem instances are used in our study.

- **A priori simulation results**

Firstly, we consider the DO2DK test problem. Figure 6.5 shows the obtained solutions after performing 100 generations (i.e., 10000 FEs) with $\varepsilon = 0.0001$, $KN = K = 4$ and

[1] The used version is MATLAB 7.4 (www.mathworks.com).

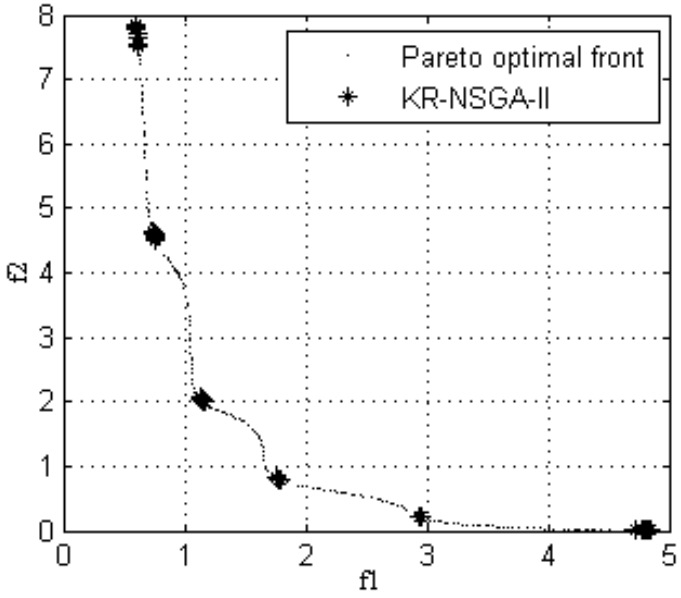



**Figure 6.5 Obtained solutions with DO2DK test problem with 4 knees.**

$s$ = 1.0. We remark that KR-NSGA-II has the ability to focus on the four knee regions of the DO2DK problem while preserving extreme solutions. As has been discussed before, extreme solutions are considered as reference points in KR-NSGA-II since these solutions play a crucial role in: (1) the determination of the extreme line $L^*$ and consequently in the detection of knees and (2) ensuring the population diversity and consequently preserving the reference point set diversity.

Next, we consider the DEB2DK test problem. Figure 6.6 demonstrates the effect of varying the cleaning parameter $\varepsilon$ on this bi-objective problem, with $KN = K = 4$, after performing 100 generations. We remark that KR-NSGA-II is able to find solutions located in the four knee regions in addition to the extreme optimal solutions. Besides, for $\varepsilon$ = 0.0001, we remark that population individuals converge sharply towards the center of each knee region in addition to extreme solutions. Solutions with other $\varepsilon$ values are shown with an offset to the true Pareto optimal front for ease of visualization. We notice that the range of the obtained solutions increases with the increase of the parameter $\varepsilon$. Hence, if the DM would like to obtain a large neighbourhood of solutions near each knee center, a large value of $\varepsilon$ should be chosen. We conclude that the DM could control the spread of the obtained ROIs (i.e., knee regions) by means of the cleaning parameter $\varepsilon$.

Finally, we consider the tri-objective DEB3DK problem. Figure 6.7 demonstrates the KR-NSGA-II obtained solutions after performing 250 generations (i.e., 25000 FEs) with $\varepsilon = 0.0001$ and $KN = K = 1$. It is clear, from this 3D-plot, that the KR-NSGA-II has the ability to focus on the knee region(s) in the three-objective case in addition to

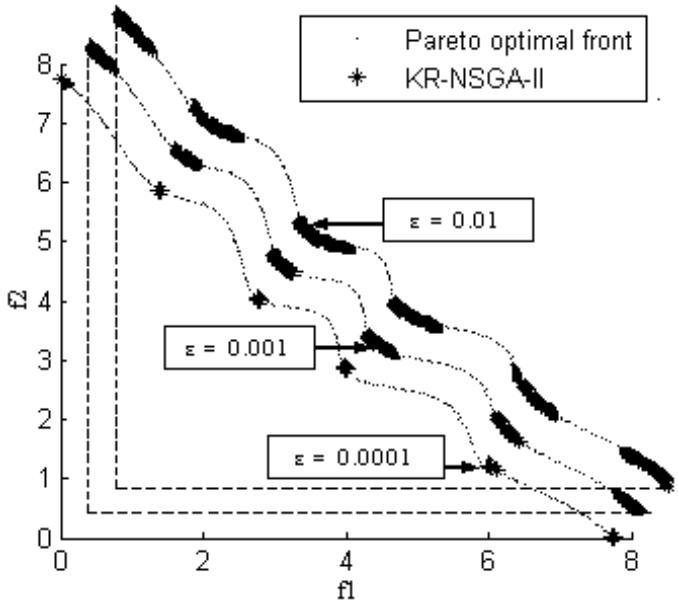



**Figure 6.6 Effect of varying the ε parameter on the DEB2DK test problem with 4 knees.**

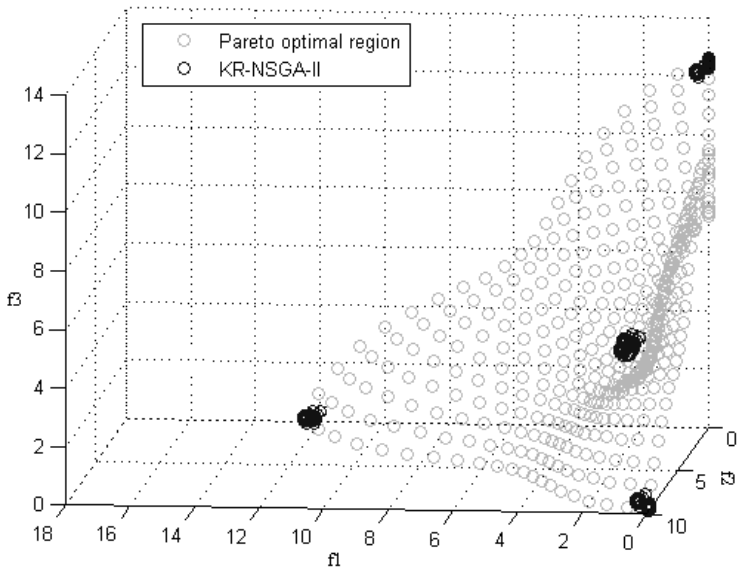



**Figure 6.7 Obtained solutions with the tri-objective DEB3DK test problem with one knee.**

finding extreme solutions. It is worth noting here that the distance computations are done according to the extreme plane $P^*$ (defined by three extreme solutions) instead of the extreme line $L^*$ as discussed previously.

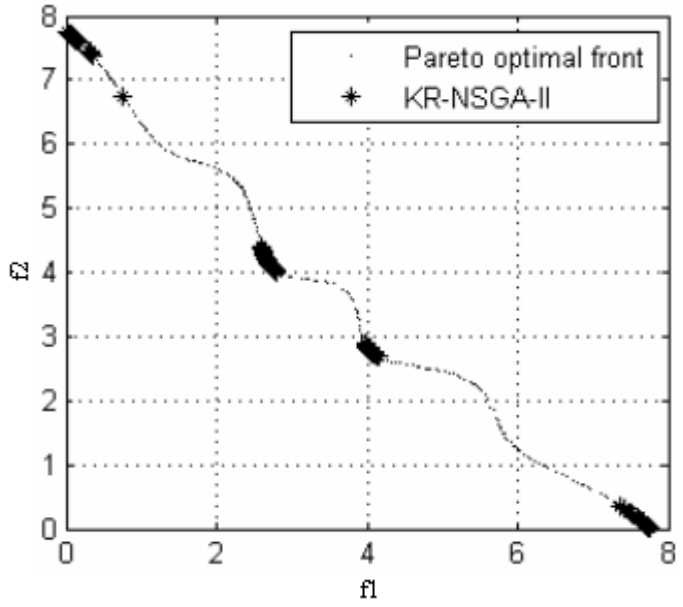
- **An interactive scenario**

In this subsection, we propose an interactive scenario for the KR-NSGA-II algorithm. This scenario is useful especially when the DM has no idea about the number of the existing knees in the Pareto optimal front of the problem under consideration. The interactive version of KR-NSGA-II can be summarized as follows:

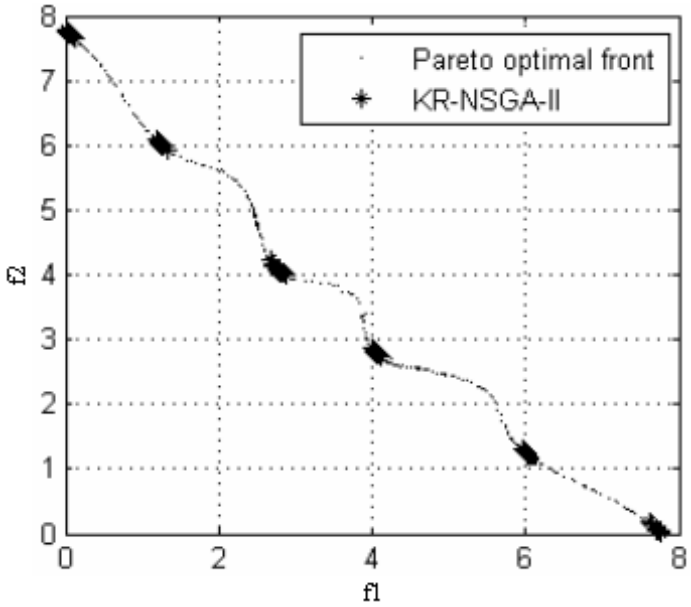
- **Step 1:** Ask the DM to supply the population size, the stopping criterion, the number of knee regions to be discovered (i.e., the *KN* parameter), the $\varepsilon$ parameter and the $\xi$ parameter;
- **Step 2:** Perform the KR-NSGA-II algorithm until the stopping criterion is met;
- **Step 3:** Supply the DM with the set of the obtained solutions. If the DM is satisfied with the supplied set of solutions then stop the optimization process, else ask the DM if he/she would like to update the parameters *KN*, $\varepsilon$, $\xi$ and/or the stopping criterion then return to **Step 2**.

In the following, we illustrate an interactive run of the KR-NSGA-II algorithm on the bi-objective DEB2DK test problem with 4 knees ($K = 4$) using a population of 100 individuals. The parameters $\varepsilon$ and $\xi$ are set initially to 0.0005 and 0.01 respectively. We suppose that the DM has no idea about the number of the existing knees in the optimal Pareto front. The overall interactive run is described by figure 6.8 and it is composed with two cycles. We call *cycle* a run of KR-NSGA-II for a certain number of generations freely fixed by the DM. Firstly, the DM performs an evolutionary cycle of 100 generations with $KN = 2$. The obtained solutions (figure 6.8(a)) show that KR-NSGA-II discovers two knee regions in addition to the extreme solutions. We suppose that the DM is dissatisfied and he/she is interested in finding all existing knees in the Pareto front. Hence, he/she sets the parameter *KN* to 6 and he/she performs a run of 100 generations. According to figure 6.8(b), we remark that the population focus on the four knee regions existing in the Pareto front of DEB2DK problem with four knees. We conclude that, although the DM sets the parameter *KN* to 6, KR-NSGA-II provides 4 knee regions. This fact emphasizes what has been discussed previously in the fourth section, i.e., the MRPUS procedure accepts $\xi$-duplicates reference points if and only if the DM sets $KN > K$ which ensures the discovery of all existing knee regions.

In summary, the KR-NSGA-II has demonstrated its effectiveness in approximating knee regions for the bi- and tri-objective cases. The interactive scenario is shown to be useful to discover all knee regions when the DM ignores the number of existing knees a priori. However, the KR-NSGA-II performance depends on the success of the algorithm in discovering the Pareto optimal extreme solutions which was reported not to be an easy task when estimating the nadir point (Deb and Miettinen 2008).




**(a)**




**(b)**

**Figure 6.8 Interactive scenario on DEB2DK with 4 knees: (a) *KN* = 2 and (b) *KN* = 6.**

### 6.2.2 Trade-off-based KR-NSGA-II (TKR-NSGA-II)

♦ **Algorithmic description**

TKR-NSGA-II (Bechikh et al. 2011a) is an enhanced version of KR-NSGA-II. As noted previously, the optimal extreme point estimation is not an easy task for MOEAs.

Hence, the failure of KR-NSGA-II in discovering the true extreme solutions engenders the loss a potential knee solutions or the discovery of spurious ones. In order to avoid such problematic, we modified the knee point characterisation in KR-NSGA-II. In fact, instead of the distance to the extreme line, we used a recently proposed trade-off worth metric designed by Rachmawati and Srinivasan (2009). In fact, although Pareto optimal solutions are equivalent, the trade-off magnitude varies across the optimal front. Trade-off characterizes two non-dominated objective vectors and can be defined as the net gain of improvement in some objective subset by the accompanying deterioration in some other criteria as a result of substituting an objective vector with another non-dominated one. Equation (3) offers a mathematical definition of the trade-off information for a pair of non-dominated solutions:

$$T(x_i, x_j) = \frac{\sum_{m=1}^{M} \max\left[0, \frac{f_m(x_j) - f_m(x_i)}{f_m^{\max} - f_m^{\min}}\right]}{\sum_{m=1}^{M} \max\left[0, \frac{f_m(x_i) - f_m(x_j)}{f_m^{\max} - f_m^{\min}}\right]} \tag{6.3}$$

We note that $f_m(x_i)$ corresponds to the $m^{th}$ objective value of solution $x_i$ and $f_m^{\max} / f_m^{\min}$ corresponds to the maximal/minimal value of the $m^{th}$ objective in the population individuals. In the above equation, normalization is performed in order to prevent some objectives being predominant over others since objectives are usually incommensurable in real world applications. In equation (5.3), the numerator expresses the aggregated improvement gained by substituting $x_j$ with $x_i$. However, the denominator evaluates the deterioration generated by the substitution. A more concise metric to compute the worth of a solution $x_i$, in terms of trade-off, relative to the set of non-dominated solutions $S$ to which it belongs is given by equation (6.4) (Rachmawati and Srinivasan 2009):

$$\mu(x_i, S) = \underset{x_j \in S, x_i \prec x_j, x_j \prec x_i}{Min} T(x_i, x_j) \tag{6.4}$$

We note that $x_j$ denotes members of the set of non-dominated solutions $S$ that are non-dominated with respect to $x_i$. The quantity $\mu(x_i, S)$ expresses the least amount of improvement per unit deterioration by substituting any alternative $x_j$ from $S$ with $x_i$. Solutions residing in convex knee regions have the highest values in terms of the trade-off metric $\mu$. Such characteristic makes knee regions almost always very important to the DM in practical context.

**T-MRPUS algorithm**

```
01. Input
02. M: the objectives number
03. KN: the knees number
04. FF: the first non-dominated front
05. OMRP: the old mobile reference point set
06. Output
07. MRP: the updated mobile reference point set
08. Begin
09.   ES ← extreme_solutions (FF, M);
10.   MRP ← ES;
11.   For i = 1 to size(FF) do
12.       FF(i).trade-off_worth ← evaluate_trade-off_metric (FF(i), FF);
13.   End For
14.   sorted_FF ← Sort (FF, 'descend');
15.   j ← 1;
16.   k ← 1;
17.   While (k <= KN) and (j <= size (FF)) do
18.       If (NOT(is_ξ_duplicate (sorted_FF(j), OMRP)) Then
19.               MRP(k) ← sorted_FF(j);
20.               OMRP(k) ← MRP(k);
21.               k ← k+1;
22.       End If
23.       j ← j +1;
24.   End While
25.   If (size (MRP) < KN) Then
26.       For i = (size (MRP) + 1) to KN do
27.               MRP(i) ← sorted_FF(i);
28.       End For
29.   End If
31. End
```

**Figure 6.9 The T-MRPUS algorithm.**

The TKR-NSGA-II is the result of substituting the distance from the extreme line computation in KR-NSGA-II by the trade-off worth measure. This modification is performed in the MRPUS algorithm. The new MRPUS procedure, called Trade-off-based MRPUS (T-MRPUS), is presented by figure 6.9. We see from this figure, that MRPUS and T-MRPUS are nearly the same; just the knee point characterization is changed. The T-MRPUS updates the mobile reference points based on the trade-off worth metric values which computation is independent of the discovery of the true extreme solutions. Solutions having highest values in terms of trade-off worth are emphasized. The control of $\xi$-duplicate reference points (cf. figure 6.3) is also carried

**TKR-NSGA-II basic iteration algorithm**

01. **Input**
02. $P_t$ : the parent population at generation t
03. $Q_t$ : the offspring population at generation t
04. M : the number of objective functions
05. **Output**
06. $P_{t+1}$ : the updated parent population at generation t+1
07. $Q_{t+1}$ : the updated offspring population at generation t+1
08. **Begin**
09. $R_t \leftarrow P_t \cup Q_t$;
10. $R_t \leftarrow$ non-domination_sort ($R_t$, M);
11. FF $\leftarrow$ select_best_front ($R_t$);
12. MRP $\leftarrow$ T-MRPUS (M, KN, FF, MRP);
13. $R_t \leftarrow$ reference_point_based_crowding ($R_t$, MRP);
14. $R_t \leftarrow$ ε-clearing ($R_t$);
15. $P_{t+1} \leftarrow$ environmental_selection ($R_t$);
16. $Q_{t+1} \leftarrow$ reproduction ($P_{t+1}$);
17. t $\leftarrow$ t+1;
18. **End**

**Figure 6.10 The TKR-NSGA-II basic iteration.**

out by the T-MRPUS in order to ensure the discovery of all existing knee regions. Additionally, it is worth noting that T-MRPUS is scalable with the number of objectives since the trade-off worth metric $\mu$ is independent of the number of criteria to optimize.

Figure 6.10 presents the basic iteration of the TKR-NSGA-II algorithm. Firstly, the parent population $P_t$ and the offspring population $Q_t$ are merged to form the combined population $R_t$ (line 9). Besides, a non-dominated sorting is applied to the population $R_t$ (line 10). Hence, the population $R_t$ becomes subdivided into several fronts. Then, the best front *FF* is selected (line 11) and passed as input parameter to the T-MRPUS procedure in order to generate the new mobile reference point set *MRP* (line 12). We note that the *MRP* set passed as input parameter (line 12) to the T-MRPUS procedure corresponds to the old mobile reference point set (i.e., the *MRP* of the previous generation). Once the *MRP* set is generated, we assign to each solution, front wise, its crowding distance based on its distance from the mobile reference points (line 13). Since solutions near reference points have better crowding factors, the population is guided gradually towards knee centers; which makes the search converging to knee regions. In order to ensure the control of the extent of the obtained knee regions and promote further population diversity, a clearing procedure is applied to $R_t$ (line 14).

This procedure is applied front wise and ensures that the minimal allowed distance between two objective vectors having the same non-domination rank is greater than the user-supplied quantity $\varepsilon$. Finally, the new parent population $P_{t+1}$ for the next generation is created by performing environmental selection on $R_t$ (line 15), the new offspring population $Q_{t+1}$ for the next generation is produced by applying genetic operators to $P_{t+1}$ (line 16) and the evolutionary process is repeated until the stopping criterion is met.

### ♦ Experimental results

This subsection is devoted to confront TKR-NSGA-II to the most representative works in this research area. We note that the TKR-NSGA-II has demonstrated similar results to KR-NSGA-II ones not only on a priori experiments but also on an interactive run (Bechikh et al. 2011a). Before presenting comparative experiments, we would like to discuss an important issue here which is the computational cost of TKR-NSGA-II. In fact, knee regions can be obtained by approximating the complete Pareto front and then knee regions can be detected, in a posteriori manner, by applying any *trade-off metric* on the final population (e.g., the smart Pareto filter (Mattson et al. 2004)) which goes without any further function call. We note that, after applying the NSGA-II to the DEB2DK test problem, the required number of FEs to approximate the whole Pareto front was about 50000 FEs. This fact shows the efficiency of TKR-NSGA-II over its base MOEA NSGA-II since it can provide optimal knee regions after 10000 FEs. Consequently, TKR-NSGA-II has two main advantages over NSGA-II. On one hand, it provides the DM with the maximal trade-offs which facilitates his/her task when selecting the final alternative. On the other hand, it reduces the required computational cost significantly. Additionally, similar observations were seen for the other test problems which proves the efficiency of TKR-NSGA-II over NSGA-II from a computational cost viewpoint and hence the need for such preference-based MOEA.

- **Assessing converging towards knee regions**

In this subsection, we compare the TKR-NSGA-II with the most representative works in this research area from a convergence viewpoint. TKR-NSGA-II is confronted to the Marginal Utility Approach (*MUA*) (Branke et al. 2004), the Weighted sum Niching Approach (*WSNA*) (Rachmawati and Srinivasan 2006b), the $S_{p2}$ version of the Parallel Local Weighted Sum Optimization approach (*PLWSO*) (Rachmawati and Srinivasan 2009) and KR-NSGA-II (Bechikh et al. 2010a). Since MOEAs focusing on knee regions are preference-based MOEAs, we cannot use the hypervolume metric because the dominated portion of the objective space depends on the distribution of the obtained solutions. For this reason, we use another well-cited metric, i.e., the generational

**Table 6.1 MOEA specific parameter settings.**

| Algorithm | MOEA specific parameter values |
|---|---|
| WSNA | $Q$ = 100, $P$ =20 |
| PLWSO | $Q$ = 100, $\delta'$ = 0.1, $S_{P2}$ = 80% |
| KR-NSGA-II | $\xi$ = 0.05 for bi-objective case and 0.08 for tri-objective case, |
| TKR-NSGA-II | $\varepsilon$ = 0.001, $KN = K$ |

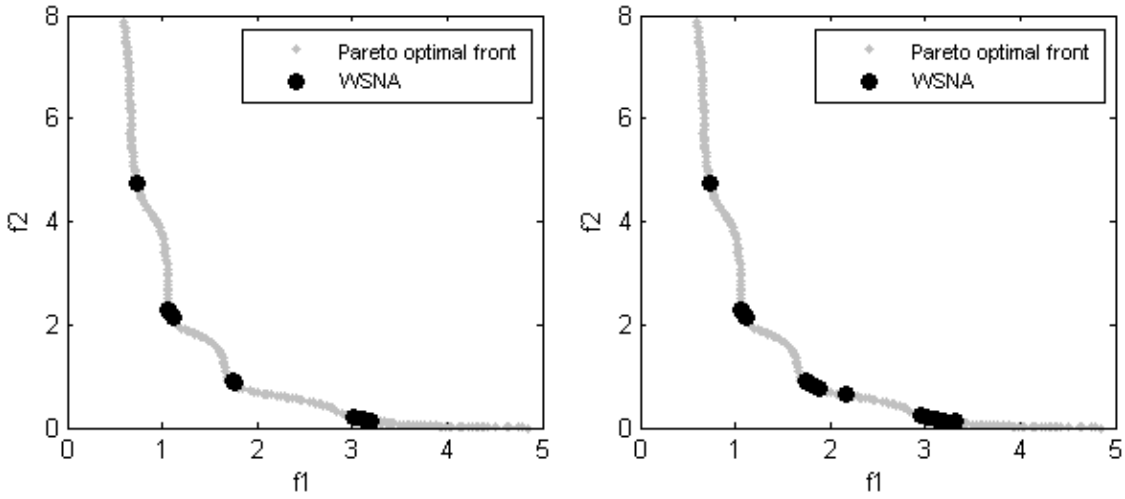
distance *GD* which expresses the proximity of the obtained solution sets to the Pareto optimal front. We recall that this performance measure is given by the average Euclidean distance separating the obtained solutions and the nearest members of a uniformly distributed reference set taken from the Pareto front. In this study, we use a reference Pareto front of size 500 for the bi-objective problems and 900 for the tri-objective problems. We use the same experimental design of the study of Rachmawati and Srinivasan (2009) in order to make fair comparisons. A set of 10 simulation runs was done for each pair (*MOEA*, *problem instance*) under MATLAB software. For all MOEAs, the termination criterion is set to 20000 FEs for DO2DK and DO2DK-1, 25000 FEs for DEB2DK, DEB2DK-1 and DEB2DK-2 and 30000 FEs for DEB3DK and DEB3DK-1 unless otherwise specified. The used genetic operators are the simulated binary crossover (SBX) and the polynomial mutation (Deb and Agrawal 1995) with crossover probability of 0.9 and mutation probability of 1/*n* (where *n* is the number of decision variables). The population size is settled to 100 for all problem instances. The other MOEA-specific parameter values used in this study are indicated in table 6.1 unless otherwise specified. We note that the marginal utility approach does not have specific parameters. The significance of the parameters figuring in table 6.1 is detailed in subsection 6.2.1 and the references therein.

Table 6.2 presents the *GD* values for the five algorithms under comparison. We note that we exploit the results published in (Rachmawati and Srinivasan 2009). We remark, from this table, that for a Pareto front containing a unique knee region, the reference point-based algorithms (i.e., KR-NSGA-II and TKR-NSGA-II) present better convergence than the three other weighted sum-based algorithms. This observation may be explained by the fact that the reference point-based methods preserve the survival of extreme solutions which promotes population diversity and hence emphasizes convergence towards the Pareto optimal front. However, the three weighted sum-based algorithms focus the search only towards the unique knee region which may reduce population diversity and hence deemphasizing convergence. For *K* greater than one, we see that (1) the KR-NSGA-II algorithm performs better than the MUA and the WSNA and (2) TKR-NSGA-II presents better results than the four other algorithms on *most* problems. The superiority of TKR-NSGA-II over KR-NSGA-II may be explained by

**Table 6.2 Generational distance values.**

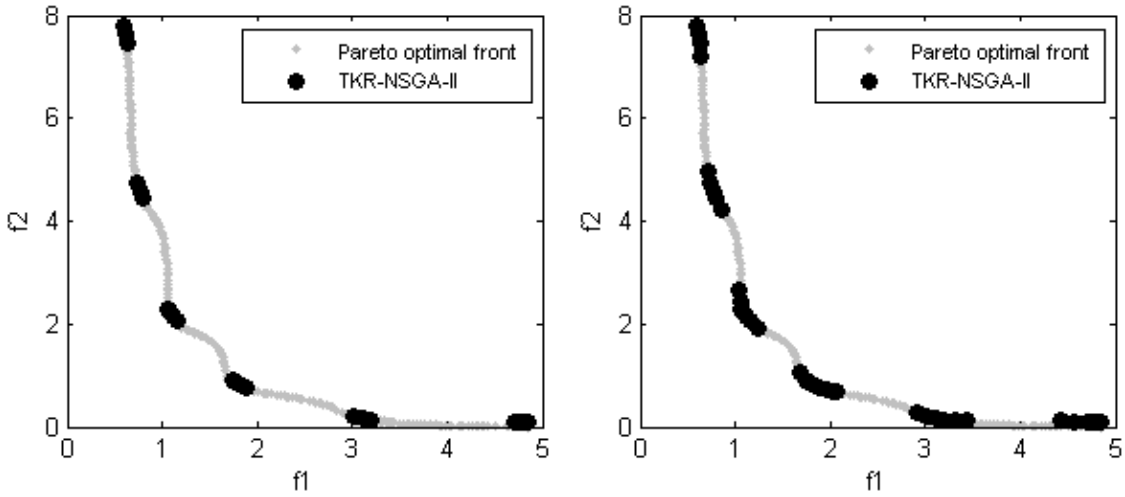
| Problem | K | MUA | | WSNA | | PLWSO | | KR-NSGA-II | | TKR-NSGA-II | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Mean | SD | Mean | SD | Mean | SD | Mean | SD | Mean | SD |
| DO2DK | 1 | 0.03505 | 0.01858 | 0.00437 | 0.00003 | 0.00329 | 0.00024 | 0.00311 | 0.00086 | 0.00299 | 0.00031 |
| | 2 | 0.02497 | 0.00628 | 0.00396 | 0.00043 | 0.00290 | 0.00066 | 0.00353 | 0.00039 | 0.00281 | 0.00054 |
| | 3 | 0.02511 | 0.00360 | 0.00463 | 0.00038 | 0.00365 | 0.00029 | 0.00374 | 0.00048 | 0.00317 | 0.00033 |
| | 4 | 0.02305 | 0.00305 | 0.00483 | 0.00073 | 0.00437 | 0.00036 | 0.00411 | 0.00039 | 0.00398 | 0.00046 |
| DO2DK-1 | 1 | 0.24272 | 0.60024 | 0.00442 | 0.00023 | 0.00339 | 0.00018 | 0.00334 | 0.00023 | 0.00308 | 0.00037 |
| | 2 | 0.01420 | 0.01566 | 0.00361 | 0.00017 | 0.00287 | 0.00018 | 0.00367 | 0.00029 | 0.00291 | 0.00017 |
| | 3 | 0.06192 | 0.10532 | 0.00348 | 0.00026 | 0.00445 | 0.00092 | 0.00331 | 0.00041 | 0.00315 | 0.00081 |
| | 4 | 0.00696 | 0.00781 | 0.00363 | 0.00025 | 0.00457 | 0.00168 | 0.00391 | 0.00046 | 0.00338 | 0.00018 |
| DEB2DK | 1 | 0.04060 | 0.00708 | 0.42352 | 0.39414 | 0.01624 | 0.00125 | 0.14971 | 0.00549 | 0.01232 | 0.00271 |
| | 2 | 0.02497 | 0.00628 | 0.00396 | 0.00043 | 0.00290 | 0.00066 | 0.00303 | 0.00052 | 0.00238 | 0.00061 |
| | 3 | 0.02511 | 0.00360 | 0.00463 | 0.00038 | 0.00365 | 0.00029 | 0.00501 | 0.00074 | 0.00294 | 0.00033 |
| | 4 | 0.01368 | 0.00139 | 0.31743 | 0.47184 | 0.00674 | 0.00114 | 0.01883 | 0.00211 | 0.00731 | 0.00319 |
| DEB2DK-1 | 1 | 0.24272 | 0.60024 | 0.54071 | 0.74018 | 0.00339 | 0.00018 | 0.00334 | 0.00021 | 0.00304 | 0.00014 |
| | 2 | 0.01420 | 0.01566 | 0.00361 | 0.00017 | 0.00287 | 0.00018 | 0.0405 | 0.00033 | 0.00229 | 0.00026 |
| | 3 | 0.06192 | 0.10532 | 0.00348 | 0.00026 | 0.00445 | 0.00092 | 0.00389 | 0.00051 | 0.00377 | 0.00081 |
| | 4 | 0.03299 | 0.00573 | 2.78440 | 3.66200 | 0.00666 | 0.00129 | 0.00591 | 0.00098 | 0.00517 | 0.00034 |
| DEB2DK-2 | 1 | 0.03446 | 0.00295 | 0.03669 | 0.00491 | 0.02538 | 0.00143 | 0.02511 | 0.00166 | 0.00219 | 0.00118 |
| | 2 | 0.03393 | 0.00351 | 0.04150 | 0.00532 | 0.01883 | 0.00096 | 0.02239 | 0.00382 | 0.01890 | 0.00102 |
| | 3 | 0.03262 | 0.00293 | 0.04632 | 0.00368 | 0.01439 | 0.00216 | 0.03848 | 0.00329 | 0.01511 | 0.00198 |
| | 4 | 0.03367 | 0.00280 | 0.04783 | 0.00388 | 0.01279 | 0.00187 | 0.04872 | 0.00196 | 0.01324 | 0.00136 |
| DEB3DK | 1 | 0.17511 | 0.11718 | 0.13382 | 0.02361 | 0.11368 | 0.01958 | 0.10322 | 0.02335 | 0.09461 | 0.01992 |
| | 2 | 0.76320 | 0.25049 | 0.29348 | 0.07652 | 0.32930 | 0.15690 | 0.33782 | 0.14088 | 0.29917 | 0.12063 |
| DEB3DK-1 | 1 | 0.13239 | 0.10110 | 2.31280 | 0.35338 | 0.11282 | 0.04535 | 0.10074 | 0.08093 | 0.09567 | 0.06511 |
| | 2 | 2.18844 | 0.60789 | 12.100 | 1.04223 | 0.27393 | 0.16129 | 0.32096 | 0.03027 | 0.25476 | 0.04258 |

the fact that the discovery of knee region centers does not depend on the success of the algorithm in finding the optimal extreme solutions which is the case for KR-NSGA-II. In fact, TKR-NSGA-II updates the mobile reference point set based on the trade-off worth metric $\mu$ whose computation does not necessitate the discovery of extreme points which is not an easy task for MOEAs as reported in (Deb et al. 2009a). The TKR-NSGA-II considers extreme objective vectors as reference points only to promote population diversity and consequently enhancing convergence. The superiority of TKR-NSGA-II over PLWSO on most test problems may be explained by the fact that the output provided by PLWSO depends heavily on the weight sets generated during the first step of the optimization process (since PLWSO is a two-step method where the first step is devoted to generate the adequate weight sets). However, TKR-NSGA-II is a one-step method which guides the search directly towards knee regions at an early stage of the evolutionary process while preserving population diversity. Besides, since PLWSO is reported to perform better than WSNA and MUA in the study of Rachmawati and Srinivasan (2009), by transitivity TKR-NSGA-II is declared to be




**(a)** **(b)**

**Figure 6.11 WSNA spread control on DO2DK with $K$ = 4: (a) $Q$ = 100, $P$ = 30 and (b) $Q$ = 100, $P$ = 50.**




**(a)** **(b)**

**Figure 6.12 TKR-NSGA-II spread control on DO2DK with $K$ = 4: (a) $\varepsilon$ = 0.001 and (b) $\varepsilon$ = 0.01.**

superior to MUA and WSNA. The obtained results presented in table 6.2 emphasize this statement.

- **Assessing knee region spread control**

Obtaining a small neighborhood of solutions in the vicinity of the knee center is more interesting for the DM than obtaining a single solution. The neighborhood size should be well-controlled by the DM in order to have a clear idea about the worthiest solutions

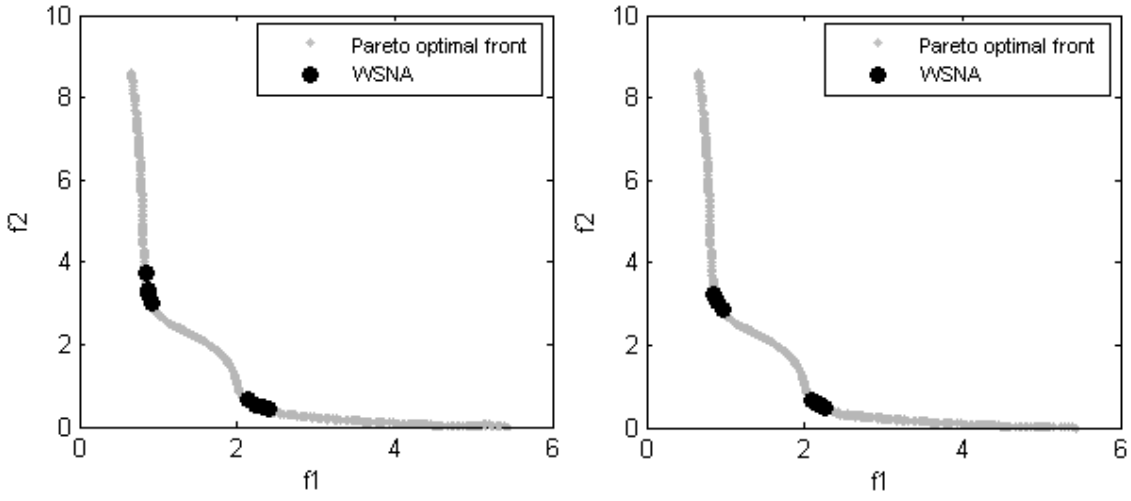
in terms of trade-off. For this reason, in this subsection, we assess the ability of the algorithms to control the extent of the furnished knee regions. The MUA is excluded from the comparison since it does not allow controlling the knee region spread. Hence, the TKR-NSGA-II is confronted only to WSNA, PLWSO and KR-NSGA-II.

- **TKR-NSGA-II versus WSNA**

WSNA allows controlling the spread of the obtained regions by means of the parameters $Q$ and $P$ (Rachmawati and Srinivasan 2006b). Figure 6.11 shows the effects of varying the parameter $P$ on the DO2DK test problem with four knees. We see from this figure that the increase of $P$ from 30 (figure 6.11(a)) to 50 (figure 6.11(b)), with $Q$ kept constant at 100, increases the neighborhood extent in each knee region. This increase does not apply uniformly over all knees which may mislead the DM in practical applications. However, TKR-NSGA-II does not face such problem. We see that the increase of $\varepsilon$ from 0.001 (figure 6.12(a)) to 0.01 (figure 6.12(b)) increases the extent of the obtained knee regions in addition to the neighborhood spread near each extreme solution. Additionally, this increase applies uniformly over all existing knees which provides the DM with a clear idea about the maximal trade-offs of the Pareto front. Figure 6.13 demonstrates the effect of increasing the parameter $P$ of WSNA from 30 (figure 6.13(a)) to 50 (figure 6.13(b)), with $Q$ kept constant at 100, on the DO2DK test problem with two knees. We see, from this figure, that changing the parameter $P$ has no effect on the spread of the knee regions. All obtained knee regions have approximately the same breadth. We conclude that the spread control by WSNA depends on the geometrical shape of the Pareto optimal front which is not the case for TKR-NSGA-II as shown by figure 6.14.

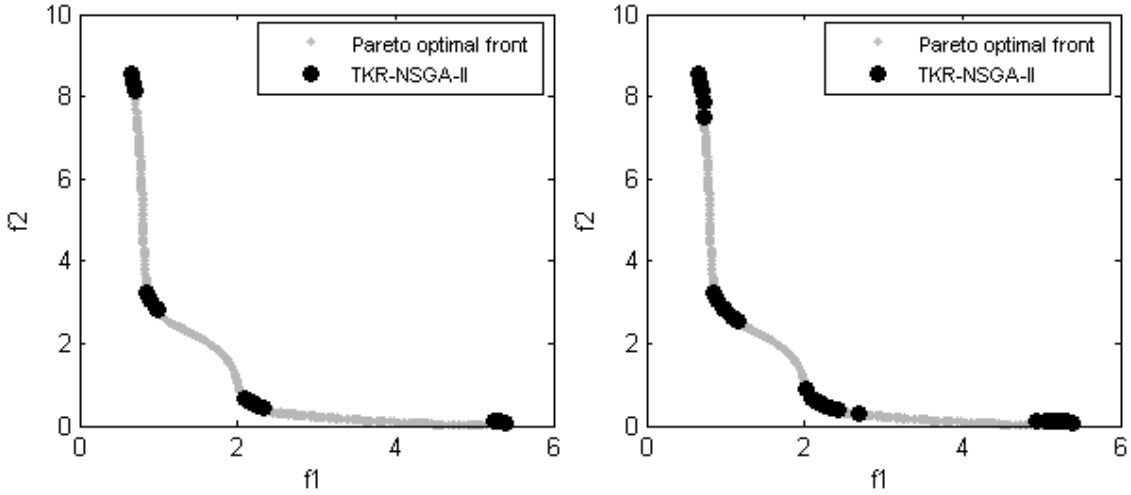
- **TKR-NSGA-II versus PLWSO**

The spread control in PLWSO is achieved by the parameter $\delta'$. Figure 6.15 shows the effect of varying the parameter $\delta'$ on the DO2DK problem with two and four knees respectively. We see that the higher the $\delta'$ value is, the larger the knee region spread is. However, PLWSO has the same inconvenience as WSNA. In fact, we observe that the spread increase does not apply uniformly for all knee regions which is not the case for TKR-NSGA-II (cf. figures 6.12 and 6.14). The uniformity of the extent of the obtained regions by TKR-NSGA-II may be explicated by the fact that the algorithm selects the mobile reference points based on the trade-off worth metric. When the reference points are stabilized on the knee centers, the population individuals are distributed near each knee center in such a way the minimal distance between two individuals in the objective space is greater than the user-specified quantity $\varepsilon$ and hence obtaining different ROIs having the same extent near the discovered knee centers. Differently, the




**(a)** **(b)**

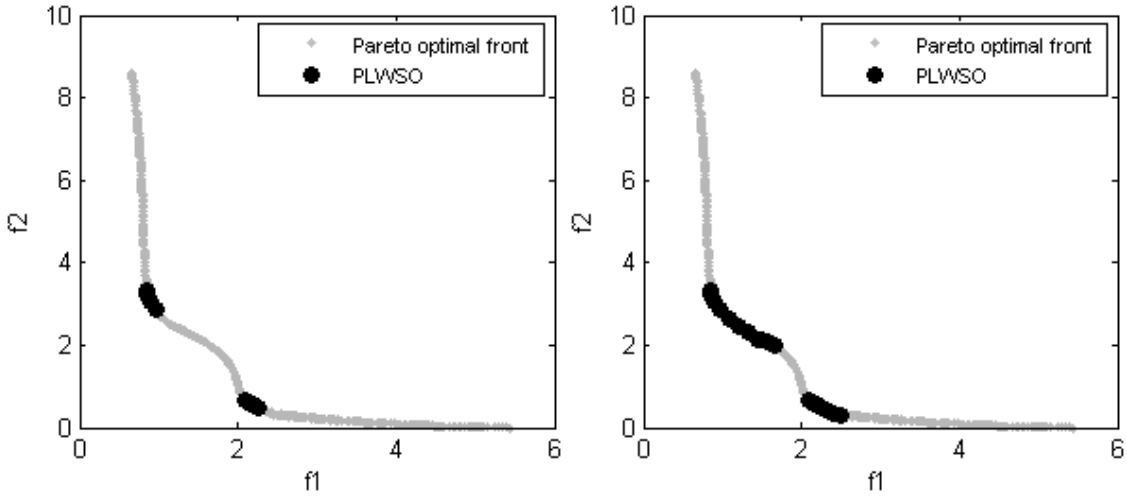
**Figure 6.13 WSNA spread control on DO2DK with $K$ = 2: (a) $Q$ = 100, $P$ = 30 and (b) $Q$ = 100, $P$ = 50.**




**(a)** **(b)**

**Figure 6.14 TKR-NSGA-II spread control on DO2DK with $K$ = 2: (a) $\varepsilon$ = 0.001 and (b) $\varepsilon$ = 0.01.**

PLWSO algorithm guides the search towards knee regions by optimizing a set of weighted sums instead of the original objectives. The spread control is achieved by modifying the weights corresponding to the potential knee solutions by means of the parameter $\delta'$ which yields knee regions with different spreads. We conclude that the spread control in PLWSO depends on the geometrical contour of the optimal knee region. That is why the algorithm provides regions with different spreads which is not the case for the TKR-NSGA-II. Additionally, Rachmawati and Srinivasan (2009) noted



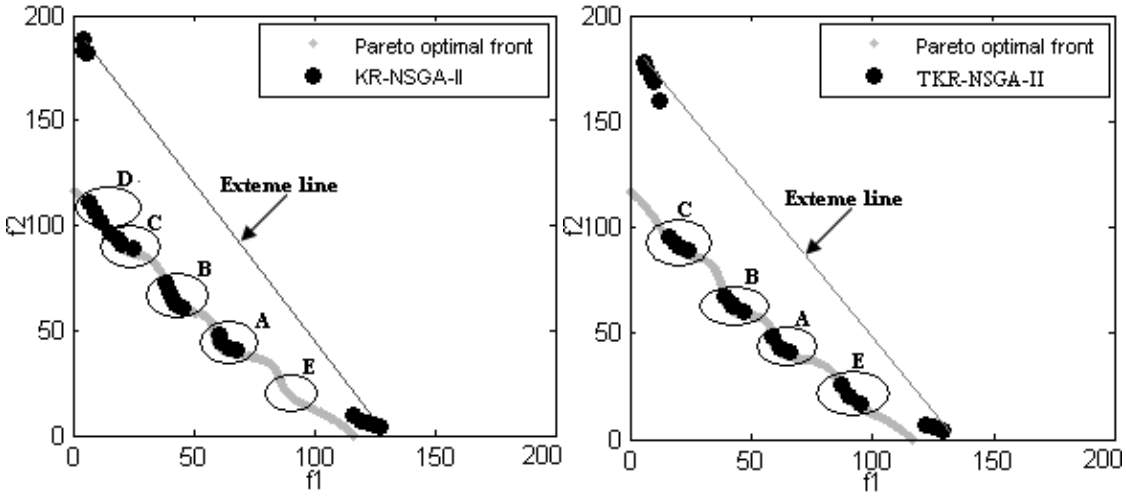

**(a)** **(b)**

**Figure 6.15 PLWSO on DO2DK with 2 knees: (a) $\delta' = 0.1$ and (b) $\delta' = 0.2$.**

that the setting of the $\delta'$ value is a critical task since a large value of this parameter makes the PLWSO algorithm missing potential knee regions while a smaller value of $\delta'$ leads to the identification of spurious ones. This fact makes the spread control ineffective in PLWSO. However, TKR-NSGA-II does not present such problem. The DM can use any positive real number for the $\varepsilon$ parameter without causing the difficulties faced by PLWSO which makes the spread control of knee regions more effective with TKR-NSGA-II than with the use of PLWSO.

- **TKR-NSGA-II versus KR-NSGA-II**

These two search methods use the same spread control mechanism. Hence, we obtain the same spread by the two algorithms. However, we would like to discuss here how the dependence of the KR-NSGA-II on the success of discovering the Pareto optimal extreme solutions may furnish bad results and mislead the DM. We perform an experiment on DEB2DK-1 test problem with 4 knees which is characterized by the non-uniformity of the solution distribution in the objective space. In fact, the density of solutions decreases when getting closer to the optimal front which makes the discovery of the true extreme solutions not easy. Figure 6.16 shows the obtained results for both algorithms after 15000 functions evaluations with $\varepsilon$ set to 0.01 and $\xi$ settled to 0.07. We see from figure 6.16(a) that KR-NSGA-II algorithm fails in discovering the optimal extreme solutions. Consequently, it provides three true knee regions (i.e., regions *A*, *B* and *C*) and a wrong knee region (i.e., region *D*). The discovery of the non-knee region *D* yields to two negative results. In one hand, the true knee *E* is lost. The algorithm is




**(a)** **(b)**

**Figure 6.16 Obtained results on DEB2DK-1 with $K$ = 4: (a) KR-NSGA-II and (b) TKR-NSGA-II.**

unable to encourage the survival of solutions in region *E* since solutions in the non-knee region *D* are farther from the extreme line. On the other hand, the true knee region *C* and the wrong knee-region *D* are adjacent in such a way they seem to form a big region with a large spread with respect to regions *A* and *B* which may mislead the DM in practical context. Contrariwise, although TKR-NSGA-II does not find the optimal extreme solutions, we see from figure 6.16(b) that it provides the four knee regions with approximately the same spread. This observation is explained by the fact that the mobile reference point set update is achieved in TKR-NSGA-II based on the trade-off worth metric $\mu$ which does not require the identification of the true extreme solutions. Extreme solutions are promoted in TKR-NSGA-II only to encourage population diversity. However, the selection of the reference points in KR-NSGA-II is based on the computation of the extreme line defined by the extreme solutions which explains the obtained results on the DEB2DK-1 test problem.

### 6.2.3 Discussion

Through section 6.2, we have addressed an important issue in multi-objective optimization consisting in finding knee regions of the Pareto front and hence providing the DM with information about the maximal trade-offs across the Pareto optimal front. This information is very important in practical context as the DM may be interested in gaining one unit in a chosen criterion at the expense of degradation in one or more other criteria. We have proposed preference-based MOEA, i.e., the KR-NSGA-II, which uses the concept of mobile reference point to guide the search towards knee regions. These mobile reference points play the role of attractors to the population

individuals, thereby directing the search towards each knee center. The knee region spread can be controlled by means of the user-specified $\varepsilon$-parameter. KR-NSGA-II has demonstrated its effectiveness and efficiency in approximating knee regions with a controllable spread. Moreover, the KR-NSGA-II interactive version was valorized via an interactive run. However, the dependence of KR-NSGA-II on the discovery of Pareto optimal extreme solutions has been shown to be the major inconvenient of this algorithm. For this reason, we have proposed an improved version of KR-NSGA-II, i.e., TKR-NSGA-II. The latter algorithm has demonstrated its superiority to the most representative works in this research field including KR-NSGA-II via a set of comparative experiments. Moreover, its independence of the Pareto front geometry and of the discovery of the true extreme solutions represents the main advantage of this enhanced version.

## 6.3 Proposed method for nadir point estimation

### 6.3.1 Algorithmic description

- **Basic idea**

In this subsection, we propose a new way to estimate the nadir point. Our aim is to ensure the accurateness of the estimation while minimizing the required computational effort. In the previous subsection, the evolutionary approaches were categorized into three classes. We suggest here a new further class called *extreme-region-of-interest-to-nadir* class. In fact, in the extreme-point-to-nadir class, the crowding distance is modified in order to concentrate the search in the vicinity of the extreme points. This *uncontrolled* focus on the extreme solutions may cause a lack of solution diversity and hence slows down the search process. The main idea in our new proposed approach (Bechikh et al. 2010b) is to guide the search towards extreme solutions while preserving a user-controlled diversity by constituting a ROI in the vicinity of each extreme non-dominated solution. The MR-NSGA-II$_N$ described in this subsection represents *an instance* of the new suggested class. In fact, we modify the preference-based MOEA R-NSGA-II of Deb et al. (2006a) in order to force the search to focus only on extreme solutions which allows the user to have an accurate estimation of the nadir objective vector quickly and reliably. Inspired from (Deb et al. 2006b), we propose to use extreme solutions (i.e., solutions having worst objective values) as reference points. Hence, the DM does not provide the set of reference points. This latter is updated automatically, in every generation of the algorithm, by making the current extreme solutions picked from the *best non-dominated front* as the current reference points. This automatic update of the reference points provides a ROI in the proximity of

each extreme point which facilitates the task of the discovery of the *Pareto optimal extreme solutions* and hence ensures the task of nadir point estimation. We recall that R-NSGA-II allows the DM to control the diversity of the obtained ROIs by using the clearing parameter $\varepsilon$. This latter allows controlling the focus of the search towards the ROIs which is not the case for Deb's approaches (Deb et al. 2006b). The diversity control allows not only escaping from local optima but also speeding up the convergence towards the reference points (i.e., the extreme solutions). The originality of our new proposed approach regarding what already exists in the literature is as follows: Instead of focusing the search strongly towards the extreme points without any control which may degrade the population diversity, this focus is controlled by a user-defined clearing parameter allowing the enhancement of solution diversification and hence speeding up the search by avoiding the stagnation in local optima.

- **Enhancement by local search**

Several works in the EMO field have shown that the incorporation of local search within MOEAs enhances the performance of these metaheuristics especially from a computational cost viewpoint (Bosman and de Jong 2005; Bechikh et al. 2008; Kumar et al. 2007; Shukla 2007). Such hybrid algorithms are termed *memetic algorithms* (Moscato 1989). Motivated by this observation, we propose to enhance our approach described in the previous subsection by hybridizing it with a gradient-based local. The aim of the local search is to push the mobile reference points further towards the true extreme Pareto optimal solutions and hence accelerating the convergence of the algorithm. Unlike the two-step local search designed by Deb et al. (2009a, 2009b), our new proposed local search is a one-step local search based on the SQP method (Wilson 1963). For this reason, we term the new designed local search *SQP-LS*. Our choice is justified by the fact that the SQP procedure has been shown to significantly enhance the overall performance when incorporated into several MOEAs (Hu et al. 2003; Kumar et al. 2007; Tiwari et al. 2009). In each generation of our memetic algorithm, the extreme solutions are picked from the best non-dominated front and then are subject to the SQP-LS. The SQP-LS algorithm is illustrated by figure 6.17. The local search procedure takes as inputs: (1) the *current reference point x* (i.e., the current extreme solution) and (2) *the critical objective index (coi)*, i.e., index of the objective in which the current reference point presents the worst value. As a result, it returns the *enhanced reference point r*. First, SQP-LS generates, from *x*, *M* neighbors (where *M* is the number of objectives) by using the SQP procedure. A neighbor *N(i)* is created by minimizing the $i^{th}$ objective individually (line 08). Then, *N(i)* is evaluated according to each objective function (line 09). It is important to note that by minimizing one of the objectives, we do not obtain dominated neighbors with respect to the reference point. Hence, the

**SQP-LS algorithm**

```
01. Input
02.   x: the reference point
03.   coi: the critical objective index
04. Output
05.   r: the enhanced reference point
06. Begin
07.   For i = 1 to M do
08.     N(i) ← SQP(x, i);
09.     N(i) ← evaluate_objectives(N(i));
10.   End For
11.   For i = 1 to M do
12.     If (N(i) Pareto dominates x) Then
13.       x← N(i);
14.     else If (N(i) and x are non-dominated) then
15.             If (N(i).f_coi > x.f_coi) then
16.               x← N(i);
17.             End If
18.           End If
19.     End If
20.   End For
21.   r ← x;
22. End
```

**Figure 6.17 The SQP-LS algorithm.**

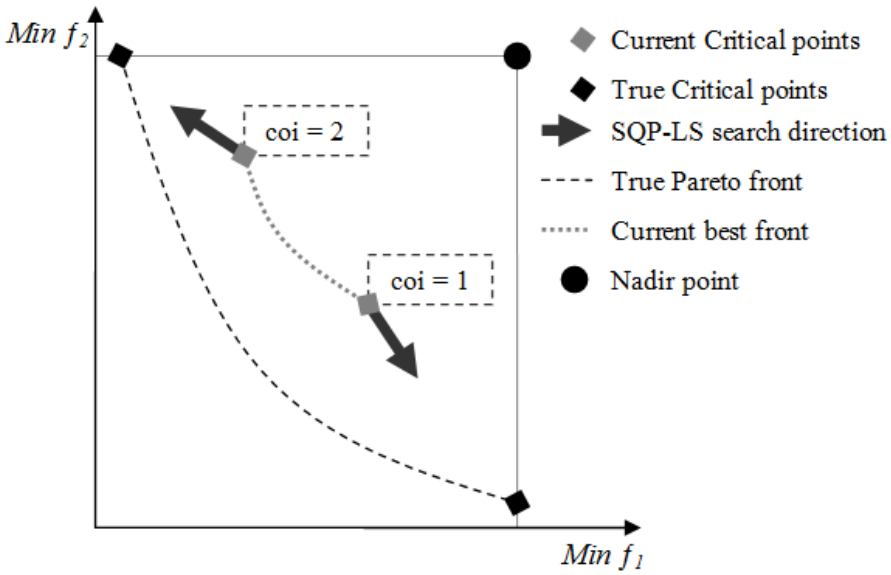



**Figure 6.18 The SQP-LS replacement strategy: the second case.**

neighborhood generation strategy preserves the order induced by the Pareto dominance relation. Once the neighborhood is generated (lines 07-10), the replacement process (lines 11-20) begins. The designed replacement strategy is as follows. The neighbor replaces the reference point in the two following cases: (1) the generated neighbor *N(i)*

Pareto-dominates the current reference point *x* (line 12) or (2) the neighbor *N(i)* and the reference solution *x* are non-dominated and *N(i)* have a worse value in the critical objective than *x* (lines 14-15). The first case emphasizes the convergence towards the optimal Pareto front. However, the second case encourages the convergence towards extreme Pareto optimal solutions from which the true nadir point is constructed as shown in figure 6.18.

Inspired from the work of Shukla (2007), we employ the Simultaneous Perturbation Method (SPM) instead of the Finite Difference Method (FDM) for gradient estimation. In fact, the one-sided FDM requires *n+1* FEs (where *n* denotes the number of decision variables) to compute the gradient. Assuming $g_i$ to be the $i^{th}$ component of the gradient, $e_i$ to be a unit vector in the $i^{th}$ direction and *c* to be the step size at each generation, then for an *n*-dimensional variable *x*, the one-sided FDM used in (Hu et al. 2003; Kumar et al. 2007; Sharma et al. 2007; Tiwari et al. 2009) requires *n+1* FEs to compute the gradient and is given by:

$$g_i(x) = \frac{f(x + ce_i) - f(x)}{c} \tag{6.5}$$

This is costly in terms of FEs (of the order O(*n*)). However, the one-sided SPM requires only two FEs to estimate the gradient independently of the number of decision variables *n* and is given by:

$$g_i(x) = \frac{f(x + c\Delta) - f(x)}{c\Delta_i} \tag{6.6}$$

where $\Delta$ is an *n*-dimensional vector of random perturbations satisfying some statistical conditions (Spall 1998) and $\Delta_i$ is its $i^{th}$ component. The computational complexity of the SPM is thus O(1) which justifies our choice.

Our gradient-based memetic MOEA proposed in this subsection is termed *Memetic R-NSGA-II for Nadir point estimation* and is denoted *MR-NSGA-II*$_N$. Figure 6.19 illustrates the basic iteration of this hybrid algorithm. We see from this figure that our new algorithm is a modified version of R-NSGA-II. The main modifications are: (1) the updating strategy of the mobile reference points set *E* (lines 11-12) and (2) the enhancement of the extreme solutions (i.e., the reference points) by the SQP-LS procedure (lines 13-15). We note that we have incorporated Deb's constraint-handling strategy (Deb 2000) in MR-NSGA-II$_N$ in order to make the algorithm able to handle constrained problems. This strategy is described as follows:

1) Any feasible solution is preferred to any infeasible solution,
2) Among two feasible solutions, the dominating one is preferred,

**MR-NSGA-II$_N$ basic iteration algorithm**

**Input**
  $P_t$ : the parent population at generation t
  $Q_t$ : the offspring population at generation t
  M : the number of objective functions
**Output**
  $P_t$ : the updated parent population at generation t
  $Q_t$ : the updated offspring population at generation t
**Begin**
  $R_t \leftarrow P_t \cup Q_t$ ;
  $R_t \leftarrow$ non-domination_sort ($R_t$, M) ;
  $F_{best} \leftarrow$ select_best_front ($R_t$) ;
  E $\leftarrow$ select_extreme_solutions ($F_{best}$, M) ;
  **For** i = 1 **to** M **do**
    E(i) $\leftarrow$ SQP-LS (E(i), i) ;
  **End For**
  $R_t \leftarrow$ distance_from_extreme_solutions ($R_t$, E) ;
  $R_t \leftarrow$ extreme_points_based_crowding ($R_t$, E) ;
  $R_t \leftarrow$ ε-clearing ($R_t$) ;
  t $\leftarrow$ t+1 ;
  $P_t \leftarrow$ environmental_selection ($R_{t-1}$) ;
  $Q_t \leftarrow$ reproduction ($P_t$) ;
**End**

**Figure 6.19 The MR-NSGA-II$_N$ basic iteration.**

3) Among two unfeasible solutions, the one having smaller overall constraint violation is preferred.

## 6.3.2 Experimental results

### ♦ Unconstrained problems

In this subsection, we assess the performance of our algorithm on three- to twenty-objective unconstrained non-linear test problems. Thus, we compare MR-NSGA-II$_N$ to: (1) the naïve NSGA-II approach, (2) the WC-NSGA-II and (3) the EC-NSGA-II. We adopt the same experimental design used in (Deb et al. 2006b) for fairness of comparison. For the SQP-LS, a neighbor is generated by means of a modified version of the `fminunc` MATLAB function by changing the gradient computing strategy as described previously. The termination criterion of the SQP function (i.e., `fminunc`) is: (1) the norm of decent direction $\|d\| \leq 10^{-8}$ or (2) the number of allowed iterations $\mu$ is elapsed ($\mu$*=50* for *M=3* and $\mu$*=20* for *M>3*). The used test problems are DTLZ1, DTLZ2, and the modified DTLZ5 (Deb et al. 2006b). The corresponding true nadir points are known and are: $(0.5,0.5,\ldots,0.5)^T$ for DTLZ1 test problem, $(0,0,\ldots,0)^T$ for DTLZ2 test problem and $\left(\left(1/\sqrt{2}\right)^{M-2},\left(1/\sqrt{2}\right)^{M-2},\left(1/\sqrt{2}\right)^{M-3},\left(1/\sqrt{2}\right)^{M-4},\ldots,\left(1/\sqrt{2}\right)^{0}\right)^T$ for

DTLZ5 test problem. Also, we use the same termination criterion described in (Deb et al. 2006b) which expresses the proximity to the true nadir point. In fact, every *p_size* (*p_size*: the population size) FEs, we compute the following quantity:

$$D = \sqrt{\sum_{i=1}^{M} \left( \frac{z_i^{I} - z_i^{est}}{z_i^{I} - z_i^{N}} \right)^2} \tag{6.7}$$

where $z_i^{I}$ is the $i^{th}$ component of the ideal objective vector, $z_i^{N}$ is the $i^{th}$ component of the true nadir point and $z_i^{est}$ is the $i^{th}$ component of the estimated nadir point. When a value smaller than a threshold $\eta$ ($\eta$ = 0.01 is used here) is found, the simulation is terminated and the algorithm is said to be successful in finding a good estimate of the true nadir point. We note that the population size is 100 for M∈ {3, 5}, 200 for M = 10, and 500 for M∈ {15, 20}. The clearing factor $\varepsilon$ is set to 0.001 for M∈ {3, 5}, and 0.0005 for M∈ {10, 15, 20}.

Figures 6.20-6.22 illustrate the median values (of 11 runs) of the required number of FEs to find a near nadir point (within $\eta$ = 0.01) by the four algorithms for DTLZ1, DTLZ2 and DTLZ5 respectively. From figure 6.22, we see that all the algorithms perform more or less similar to each other for 3-, 5- and 10-objective DTLZ5. For 15- and 20-objective DTLZ5, MR-NSGA-II$_N$ presents slightly better results than the three other algorithms. This similarity of results is explicated by the one-dimensional nature of the Pareto optimal front of DTLZ5 which makes the discovery of the true nadir point an easy task to achieve for the four search methods. From figures 6.20-6.21, we observe that WC-NSGA-II, EC-NSGA-II and MR-NSGA-II$_N$ are strictly better than the naïve NSGA-II approach on DTLZ1 and DTLZ 2 especially when M≥5. For 10-objective DTLZ2, the naïve NSGA-II achieves a normalized difference measure $D$ = 6.002 after 10 million FEs which signifies that this approach is unable to find a near nadir point. For this reason, we do not show the median value of this method in figure 6.21 for M≥10. Additionally, in figure 6.20, we do not show the median of the needed number of FEs for the naïve approach on DTLZ1 for M≥15 since this value is very high on the 10-objective DTLZ1 compared to the three other algorithms. Let we now compare the MR-NSGA-II$_N$ algorithm to the two modified versions of NSGA-II on DTLZ1 and DTLZ2. In fact, our hybrid algorithm performs: (1) slightly better than WC-NSGA-II for M∈ {3, 5} and (2) strictly better than WC-NSGA-II for M∈ {10, 15, 20}. Besides, MR-NSGA-II$_N$ provides: (1) slightly better results than EC-NSGA-II for M∈ {3, 5, 10} and (2) strictly better results for M∈ {15, 20}. The performance of MR-NSGA-II$_N$ may be explicated by two reasons. The first reason is that our hybrid algorithm does not

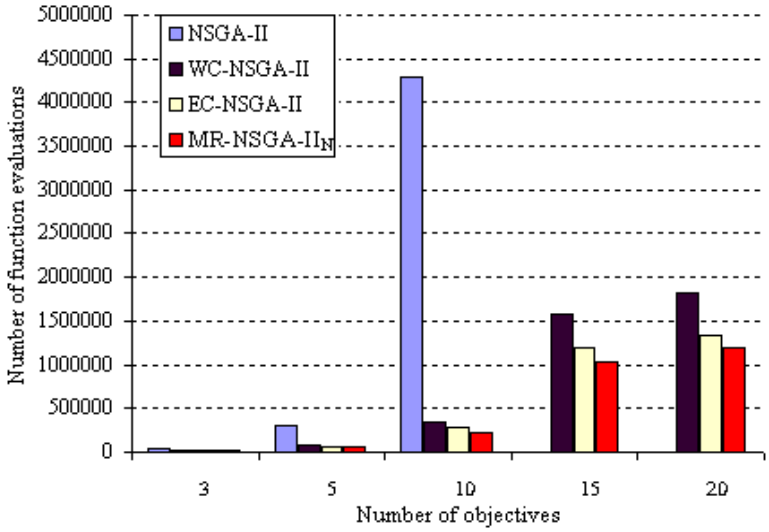



**Figure 6.20 Required number of FEs for DTLZ1.**

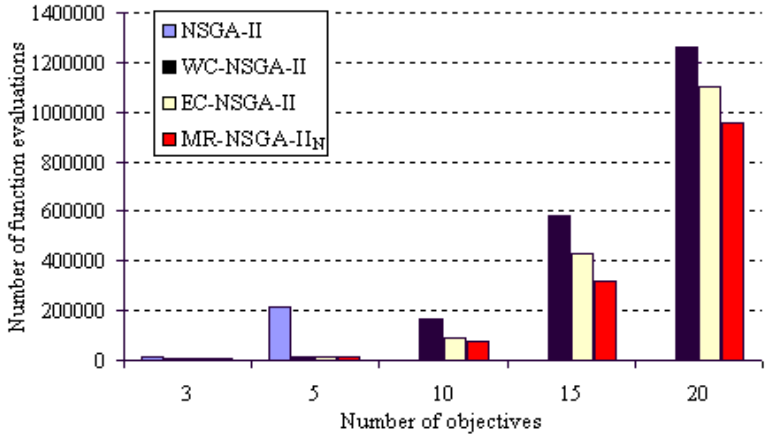



**Figure 6.21 Required number of FEs for DTLZ2.**

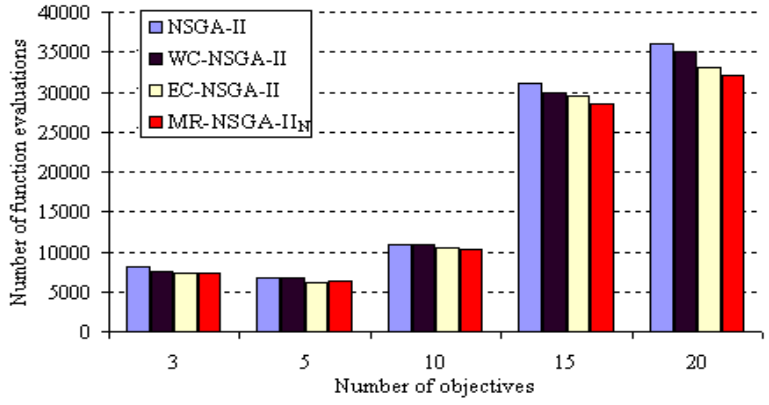



**Figure 6.22 Required number of FEs for DTLZ5.**

concentrate the search *strongly* in the vicinity of the extreme solutions as done in the EC-NSGA-II. However, our algorithm constitutes a neighborhood (i.e., an ROI) in the proximity of each extreme solution in such a way the diversity of each neighborhood is controlled by the $\varepsilon$-clearing parameter. This diversity control not only avoids the slowing down of the search and the stagnation of the population in a certain region of the search space but also it prevents spurious non-dominated non Pareto optimal solutions to remain in the population as observed for WC-NSGA-II (Deb et al. 2009a). Consequently, this diversity control accelerates the search towards the Pareto optimal extreme solutions. The second reason is the use of the gradient-based local search as an additional operator of the evolutionary algorithm which emphasizes more the convergence towards the optimal extreme solutions. We conclude that MR-NSGA-$II_N$ maintains a good balance between convergence and diversity which explicates the obtained results on DTLZ1 and DTLZ2.

♦ **Constrained problems**

In this subsection, we assess the ability of our memetic algorithm to solve constrained non-linear problems. We consider the tri-objective KM problem (Klamroth and Miettinen 2008) which is defined as follows:

$$Min \begin{cases} -x_1 - x_2 + 5 \\ \frac{1}{5}\left({x_1}^2 - 10x_1 + {x_2}^2 - 4x_2 + 11\right) \\ (5 - x_1)(x_2 - 11) \end{cases} \tag{6.8}$$

Subject to:

$$\begin{cases} 3x_1 + x_2 + -12 \le 0 \\ 2x_1 + x_2 - 9 \le 0 \\ x_1 + 2x_2 - 12 \le 0 \\ 0 \le x_1 \le 4 \\ 0 \le x_2 \le 6 \end{cases} \tag{6.9}$$

It was reported in (Klamroth and Miettinen 2008) that this problem makes difficulty for the payoff table. Individual minimization of objectives yields the following three objective vectors: $(-2,0,-18)^T$, $(0,-3.1,-14.25)^T$, and $(5,2.2,-55)^T$ thereby the payoff table provides $(5,2.2,-14.25)^T$ as an estimate of the nadir point. However, it was reported in (Deb et al. 2009a; 2009b) that the true nadir point of this problem is $(5,4.6,-14.25)^T$. Consequently, the nadir point estimation supplied by the payoff table

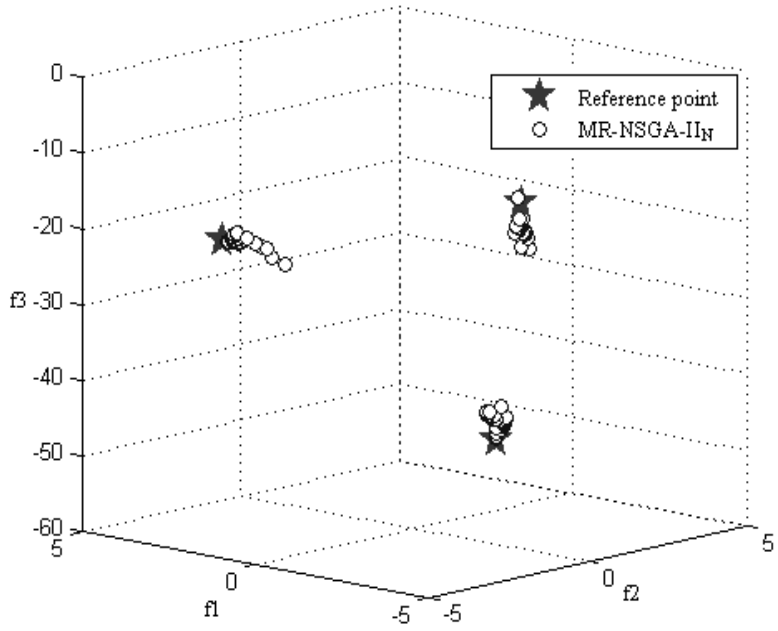



**Figure 6.23 The obtained solutions for the constrained KM problem.**

is wrong. In order to assess the ability of our algorithm in the constrained case, we apply MR-NSGA-II$_N$ to the KM problem. For the SQP-LS, a neighbor is generated by means of a modified version of the `fmincon` MATLAB as described previously. The population size is settled to 40 and the other parameters are set as previously. Figure 6.23 shows the mean of the obtained solutions of 10 runs. The final reference points are drawn with stars. We observe that our algorithm provides a ROI in the vicinity of each reference point. It is important to note that the mobile reference points (i.e., $(5.000,2.209,-55.003)^T$, $(-1.000,4.604,-25.003)^T$ and $(0.000,-3.101,-14.252)^T$) have converged to the true extreme Pareto optimal solutions (i.e., $(5.000,2.200,-55.000)^T$, $(-1.000,4.600,-25.000)^T$ and $(0.000,-3.100,-14.250)^T$) within $\eta$ = 0.01 at the end of the evolutionary process. Consequently, MR-NSGA-II$_N$ is declared to be successful in finding the true nadir point of the KM problem (i.e., $(5,4.6,-14.25)^T$). We note that each run takes a reasonable number of FEs (about 4000 FEs) to find a near nadir point. It is worth noting that figure 6.23 illustrates the working principle of MR-NSGA-II$_N$. In fact, we see from this figure that an ROI is formed near each reference point. Then, the reference points are updated by: (1) selecting the extreme solutions from each ROI and (2) applying the SQP-LS procedure to these solutions. When the reference points are stabilized at the true extreme solutions, the algorithm stops and provides a good estimate of the nadir point.

### 6.3.3 Discussion

Through section 6.3, we have addressed an old-age yet important issue in multi-objective optimization, i.e., that is the estimation of the nadir objective values. As discussed previously, the nadir point could be seen as a form of implicit DM's preferences. After discussing the classification of the existing nadir point estimation methods into three classes, we have proposed a new class called the extreme-ROI-to-nadir class. This proposition is motivated by the fact that, in the extreme-point-to-nadir approach, the progressive reduction of population diversity may slow down the search process. Hence, the constitution of a ROI in the vicinity of each extreme point from the population where the ROI diversity is controlled by the $\varepsilon$ parameter seems to be a promising approach. The MR-NSGA-II$_N$, which is an instance of the ROI-to-nadir class, has demonstrated its effectiveness and efficiency in providing a good nadir point estimation. Moreover, this algorithm has been shown to be superior to WC-NSGA-II and EC-NSGA-II which belong to the extreme-point-to-nadir class. It is important to note that other algorithms could be designed based on the extreme-ROI-to-nadir class principle.

## 6.4 Conclusion

In this chapter, we have contributed to the search for special points of the Pareto front that correspond to implicit DM's preferences. This is achieved through the approximation of knee regions and nadir objective values. The proposed algorithms used the new concept of mobile reference point (Bechikh et al. 2010a). The mobile reference points play the role of attractors to the population members and hence they guide the search towards special parts of the Pareto front that are implicitly preferred by the DM. This type of reference points is updated automatically by the MOEA with respect to a predefined characterization (i.e., maximal trade-off or extreme point). Similarly to the r-NSGA-II, the diversity control mechanism is very important in the three algorithms: (1) KR-NSGA-II, (2) TKR-NSGA-II and (3) MR-NSGA-II$_N$ since focusing the search towards a particular region of the search space reduces significantly the solution diversification which could make the search decelerating or even stagnating. The diversity preservation in the three proposed algorithms was ensured by means of the $\varepsilon$-clearing procedure in addition to the extreme solution preservation. Such mechanism enabled these algorithms to converge towards the Pareto optimal knee regions and the true extreme solutions. The three suggested algorithms have been confronted to recent existing works from the specialized literature and promising results were obtained.

# Chapter 7

# Incorporating DM Group Preferences in Evolutionary Multi-objective Optimization

## 7.1 Introduction

As noted in the second chapter, most preference-based MOEAs assume the uniqueness of the DM. Few of these algorithms consider the hypothesis that there exist more than one DM by injecting several reference points in the MOEA each corresponding to a particular DM than the algorithm provides an average ROI (e.g., (Pfeiffer et al. 2008)). However, this mechanism does not resolve the problem since most DMs are still dissatisfied. In fact, the task of DMs' preference aggregation cannot be delegated to a MOEA. This latter cannot achieve a consensus between the different negotiators. Motivated by this observation and inspired from the works (Ben Jaâfar and Ghédira 2007; Conitzer 2006; Herrera-Viedma et al. 2007), we propose in this chapter a negotiation support system called NSS-GPA (Negotiation Support System for Group Preference Aggregation) (Bechikh et al. 2011b). NSS-GPA takes as inputs the DMs' preferences modelled as reference points and provides as output a single Social Reference Point (SRP) that corresponds to an aggregation of all DM's preferences. By running the preference-based MOEA with this SRP, we obtain a social ROI corresponding to aggregated DMs' preferences. Each solution picked from this region is considered as a satisfying solution for each of the DMs. We aim by NSS-GPA to ensure the highest level of satisfaction for all DMs. Since, in real world situations, the DMs' preferences are usually conflicting, NSS-GPA offers the DMs a framework of negotiation to confront and update their preferences through a number of negotiation rounds. This chapter is structured as follows. The next section describes in detail the NSS-GPA mechanism. The third section demonstrates the usefulness of our system through a randomly generated case study. The fourth section valorizes the combination between NSS-GPA and r-NSGA-II (cf. section 5.2.4) through the resolution of a practical instance of the portfolio selection problem with multiple DMs (Bechikh et al. 2012b). The last section concludes this chapter.

## 7.2 NSS-GPA

### 7.2.1 Overview

Since our goal consists in providing a framework for different DMs to negotiate their preferences, we choose to exploit the software agent paradigm (Morge and Beaune 2004). Hence, we propose an agent-based system for group preference negotiation (i.e., NSS-GPA) to bring closer DMs' reference points through a certain number of negotiation rounds. In fact, each DM is assisted by a software agent called Assistant agent. The overall process is supervised and controlled by a software Moderator agent.

Initially, each human DM agent $E_j$ $(j = 1,...,q)$ expresses his/her preferences as an aspiration level vector $AV_j = [a_{j1},...,a_{jM}]$ where $M$ is the number of objectives and $q$ is the number of DMs. Additionally, he/she provides for each objective $f_i$ $(i = 1,...,M)$ the acceptable deviation quantity $\sigma_{ji}$ from his/her specified aspiration level $a_{ji}$ so that the agent is still satisfied with these specified deviations. Once all DM agents have expressed their preferences, the negotiation process begins. At each negotiation round of NSS-GPA, a DM agent may have one of the following two states: (1) *satisfied* or (2) *dissatisfied*. These two states are defined as follows:

**Definition 7.1: Satisfied DM agent**

A DM agent $E_j$ $(j = 1,...,q)$ is said to be satisfied if all its aspiration levels $a_{ji}$ $(i = 1,...,M)$ *σ-coincide* with the social aspiration levels. An aspiration level $a_{ji}$ is said to σ-coincide with the social aspiration level $a_i^{avg}$ if and only if: $a_i^{avg} - \sigma_{ji} \le a_{ji} \le a_i^{avg} + \sigma_{ji}$. We note that $a_i^{avg} = \left(\sum_{j=1}^{q} a_{ji}\right) \Big/ q$ and $SRP = [a_1^{avg},...,a_M^{avg}]$.

**Definition 7.2: Dissatisfied DM agent**

A DM agent $E_j$ $(j = 1,...,q)$ is said to be dissatisfied if at least one of his/her aspiration levels does not σ-coincide with the relative social aspiration level.

After each negotiation round, the Moderator agent builds a set of direction rules from the observed DM agents' preferences (cf. section 7.2.2.1). These direction rules guide the DMs when updating their preferences so that the consensus rate increases and hence the negotiation process converges towards a SRP more quickly. A satisfied DM agent can follow the direction rules in order to stop the negotiation processes as soon as

possible so that he/she finishes the negotiation with a satisfied state. However, a dissatisfied DM agent can have two attitudes: (1) *passive* who will pursue the Moderator's direction rules or (2) *active* who has one of the two following behaviors:

- *manipulator*: this kind of DMs will lie about his/her true preferences in order to direct the SRP towards his/her preferences. For example, an agent $E_j$ aspiration level $a_{ji}$ is set to $0.4$ (with $\sigma_{ji} = 0.1$). However, the corresponding current social aspiration level $a_i^{avg}$ is found to be equal to 0.7. $E_j$ will lie in the next negotiation round by putting $a_{ji}$ to $0.1$ in order to try to decrease $a_i^{avg}$ towards his/her true preferences about the $i^{th}$ objective (i.e., $0.4$).
- *non-manipulator*: this kind of DMs will search for the dissatisfied DMs and will invite them to update their preferences with the aim of modifying the SRP towards his/her preferences. For example, for two dissatisfied DMs $E_1$ and $E_2$, we suppose that, for the objective $f_1$, we have: $a_{11} = 0.5$, $a_{21} = 0.3$ and $a_1^{avg} = 0.8$. $E_1$ will send a request to $E_2$ in order to invite him/her to decrease $a_{21}$ (eventually, while respecting his/her specified acceptable deviation $\sigma_{21}$). In fact, by decreasing $a_{11}$ and $a_{21}$, $a_1^{avg}$ value will decrease and will become closer to $a_{11}$ and $a_{21}$ which decreases the dissatisfaction level for both DMs.

Manipulation in voting systems is seen to be a dishonest behavior which should be avoided (Conitzer 2006, Xia and Conitzer 2008). For this reason, the Moderator agent which has a global overview about the overall negotiation system, may perceive that there are some manipulations during the negotiation rounds and hence detects the manipulator agents. In fact, in NSS-GPA, a manipulation is seen as an abrupt change in the DM's preferences which aims to modify one or more social aspiration levels in order to increase the DM's individual welfare. Manipulation seems to be a selfish and dictatorial behavior because if all agents are manipulators, the consensus will never be reached and hence the negotiation process will never end. For this reason, the Moderator agent has the role to detect manipulators and to punish them by retrieving them the right to update their preferences without following the global direction rules. In fact, manipulators will be punished by forcing them to pursue the global direction rules issued from the Moderator software agent.

## 7.2.2 Conceptual details

### 7.2.2.1 Production of direction rules and process control

♦ **Determination of the set of preferences to be updated**

For each aspiration level $a_{ji}$ of a DM agent $E_j$, the Moderator agent computes the mean gap separating $a_{ji}$ from the aspiration levels $a_{ki}$ of the other agents as follows:

$$MG\_aspiration(a_{ji}) = \sum_{k=1,k\neq j}^{q} \left| a_{ji} - a_{ki} \right| \Big/ (q-1) \tag{7.1}$$

After that, the Moderator agent calculates the average mean gap for each aspiration component $a_i$ as follows:

$$Average\_MG(a_i) = \sum_{j=1}^{q} MG\_aspiration(a_{ji}) \Big/ q \tag{7.2}$$

The Moderator agent can now determine the preferences to be updated in order to increase the consensus level. In fact, the Moderator agent aims to minimize the mean gap of each aspiration level by using the following rule (**R1**):

**If** ( $MG\_aspiration(a_{ji}) > Average\_MG(a_i)$ ) **Then**

**Update** ( $a_{ji}$ );

**End If**

♦ **Determination of the set of DM agents invited to update their preferences**

At the beginning of the negotiation process, all DM agents can be invited to update their preferences. In fact, if an aspiration level evaluation $a_{ji}$ is to be changed, then automatically the DM agent $E_j$ is invited to modify his/her preferences. This mechanism allows evading the problem of group tyranny (Saint and Lawson 1994). After some negotiation rounds, the consensus level will increase. In order to preserve this increase and encourage the consensus improvement, we minimize the number of aspiration level evaluations to be updated. This is achieved by minimizing the number of DM agents invited to modify their preferences. These agents are identified as follows:

For each DM agent $E_j$, the Moderator agent computes the mean gap separating the agent's preferences from all other agents' preferences as stated by equation (7.3):

$$MG\_agent(E_j) = \sum_{k=1,k\neq j}^{q} Gap(E_j, E_k) \Big/ (q-1) \tag{7.3}$$

where

$$Gap(E_j, E_k) = \sum_{i=1}^{M} \left| a_{ji} - a_{ki} \right| \Big/ M \tag{7.4}$$

Then, the Moderator calculates the average agent mean gap:

$$Average\_AMG = \sum_{j=1}^{q} MG\_agent(E_j) \Big/ q \tag{7.5}$$

Now, the Moderator agent can identify the agents that should update their reference points in order to augment the consensus level. This is achieved by minimizing the mean gap of each agent by the following rule (**R2**):

**If** ( $MG\_agent(E_j) > Average\_AMG$ ) **Then**
  **Invite_for_update** ( $E_j$ );
**End If**

- **Process control**

Here, the Moderator agent firstly computes the SRP which corresponds to the arithmetic mean of all DM agents' reference points, i.e., $SRP = [a_1^{avg},..., a_M^{avg}]^T$ such that:

$$a_i^{avg} = \left( \sum_{j=1}^{q} a_{ji} \right) \Big/ q \qquad \forall i = 1,..., M \tag{7.6}$$

After that, the Moderator agent calculates the gap separating each DM agent's reference point from the SRP:

$$Gap\_from\_SRP(E_j) = \sum_{i=1}^{M} \left| a_{ji} - a_i^{avg} \right| \Big/ M \tag{7.7}$$

Then, the Moderator agent computes the average of all gaps separating the agents from the collective opinion:

$$Average\_Gap\_from\_SRP = \sum_{j=1}^{q} Gap\_from\_SRP(E_j) \Big/ q \tag{7.8}$$

The overall negotiation process is controlled by the Moderator agent based on: (1) the *Consensus Rate* (*CR*) and (2) the parameter *MaxIter*. The *CR* is expressed as follows:

$$CR = 1 - Average\_Gap\_from\_SRP \tag{7.9}$$

*MaxIter* corresponds to the maximum allowed number of negotiation rounds. This parameter is important since it ensures that the process ends after a specified number of negotiation rounds.

Based on *CR* and *MaxIter*, the Moderator agent controls the negotiation process by the following rule (**R3**):

**If** (*numIter* ≤ *MaxIter*) **Then**
  **If** ( $CR < \alpha$ ) **and** (**NOT** All_Satisfied) **Then**
    **Execute_update** (R1); /*All DM agents can be invited to the update operation*/
  **Else If** ( $CR < \beta$ ) **and** (**NOT** All_Satisfied) **Then**
    **Execute_update** (R1, R2); /*Some DM agents can be invited to the update operation*/
  **Else**
    **Stop_negotiation** ( );
  **End If**
**End If**
**Else**
  **Stop_negotiation** ( );
**End If**

where *numIter* is the current negotiation round index; $\alpha$ and $\beta$ are two control parameters which are specified before the beginning of the negotiation process such that $\alpha \in [0,1[$, $\beta \in [0,1]$ and ( $\alpha < \beta$ ). *All_Satisfied* is a Boolean variable indicating whether all DMs are satisfied (cf. Definition 7.1).

- **Direction rule production**

Once the preferences to be updated and the agents invited for the update operation are identified, the Moderator agent furnishes the advice rules to the DM agents as follows:

**Advice rule A1:** **If** ( $a_{ji} < a_i^{avg}$ ) **Then**
  **Invite the agent** $E_j$ **to increase** $a_{ji}$ ;
**End If**

**Advice rule A2:** **If** ( $a_{ji} > a_i^{avg}$ ) **Then**
  **Invite the agent** $E_j$ **to decrease** $a_{ji}$ ;
**End If**

The objective of these rules is to bring closer DMs' preferences in order to reach a high level of consensus.

#### 7.2.2.2 Manipulator isolation

Manipulation is a bad and undesirable behaviour in group decision making situations and especially in social choice theory (Xia and Conitzer 2008). For this reason, we offer the Moderator agent the ability to detect such behaviour and to penalize

manipulator DMs. In NSS-GPA, a manipulation is an abrupt change in one DM's aspiration level, which does not respect the relative accepted deviation of the last negotiation round, in such a way the preference update modifies the SRP in the direction of the manipulator preferences. We assume, in our system, that each DM has the right to make $M$ manipulations during the negotiation since, in real world negotiation situations, a negotiator may give up one of his/her aspiration levels without aiming to manipulate the negotiation. For example, if a DM agent $E_j$ is satisfied with all social aspiration levels except for one aspiration level $a_{ji}$ where the $i^{th}$ objective is not so important for him/her, then he/she prefers updating his/her preferences so that to increase the consensus rate which augments the chance of ending the negotiation process with an almost satisfied state. Thus, the Moderator agent considers a DM agent to be a manipulator if he/she performs $(M+1)$ manipulations. When, a DM agent is detected as a manipulator, the Moderator agent deprives him/her not only of manipulations but also of sending/receiving messages to/from other agents. Hence, the manipulators are isolated and are obliged to update their preferences according to the global direction rules; thereby increasing the consensus rate and making the negotiation process further converging. Manipulator isolation is an important mechanism to avoid selfishness, untrustworthiness and dictatorship behaviors.

#### 7.2.2.3 Dissatisfied non-manipulator DMs' communication

A dissatisfied non-manipulator DM agent would like to decrease his/her dissatisfaction degree by negotiating with other dissatisfied DMs. We note that each negotiator has a complete vision over the preferences of the others. As mentioned above, a DM is said to be dissatisfied if at least one of his/her aspiration levels does not σ-coincide with the corresponding social one (cf. Definition 7.2). For example, consider the case of 5 objectives and 10 DMs, and suppose that there are three DM agents $E_1$, $E_2$ and $E_3$ which are dissatisfied with the value of the third social aspiration level $a_3^{avg}$ such that: $a_3^{avg} = 0.6$, $a_{13} = 0.4$, $a_{23} = 0.8$ and $a_{33} = 0.2$. We suppose also that there is a satisfied agent $E_4$ having $a_{43} = 0.8$. It is interesting to agent $E_1$ to contact agents $E_2$, $E_3$, $E_4$ by sending them requests to decrease their aspiration level evaluations $a_{23}$, $a_{33}$ and $a_{43}$ while respecting their $\sigma$ values. In fact, if $E_2$, $E_3$, $E_4$ agree about that, there is more chance that $a_3^{avg}$ decreases towards the value of $0.4$ which decrease the dissatisfaction level of $E_1$. Thus, the aim of communicating with other agents is to bring closer the collective opinion towards $E_1$ preferences. Agent $E_3$ seems to be

interested with such proposal since driving $a_3^{avg}$ towards $a_{13} = 0.4$ is equivalent to driving $a_3^{avg}$ towards $a_{33} = 0.2$. However, in order to convince $E_3$ to accept his/her request, $E_1$ should promise $E_3$ that he/she will decrease his/her aspiration level evaluation $a_{13}$ if $E_3$ accepts the proposal and performs the decrease. However, in real world situation, a DM may be a liar. So, if $E_1$ lies to $E_3$ then $E_3$ marks $E_1$ as a liar and does not accept his/her future proposals. Besides, if $E_1$ takes his/her promise, $E_3$ marks $E_1$ as a trustworthy agent and may accept his/her future requests. On the contrary to agent $E_3$, agent $E_2$ is not interested to such request since decreasing $a_3^{avg}$ makes the SRP roll away from his/her preferences. Consequently, $E_2$ simply rejects the request. The satisfied agent $E_4$ may be interested in accepting $E_1$ proposal in order to improve the consensus level and hence augmenting the probability of ending the negotiation sooner with a satisfied state.

### 7.2.3 Implementation details

In this subsection, we give some important implementation details of our system. NSS-GPA is implemented by using the Java Agent DEvelopment framework (JADE) (Bellifemine et al. 2007) and the ECLIPSE programming tool (d'Anjou et al. 2004). Each Assistant agent has a set of cyclic behaviors allowing it to perceive its environment, to communicate with other agents and to update its preferences. The Moderator agent also has a set of cyclic behaviors allowing him to perceive the DM agents' preferences, to produce then broadcast the advice rules for them, to detect then punish manipulators and to control the negotiation process based on the control parameters $\alpha$, $\beta$ and *MaxIter*.

Assuming a minimization MOP, the system provides the DMs with the ideal objective vector in addition to the nadir one by using, for example, our MR-NSGA-II$_N$ method (Bechikh et al. 2010b). In this way, each DM's aspiration level value lies in the interval $[f_i^{ideal}, f_i^{nadir}]$. After that, the system uses the normalized aspiration values (which lie in the interval $[0,1]$) in order to ensure that all mean/average gap values lie between 0 and 1. In this way, the negotiation can be well-controlled and the recommendations can be fairly produced based on the designed rules (cf. section 7.2.2.1). Additionally, the system works with the normalized accepted deviation values which can be expressed as follows:

$$\sigma_{ji}^{norm} = \sigma_{ji} \Big/ (f_i^{nadir} - f_i^{ideal}) \tag{7.10}$$

We note that NSS-GPA imposes that $\sigma_{ji}^{norm} \in [0, \sigma_{\max}^{norm}]$ in order to control manipulations where $\sigma_{\max}^{norm}$ is specified before the beginning of the negotiation by a human Moderator agent. At this stage, we can define a DM's manipulation analytically. Assuming $a_{ji}^{t-1}$ an aspiration level fixed by DM agent $E_j$ for the $i^{th}$ objective at the previous negotiation round $(t-1)$ and $a_{ji}^{t}$ is the updated value of $a_{ji}^{t-1}$ at the actual negotiation round $(t)$. The update operation is said to be a manipulation if and only if:

$$\left| a_{ji}^{t} - a_{ji}^{t-1} \right| > \sigma_{ji}^{t-1} \tag{7.11}$$

where $\sigma_{ji}^{t-1}$ is the accepted deviation fixed by $E_j$ for the $i^{th}$ objective at the generation $(t-1)$.

## 7.3 Case study

This section is devoted to describe a run of NSS-GPA on a case study with 10 arbitrary chosen DMs and 4 objectives. $\alpha$, $\beta$ and *MaxIter* are settled to 0.5, 0.8 and 50 respectively. $\sigma_{\max}^{norm}$ is set to 0.3. These parameters are fixed by a human Moderator agent. Table 7.1 shows the initial DM agents' reference points in addition to the accepted deviations (mentioned between parentheses). The DM interacts with the NSS-GPA through a guided user interface (cf. figure 7.1) which is composed with six panels:

- **Agent Preferences**: where the DM can see the reference points of all DM agents.
- **Collective Preferences**: where the DM can see the collective opinion (i.e., the SRP).
- **Moderator Recommendations**: where the DM receives the global advice rules from the Moderator agent.
- **Agent Proposals:** where the human DM receives proposals from other DM agents. The user can accept or deny such proposals.
- **Requests:** where the DM can send requests to other DM agents and can verify for each one of his/her sent requests whether it was accepted or denied.
- **Preference Update:** where the DM can update his/her reference point and his/her accepted deviation vector.

Based on the used parameter setting, the 10 human DMs have confronted their preferences, through NSS-GPA, while being supervised and guided by the Moderator software agent. Figure 7.2 shows the parallel coordinate plots of the DMs' reference

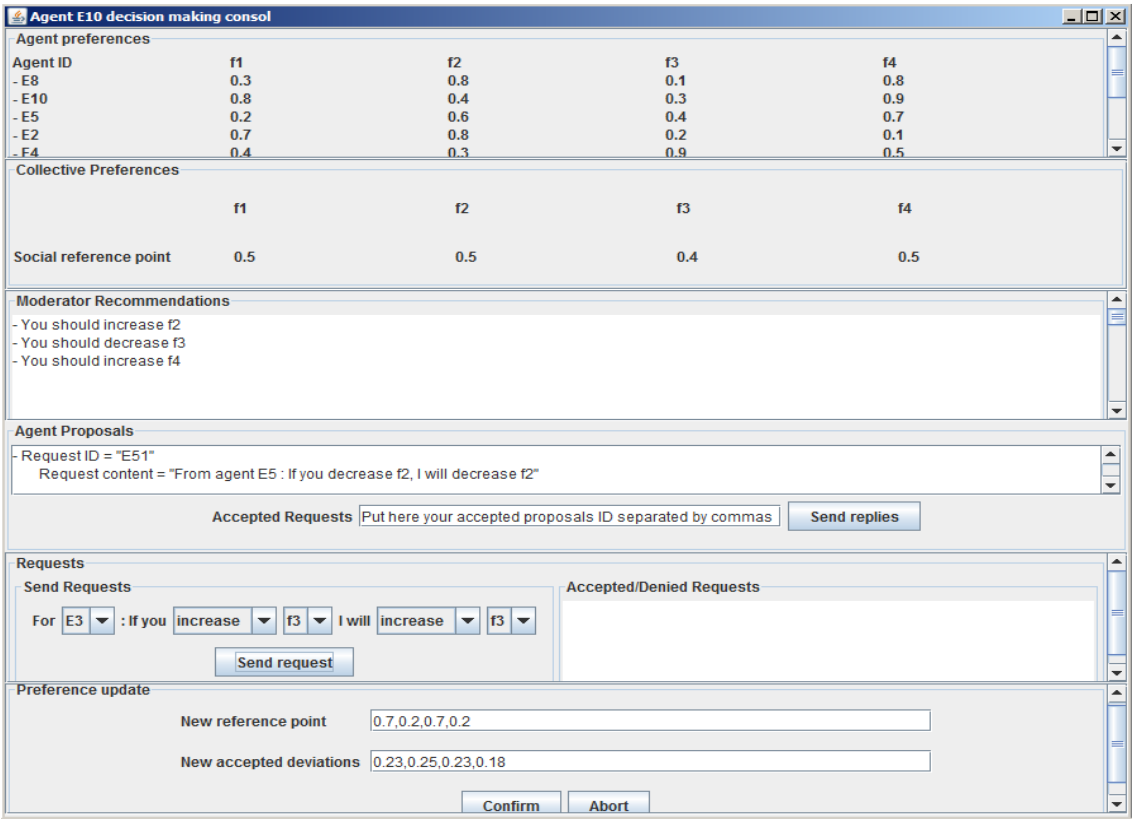
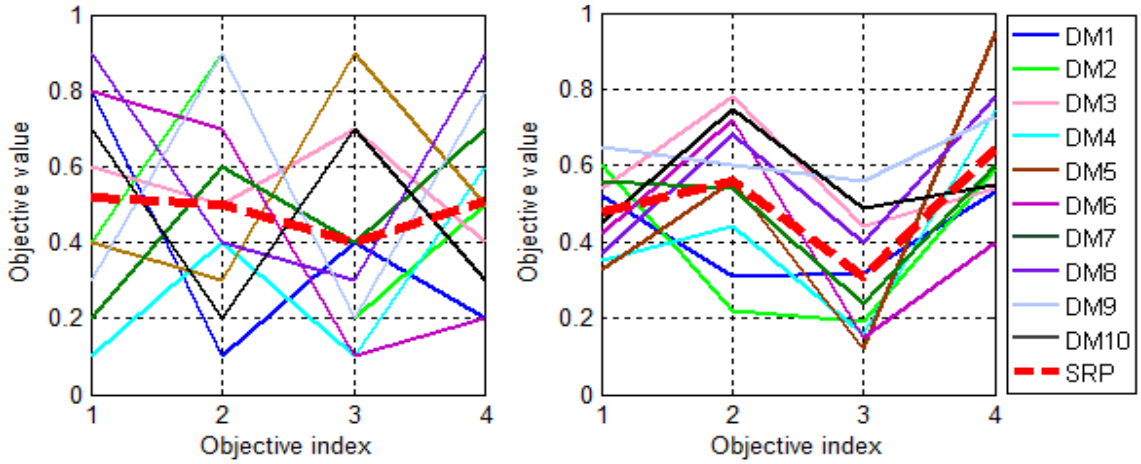



**Figure 7.1 NSS-GPA decision making consol.**

**Table 7.1 Initial DM's aspiration levels (± accepted deviations).**

| | $f_1$ | $f_2$ | $f_3$ | $f_4$ |
|---|---|---|---|---|
| **DM1** | 0.8 (± 0.10) | 0.1 (± 0.05) | 0.4 (± 0.18) | 0.2 (± 0.11) |
| **DM2** | 0.4 (± 0.18) | 0.9 (± 0.09) | 0.2 (± 0.15) | 0.5 (± 0.25) |
| **DM3** | 0.6 (± 0.28) | 0.5 (± 0.07) | 0.7 (± 0.22) | 0.4 (± 0.26) |
| **DM4** | 0.1 (± 0.09) | 0.4 (± 0.18) | 0.1 (± 0.05) | 0.6 (± 0.30) |
| **DM5** | 0.4 (± 0.08) | 0.3 (± 0.11) | 0.9 (± 0.09) | 0.5 (± 0.20) |
| **DM6** | 0.8 (± 0.07) | 0.7 (± 0.30) | 0.1 (± 0.09) | 0.2 (± 0.08) |
| **DM7** | 0.2 (± 0.20) | 0.6 (± 0.14) | 0.4 (± 0.25) | 0.7 (± 0.11) |
| **DM8** | 0.9 (± 0.08) | 0.4 (± 0.22) | 0.3 (± 0.11) | 0.9 (± 0.09) |
| **DM9** | 0.3 (± 0.21) | 0.9 (± 0.10) | 0.2 (± 0.21) | 0.8 (± 0.12) |
| **DM10** | 0.7 (± 0.23) | 0.2 (± 0.15) | 0.7 (± 0.16) | 0.3 (± 0.18) |

points (in addition to the SRP): (a) at the beginning of the negotiation process (cf. table 7.1) and (b) at the end of this process. Figure 7.2(a) shows how the initial preferences are so conflicting. In fact, there are large gaps between DMs' reference points themselves. Besides, these initial aspiration level vectors are so conflicting with the




**(a)** **(b)**

**Figure 7.2 DMs' preferences: (a) Initial aspiration level vectors and (b) final aspiration level vectors.**

**Table 7.2 DM profiling statistics.**

| | ***NSR*** | ***NAR*** | ***NDR*** | ***NM*** | ***NARec/NRRec*** |
|---|---|---|---|---|---|
| **DM1** | 61 | 37 | 14 | 1 | 15/36 (41.67%) |
| **DM2** | 8 | 3 | 5 | 5 | 18/29 (62.07%) |
| **DM3** | 42 | 23 | 19 | 2 | 12/18 (66.67%) |
| **DM4** | 63 | 48 | 15 | 0 | 17/26 (65.38%) |
| **DM5** | 11 | 6 | 5 | 5 | 19/31 (61.29%) |
| **DM6** | 21 | 9 | 12 | 4 | 14/33 (42.42%) |
| **DM7** | 81 | 69 | 12 | 0 | 21/28 (75.00%) |
| **DM8** | 89 | 76 | 13 | 0 | 22/25 (88.00%) |
| **DM9** | 14 | 6 | 8 | 3 | 26/38 (68.42%) |
| **DM10** | 54 | 26 | 28 | 1 | 22/28 (78.57%) |

initial SRP (0.52,0.50,0.40,0.51). Figure 7.2(b) illustrates the final reference points at the end of the negotiation. We see from this figure how the final reference points are less conflicting and so convergent towards the final SRP (0.479,0.560,0.307,0.642). We conclude that NSS-GPA has achieved a good consensus between the different DMs about a SRP. We can say that NSS-GPA has succeeded to bring closer DMs' preferences through the negotiation rounds. Table 7.2 presents some statics provided by our system that we call *profiling statistics* since they allow drawing the profile of each DM. These statistics are: (1) *NSR*: the Number of Sent Requests by the DM, (2) *NAR*: the Number of Accepted Requests, (3) *NDR*: the Number of Denied Requests, (4) *NM*: the Number of Manipulations and (5) *NARec*/*NRRec*: the Number of Accepted Recommendations divided by the Number of Received Recommendations (the

acceptance percentage is set between parentheses). From figure 7.2(b), we observe that DM4, DM7 and DM8 are the most satisfied DMs since their final reference points are the nearest to the final SRP. The satisfaction of such DMs may be explained by the obtained results in table 7.2. In fact, these satisfied agents are the most communicating agents since they have large values for the *NSR* statistic. Additionally, DM4, DM7 and DM8 have succeeded to have a large number of accepted requests, they are said to be the most trustworthy agents. DM2 and DM5 are detected as manipulators according to the *NM* values (for a 4-objective case, $NM = 5$ means that the DM agent is a manipulator). DM2 and DM5 are the most dissatisfied DM agents according to figure 7.2(b). This observation may be explained not only by their manipulation behavior but also by their poor communication with other DM agents (cf. *NSR* values from table 7.2). Intuitively, for a particular DM, the larger his/her *NARec*/*NRRec* ratio value is, the greater his/her satisfaction level is. However, this is not sufficient since the DM's satisfaction depends also on its communication skills and attitudes. For example, the ratio *NARec/NRRec* of DM9 is greater than DM1 one. Nevertheless, from figure 7.2(b), we see that DM1 reference point is nearer to the final SRP than DM9 one. This observation can be explained by the superiority of DM1 over DM9 in terms of communication skills and trustworthiness (cf. *NSR*, *NAR* and *NDR* values from table 7.2). We can say that NSS-GPA favors communicating DMs over non-communicating ones.

Once the DMs' negotiation is performed, we can search for the Pareto optimal ROI of the considered MOP by running any reference point-based EMOA (e.g., r-NSGA-II (Ben Said et al. 2010), PBEA (Thiele et al. 2009), R-NSGA-II (Deb et al. 2006), etc.) with the final SRP obtained by NSS-GPA. We choose to perform a run of r-NSGA-II with the final SRP (0.479,0.560,0.307,0.642) on the four-objective DTLZ2 minimization test problem which is described in (Deb et al. 2002b). The population size and the number of generations are set to 200 and 500 respectively. The parameter $\delta$ which controls the breadth of the obtained ROI (Ben Said et al. 2010) is set to 0.3. Figure 7.3 shows the parallel coordinates plot of the obtained preferred solutions. From this figure, we remark that, although the objective values lie in [0,1], most obtained solutions are concentrated near the reference point designed with a dashed bold gray line which would be the region closest to the final SRP furnished by NSS-GPA. When computing $\sum_{i=1}^{n} f_i^2$ for all obtained solutions, the values are found to lie within [1.051, 1.311], thereby meaning that all solutions are near the true Pareto region (since Pareto optimal solutions of DTLZ2 satisfy $\sum_{i=1}^{n} f_i^2 = 1$ (Deb et al. 2002b)). We can say that r-NSGA-II has provided a *social ROI* and, as noted in the first section, each

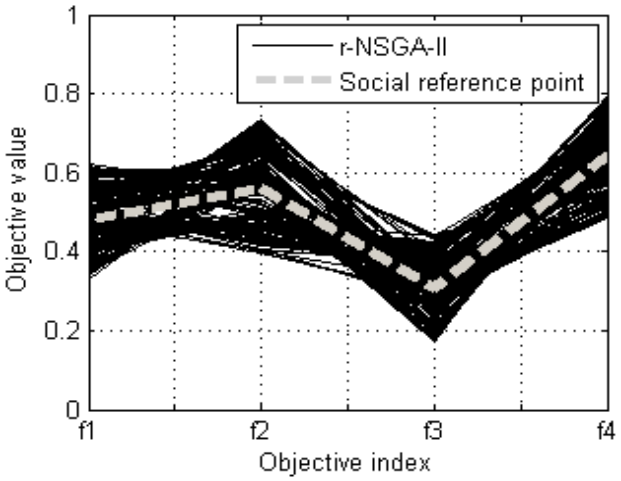



**Figure 7.3 The r-NSGA-II ROI with the final SRP provided by NSS-GPA (0.479, 0.560, 0.307, 0.642) on the four-objective DTLZ2.**

non-dominated solution picked from this region is considered as a satisfying solution for each of the considered DMs.

## 7.4 Application to a practical portfolio selection problem

In this section, we demonstrate the usefulness of NSS-GPA on the bi-objective portfolio selection problem with practicalities (Deb et al. 2011). In a portfolio optimization problem with an asset universe of *n* securities, let $x_i$ $(i = 1,...,n)$ designate the initial capital proportion to be allocated to security *i*. Typically, there are two conflicting objectives: (1) minimize the portfolio risk and (2) maximize the expected portfolio return. These two objectives have received the most attention and such formulation is known as the mean-variance model of Markowitz (1952). The most basic form of this problem can be expressed as follows:

$$\begin{cases} Min \;\; f_1(x) = \sum_{i=1}^{n}\sum_{j=1}^{n} x_i \sigma_{ij} x_j \\ Max \;\; f_2(x) = \sum_{i=1}^{n} r_i x_i \\ \sum_{i=1}^{n} x_i = 1 \\ x_i \geq 0 \quad \forall i = 1,...,n \end{cases} \tag{7.12}$$

The first objective corresponds to the portfolio risk that is usually computed from an $n \times n$ covariance matrix $[\sigma_{ij}]$. The second objective corresponds to the expected

portfolio return as computed from a weighted sum of the individual security expected returns. The first constraint ensures the investment of all funds while the second one ensures the non-negativity of each investment. Such bi-objective problem gives rise to a front of several optimal trade-off solutions which should be found to investigate the risk-return relationships. One way to solve this MOP is to convert it to a SOP using the $\varepsilon$-constraint method as follows:

$$\begin{cases} Min \;\; f_1(x) = \sum_{i=1}^{n}\sum_{j=1}^{n} x_i \sigma_{ij} x_j \\ f_2(x) = \sum_{i=1}^{n} r_i x_i \geq R \\ \sum_{i=1}^{n} x_i = 1 \\ x_i \geq 0 \quad \forall i = 1,...,n \end{cases} \tag{7.13}$$

In order to generate a representative approximation of the Pareto front, the above quadratic problem is solved repetitively for many different values of $R$ which corresponds to the minimal acceptable return value.

According to the study of Deb et al. (2011), it can be expected that almost any solution of (7.13) contains many of its securities at the zero level, i.e., for many $i$, $x_i^* = 0$. It can be also expected, for at least a few securities that $x_i^*$ is a very small quantity. However, to have a practical portfolio, very small investments in any security may not be desired and are to be avoided. Thus, there is the practicality that, for any portfolio to be of interest, there is to be a lower limit on any non-zero investment, i.e., either $x_i^* = 0$ (meaning no investment in the ith security) or $x_i^* \geq \lambda$ (meaning that there is a minimum non-zero investment amount for the $i^{th}$ security). There may also be an upper bound $\omega$ on the proportion of any security in any portfolio. Unfortunately, the solution of (7.13) for any given $R$ does not guarantee the possession of any of these characteristics.

In addition to the above, there is a second practicality and it is about the number of non-zero securities contained in the portfolios along the Pareto front. Over this, a user may wish to exert control. To generate practical portfolios, a user may be interested in specifying a given number of non-zero investments or a range in the number of non-zero investments a portfolio has to contain. This is a cardinality constraint and it has also been the subject of some research attention (Stein et al. 2008; Streichert and

Tanaka-Yamawaki 2006). Taking both practicalities into account, we have the following bi-objective optimization problem:

$$\begin{cases} Min \;\; f_1(x) = \sum_{i=1}^{n}\sum_{j=1}^{n} x_i \sigma_{ij} x_j \\ Max \;\; f_2(x) = \sum_{i=1}^{n} r_i x_i \\ \sum_{i=1}^{n} x_i = 1 \\ x_i = 0 \text{ or } \lambda \le x_i \le \omega \\ d_{\min} \le d(x) \le d_{\max} \\ 0 < \lambda \le \omega < 1 \end{cases} \tag{7.14}$$

where $d(x)$ is given as follows:

$$d(x) = \sum_{i=1}^{n} \begin{cases} 1 & \text{if } x_i > 0 \\ 0 & \text{if } x_i = 0 \end{cases} \tag{7.15}$$

Standard quadratic problem solvers face difficulties in the presence of discontinuities and other complexities. For instance, the second constraint, requires an "*or*" operation. While $x_i = 0$ or $x_i = \lambda$ are allowed, values between the two are not. This introduces discontinuities in the search space. The third constraint involves a parameter *d* which is defined by a discontinuous function of the decision variables given in (7.15). The second and third constraints make the application of standard quadratic problem solvers difficult which is not the case for the MOEAs (Deb et al. 2011).

After presenting the problem details, we can now describe the case study concerning the application of NSS-GPA on this practical bi-objective constrained portfolio selection problem (Bechikh et al. 2012b). We consider an instance used in (Deb et al. 2011) with 88 securities, $\lambda = 0.005$, $\omega = 0.04$ and $d \in [30,45]$. As noted previously in the fifth chapter, the nadir point helps the DM to express his/her preferences so that each aspiration level lies between the ideal value and the nadir one. Figure 7.4 demonstrates the effects of expressing: (1) a reference point in the region delimited by the ideal point and the nadir one *A*(0.7,0.4) and (2) a reference point outside of this region *B*(0.7,0.9). We see, from this figure, that although the two reference points have the same aspiration level value for the first objective, they provide quite different ROIs. This observation emphasizes the importance of the nadir point and the ideal one in the preference expression process. The ideal point can be easily obtained by minimizing

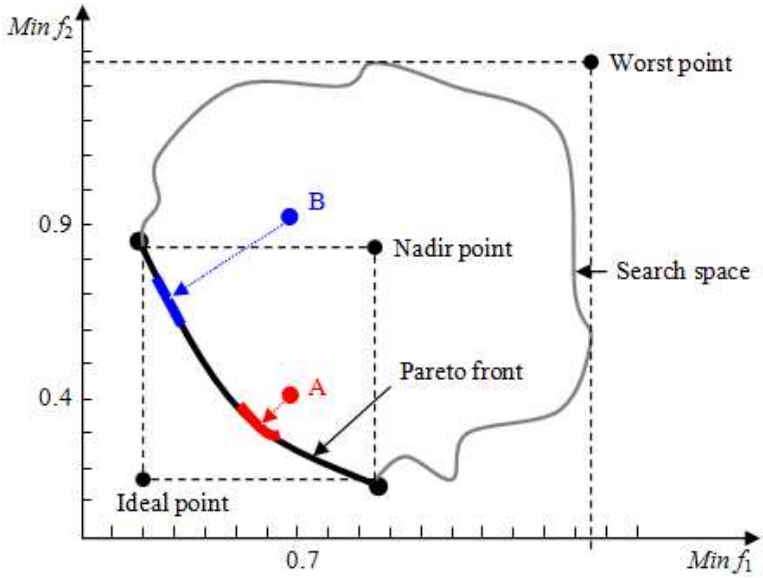



**Figure 7.4 Importance of the nadir objective vector and the ideal one for DM's preference expression.**

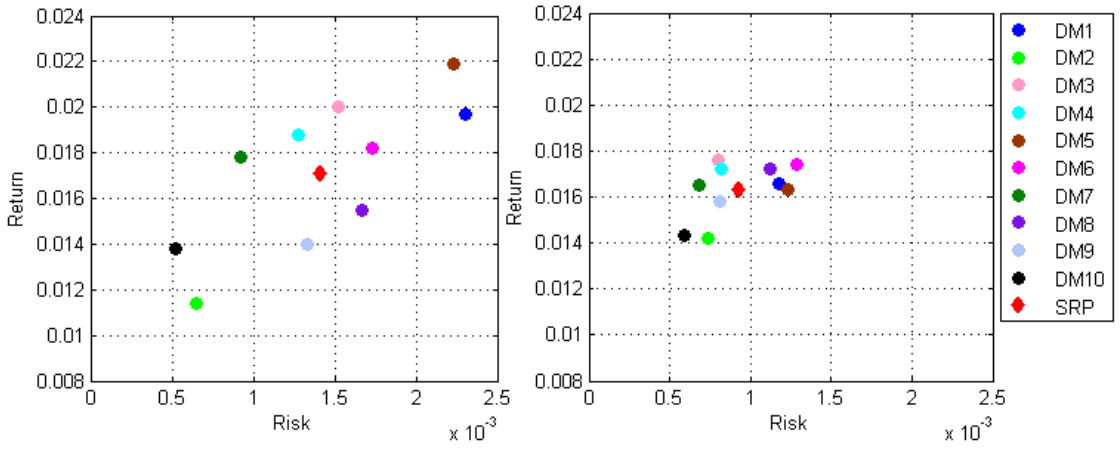
each objective function *individually* which is not the case for the nadir point. For this reason, we apply our algorithm MR-NSGA-II$_N$ (described in the previous chapter) in order to estimate the nadir point. We note that we use, throughout this experiment, the repair mechanisms proposed in (Deb et al. 2011) in order to generate feasible solutions when initializing the population and generating the children. The used parameter setting is described as follows: population size = 200, number of generations = 800, crossover probability = 0.9, crossover (SBX operator) distribution index = 10, mutation probability = 0.1, polynomial mutation index = 50 and $\varepsilon$ = 0.0005. For the SQP-LS used in our algorithm MR-NSGA-II$_N$, the termination criterion is: (1) the norm of descent direction $\|d\| = 10^{-8}$ or (2) the number of allowed iterations $\mu = 40$ is elapsed. The obtained nadir point approximation is *NADIR* (0.0024, 0.0102). For the bi-objective case, the ideal point can be deduced from the nadir one; however the opposite is not true. Thus, we use the outlier solutions found by MR-NSGA-II$_N$ in order to find the ideal point approximation which is found to be *IDEAL* (0.000123, 0.0238). We supply the 10 DMs involved in this experiment with these values in order to express their reference points in addition to their accepted deviation vectors. The used parameter setting for NSS-GPA is as follows: $\alpha$, $\beta$ and *MaxIter* are settled to 0.4, 0.75 and 50 respectively. $\sigma_{\max}^{norm}$ is set to 0.3. These parameters are fixed by a human Moderator agent. Consequently, the risk aspiration levels should lie in the interval [0.000123, 0.002400] with an accepted deviation of (0.002400 - 0.000123) * 0.30 = 0.000683. The return aspiration levels should lie between [0.0102, 0.0238] with an

**Table 7.3 Initial DM's aspiration levels (± accepted deviations) for the practical portfolio selection problem.**

| | Risk | Return |
|---|---|---|
| **DM1** | 0.002300 (± 0.000132) | 0.0197 (± 0.0018) |
| **DM2** | 0.000648 (± 0.000465) | 0.0114 (± 0.0033) |
| **DM3** | 0.001526 (± 0.000206) | 0.0200 (± 0.0030) |
| **DM4** | 0.001276 (± 0.000369) | 0.0188 (± 0.0023) |
| **DM5** | 0.002235 (± 0.000103) | 0.0219 (± 0.0028) |
| **DM6** | 0.001730 (± 0.000476) | 0.0182 (± 0.0024) |
| **DM7** | 0.000925 (± 0.000258) | 0.0178 (± 0.0012) |
| **DM8** | 0.001667 (± 0.000587) | 0.0155 (± 0.0018) |
| **DM9** | 0.001335 (± 0.000355) | 0.0140 (± 0.0011) |
| **DM10** | 0.000524 (± 0.000405) | 0.0138 (± 0.0027) |

accepted deviation of (0.0238 - 0.0102) * 0.30 = 0.00408. Table 7.3 shows the initial DMs' preferences. We remark, from this table, that there are several kinds of DMs (risk-averse investors, risk-neutral investors, risk-seeking investors) with different whishes of return which makes the initial DMs' reference points so conflicting. Based on the used parameter settings, the 10 DMs have confronted their preferences through NSS-GPA. Figure 7.5 shows a confrontation, in the risk-return space, between the: (1) the initial DMs' reference points and (2) the final ones. We see, from this figure, how the initial preferences are so conflicting not only between themselves but also with the initial SRP (0.001416, 0.0171). Figure 7.5(b) illustrates the final reference points at the end of the negotiation. We see, from this figure, how the final reference points are less conflicting and so convergent towards the final SRP (0.000927, 0.0163). We conclude that NSS-GPA has achieved a good consensus between the different DMs about a SRP. We can say that NSS-GPA has succeeded to bring closer the DMs' risk-return aspiration level vectors. We remark also from figure 7.5 that: (1) most DMs' who have decreased their risk aspiration values have also decreased their return ones and (2) most DMs' who have increased their risk aspiration values have also increased their return ones. These two observations emphasize the fact that higher return is usually obtained with higher risk.

We can now apply our reference point-based method r-NSGA-II with the final SRP in order to find the social ROI for the considered instance of the practical portfolio selection problem. We use a population size of 300 and a number of generations of 1500. The non-r-dominance threshold $\delta$ which controls the breadth of the ROI is set to 0.3. We note that we use the same repair mechanisms proposed in (Deb et al. 2011) in




**(a)** **(b)**

**Figure 7.5 DMs' preferences in terms of risk and return: (a) initial aspiration level vectors and (b) final aspiration level vectors.**

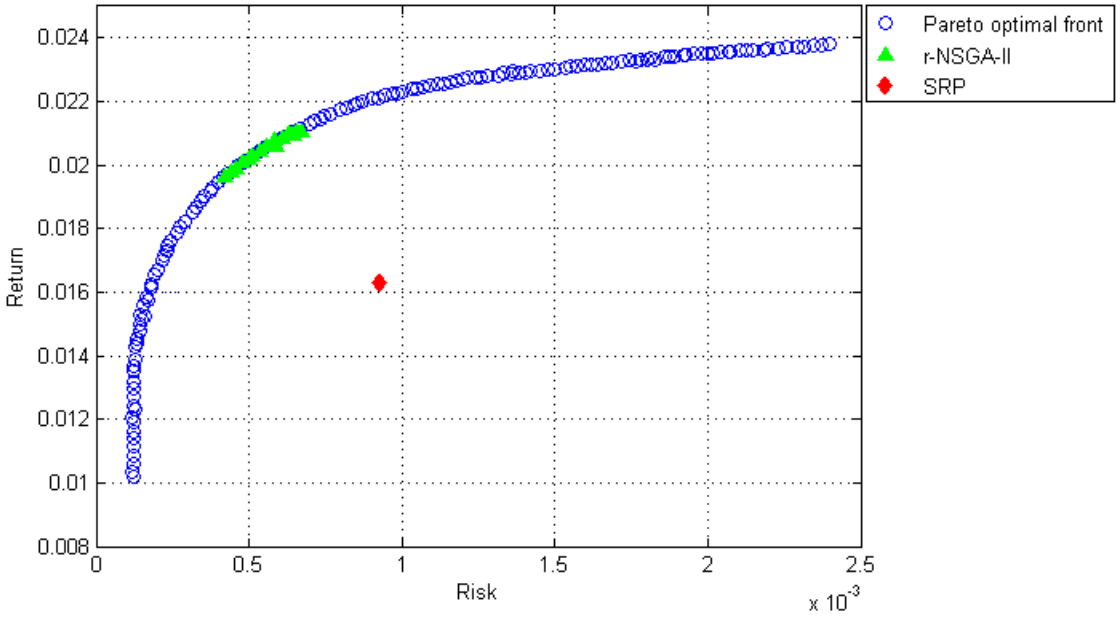



**Figure 7.6 Obtained social ROI by running r-NSGA-II with the final SRP (0.000927, 0.0163) on the practical portfolio selection problem.**

order to generate feasible solutions. Figure 7.6 shows the obtained social ROI designed with green triangles. We see from this figure how this region is composed with the nearest Pareto optimal portfolios to the social reference point in the risk-return space. Besides, we remark that surprisingly the social ROI corresponds to the knee region composed with the worthiest portfolios in terms of risk-return trade-off. Hence, if the DMs make a consensus about selecting a portfolio from the knee region, the negotiation can be ignored and we can directly use the TKR-NSGA-II method (cf. chapter 6) in order to approximate the knee region. Finally, as noted at the beginning of this chapter,

each portfolio selected from the social ROI is considered as a satisfying portfolio for most DMs.

From a computational viewpoint, we note that the customized NSGA-II approach (without using local search) of Deb et al. (2011) needs about 1.5 $10^6$ FEs to approximate the whole Pareto front according to the experimental results presented in the corresponding paper. However, our approach, which uses MR-NSGA-$II_N$ for estimating the nadir point and then r-NSGA-II to approximate the social ROI, requires about 0.7 $10^6$ FEs. This observation emphasizes the computational efficiency of our approach regarding the use of a general-purpose MOEA (without any preference-based mechanism) and then selecting a portfolio in a posteriori manner.

## 7.5 Conclusion

In this chapter, we have proposed an agent-based system for group preference negotiation. Reference point negotiation can be seen as a special case of social choice where the different DMs are searching for a consensus about a SRP. In order to avoid the impossibility results of social choice theory (Arrow 1951, Gibbard 1973, Xia and Conitzer 2008), we have chosen to use the concept of agent-based negotiation by proposing NSS-GPA. In fact, we have seen how the DM's preferences change over time through a negotiation round to another one based on the current group preferences in addition to the current SRP. Indeed, each DM tries to attract the SRP towards his/her own preferences by communicating with other DMs and following the global direction rules. The originality of this work resides in the consideration of different human profiles and attitudes that a DM can have during a negotiation. NSS-GPA was assessed through a case study with 10 DMs and four-objectives. It was demonstrated to: (1) discourage dictatorship, manipulation and untrustworthiness behaviors and (2) encourage communication between the negotiators. Additionally, NSS-GPA was shown to be helpful in confronting and adjusting DMs' preferences; thereby providing a satisfying SRP to be injected in any reference point-based MOEA with the purpose to guide the search towards a social ROI. Subsequently, the DMs can select the solution to realize for the considered MOP from the obtained region. Finally, we have valorized the usefulness of our system through the resolution of a practical instance of the portfolio selection problem with two objectives (risk and return) and 10 DMs and the obtained results were promising.

# Chapter 8

# Conclusions and Future Works

## 8.1 Key results

Through this doctoral thesis, we have contributed to the incorporation of DM's preferences in EMO. In detail, the major contributions could be summarized as follows:

- The proposition of a new reference point-based dominance relation, i.e., the r-dominance, which has the ability to create a strict partial order on the set of Pareto-equivalent solutions. Such characteristic makes such a relation able to guide the search towards the interesting parts of the Pareto optimal region based on the DM' preferences expressed as a set of aspiration levels. After integrating the new dominance relation in NSGA-II, the efficacy and the usefulness of the modified procedure (i.e., r-NSGA-II) have been assessed through two- to ten-objective test problems a priori and interactively. Moreover, the proposed approach provided competitive and better results when compared to other recently proposed preference-based EMO approaches.
- The suggestion of new approaches for approximating knee regions which represent a form of implicit DM's preferences. The proposed approaches, i.e., KR-NSGA-II and its enhanced version TKR-NSGA-II, have demonstrated their effectiveness and efficiency in discovering knee regions on a set of knee-based test problems commonly used to assess the ability of MOEAs to find knee regions. Moreover, the interactive versions of these approaches are demonstrated as tools to handle the case where the DM has no a priori information about the number of existing knee regions in the Pareto optimal front. Moreover, the TKR-NSGA-II has been confronted to recently proposed knee-based MOEAs including KR-NSGA-II via a comparative experimental study. The obtained results have shown the superiority of the TKR-NSGA-II over the other approaches.
- The proposition of a new class of algorithms for nadir point estimation using EAs, i.e., the extreme-region-of-interest-to-nadir class. The MR-NSGA-$II_N$, which is an instance of our newly suggested class, has been assessed on a set of three- to

twenty-objective unconstrained/constrained problems. The proposed approach has shown competitive and better results when compared to other recently proposed nadir point estimation approaches.

- The incorporation of the preferences of a group of DMs in MOEAs which represents an omitted problematic in the EMO research field. We have addressed this problematic by proposing an agent-based negotiation support system (i.e., NSS-GPA) that allows the different DMs to negotiate their reference points before the beginning of the optimization process. The system output is a SRP to be injected subsequently in any reference point-based MOEA, such as r-NSGA-II, in order to guide the search towards a social ROI. NSS-GPA has been demonstrated to: (1) discourage dictatorship, manipulation and untrustworthiness behaviors and (2) encourage communication between the negotiators. Additionally, the usefulness of the collaboration between NSS-GPA, MR-NSGA-II$_N$ and r-NSGA-II has been shown on a practical instance of the portfolio selection problem.

## 8.2 Future works

Regarding what has already been achieved in the preference-based EMO research field, several open questions and perspectives are still to investigate. These perspectives can be classified into two main classes. The first one concerns the development of the hybridization between EMO and decision making. The second one concerns the exploitation of preference incorporation in other EMO related subfields.

### 8.2.1 Developing the hybridization between EMO and decision making

Among the possible future works that fall in this class of perspectives, we propose:

- *Preference elicitation*: An important issue in preference-based algorithms is how to interact with the DM(s) and how to guide the DM(s) during the interaction in order to elicit his/her preferences in a consistent manner. It seems to be interesting to design some specific methods for preference extraction.
- *Interaction between EMO and group decision making*: The NSS-GPA supposes that reference points should be negotiated and aggregated before the beginning of the search in order to find a satisfying SRP. It would be interesting to investigate the possibility of designing an agent-based distributed system where each software Assistant agent is deployed on a single computer and launch the preference-based MOEA with the corresponding DM's reference point. In this way, the preference negotiation and the optimization process are performed in parallel. Additionally, the interaction between the different agents could be made based on an argumentation

system that exploits information issued from the current states of the MOEAs' populations in the search space. Furthermore, we can enrich this perspective by considering the hierarchy and the coalitions between the different DMs.

- *Investigating the incorporation of preferences within other metaheuristics*: In addition to EAs, several other metaheuristics have demonstrated their efficacy in solving MOPs such as PSO (Reyes-Sierra and Coello 2006) and Artificial Immune System (AIS) (Freschi et al. 2009). Hence, it is interesting to incorporate preference information in such population-based metaheuristics (Azzouz et al. 2012).

### 8.2.2 Exploiting preference incorporation in other EMO issues

Among the possible ideas that fall in this class of perspectives, we suggest:

- *Many-objective optimization*: When dealing with MOPs involving more than three or four objective functions, the task of approximating the Pareto front becomes more difficult due to: (1) the loss of the ability of MOEAs to adequately order solutions in terms of objective function values and (2) the exponential increase in the number of solutions required to generate a good sample of the Pareto front. One of the attempts to tackle such problems is to use modified forms of the Pareto dominance based on DM's preferences (Adra et al. 2007); thereby finding only portion of the front that satisfies the DM, i.e., the ROI. For example, in the fifth chapter, our r-NSGA-II algorithm has demonstrated its ability to find the ROI for up to ten objectives. Hence, more elaborated preference-based dominance relations could be designed to solve many-objective problems.
- *Expensive multi-objective optimization*: In some engineering MOPs, the evaluation of candidate solutions could be extremely computationally and/or financially expensive since it requires time-consuming computer simulations. Therefore, a method is of great practical interest if it is able to produce reasonably good solutions within a given often very tight budget in terms of computational time (Zhang et al. 2010). As demonstrated through our thesis, approximating a ROI requires fewer FEs than finding the whole Pareto front. Consequently, the incorporation of DM's preferences within an EMO approach specifically designed for expensive MOPs may further minimize the required number of FEs; thereby decreasing the computational time.
- *Robustness*: In optimization studies including multi-objective optimization, the main focus is placed on finding the global optimum or global Pareto optimal solutions representing the best possible objective values. However, in practice, users may not always be interested in finding the so-called global best solutions,

particularly when these solutions are quite sensitive to variable perturbations which cannot be avoided in practice. In such cases, practitioners are interested in finding the robust solutions which are less sensitive to small perturbations in variables (Deb and Gupta 2006). Hence, it is interesting to search for the *robust ROI*.

- *Further applications to real world MOPs*: We remark, from the described works in this thesis, that most researchers have assessed their preference-based MOEAs on academic benchmarks such as the ZDT and DTLZ benchmark suites. For this reason, the researchers are encouraged to apply these algorithms to handle real world problems, for the discrete case (Loukil et al. 2007, Benlic and Hao 2012) and the continuous one (Belgasmi et al. 2008), in an attempt to further valorize the preference-based EMO research field.

# Bibliography


Adra S F, Griffin I, Fleming P J (2007) A Comparative study of progressive preference articulation techniques for multiobjective optimisation. In: Proceedings of the 4th international conference on Evolutionary Multi-criterion Optimization (EMO'07), pp 908–921.

Allmendinger R, Li X, Branke J (2008) Reference point-based particle swarm optimization using a steady-state approach. In: Proceedings of the 7th international conference on Simulated Evolution and Learning (SEAL'08), pp 200–209.

Alves M J, Costa J P (2009a) An exact method for computing the nadir values in multiple objective linear programming. European Journal of Operational Research, 198(2):637–646.

Alves M J, Costa J P (2009b) An evolutionary algorithm to estimate the nadir point in MOLP. In: Proceedings of the 5th international conference on Evolutionary Multicriterion Optimization (EMO'09), pp 540–553.

Arrow K (1951) Social choice and individual values. John Wiley & Sons, New York, USA.

Auger A, Bader J, Brockhoff D, Zitzler E (2009) Articulating user preferences in many-objective problems by sampling the weighted hypervolume. In: Proceedings of the 11th Genetic and Evolutionary Computation COnference (GECCO'09), pp 555–562.

Azzouz R, **Bechikh S**, Ben Said L (2012) Articulating decision maker's preference information within multiobjective artificial immune systems. In: Proceedings of the 24th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'12), accepted (Contribution rate: 40%).

Battiti R, Brunato M, Mascia F (2008) Reactive search and intelligent optimization. Operations Research/Computer Science Interfaces Series, Volume 45, Springer-Verlag.

Battiti R, Passerini A (2010) Brain–computer evolutionary multiobjective optimization: A genetic algorithm adapting to the decision maker. IEEE Transactions on Evolutionary Computation, 14(5):671–687.

**Bechikh S**, Belgasmi N, Ben Said L, Ghédira K (2008) PHC-NSGA-II: A novel multi-objective memetic algorithm for continuous optimization. In: Proceedings of the 20th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'08), pp 180–189 (Contribution rate: 75%).

**Bechikh S**, Ben Said L, Ghédira K (2010a) Searching for knee regions in multi-objective optimization using mobile reference points. In: Proceedings of the 25th ACM symposium on Applied Computing (ACM SAC'10–Best Paper Award), pp 1118–1125 (Contribution rate: 85%).

**Bechikh S**, Ben Said L, Ghédira K (2010b) Estimating nadir point in multi-objective optimization using mobile reference points. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC'10), pp 2129–2137 (Contribution rate: 90%).

**Bechikh S**, Ben Said L, Ghédira K (2011a) Searching for knee regions of the Pareto front using mobile reference points. Soft Computing – A Fusion of Foundations, Methodologies and Applications, 15(9):1807–1823 (Contribution rate: 90%).

**Bechikh S**, Ben Said L, Ghédira K (2011b) Negotiating decision makers' reference points for group preference-based evolutionary multi-objective optimization. In: Proceedings of the 11th international conference on Hybrid Intelligent Systems (HIS'11), pp 377–382 (Contribution rate: 90%).

**Bechikh S**, Ben Said L, Ghédira K (2012a) Preference incorporation in evolutionary multi-objective optimization: A survey of the-state-of-the-art. Submitted to ACM Computing Surveys, under review: minor revision (Contribution rate: 90%).

**Bechikh S**, Ben Said L, Ghédira K (2012b) Group preference-based evolutionary multi-objective optimization with non-equally important decision makers: Application to the portfolio selection problem. International Journal of Computer Information Systems and Industrial Management Applications, accepted (Contribution rate: 90%).

Belgasmi N, Ben Said L, Ghédira K (2008) Evolutionary multiobjective optimization of the multi-location transshipment problem. Operational Research, 8(2):167–183.

Bellifemine F L, Caire G, Greenwood D (2007) Developing multi-agent systems with JADE. Wiley, USA.

Ben Jaâfar I, Ghédira K (2007) Coordination based multiple criteria decision making. Journal of Decision Systems, 16(2):37–56.

Ben Said L, **Bechikh S**, Ghédira K (2010) The r-dominance: A new dominance relation for interactive evolutionary multi-criteria decision making. IEEE Transactions on Evolutionary Computation, 14(5):801–818 (Contribution rate: 85%).

Benayoun R, de Montgolfier J, Tergny J, Laritchev P (1971) Linear programming with multiple objective functions: Step method (STEM). Mathematical Programming, 1(3):366–375.

Benlic U, Hao J K (2012) Hybrid metaheuristics for the graph partitioning problem. In: Talbi EG (editor), Hybrid metaheuristics, studies in computational intelligence, chapter 9.

Beume N, Naujoks B, Emmerich M (2007) SMS-EMOA: Multiobjective selection based on dominated hypervolume. European Journal of Operational Research, 181(3):1653–1669.

Biermann D, Weinert K, Wagner T (2008) Model-based optimization revisited: Toward real-world processes. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC'08), pp 2980–2987.

Bleuler S, Laumanns M, Thiele L, Zitzler E (2003) PISA: A platform and programming language independent interface for search algorithms. In: Proceedings of the 2nd international conference on Evolutionary Multicriterion Optimization (EMO'03), pp 494–508.

Bosman P A N, de Jong E D (2005) Exploiting gradient information in numerical multi-objective evolutionary optimization. In: Proceedings of the 7th Genetic and Evolutionary Computation COnference (GECCO'05), pp 755–762.

Bouyssou D, Marchant T, Perny P (2009) Social choice theory and multicriteria decision aiding. In: Bouyssou D, Dubois D, Pirlot M and Prade H (editors), Decision-making process concepts and methods, pp 779–810.

Branke J (2008) Consideration of partial user preferences in evolutionary multiobjective optimization. In: Branke J, Deb K, Miettinen K and Slowinski R (editors), Multiobjective Optimization—Interactive and Evolutionary Approaches, pp 157–178.

Branke J, Deb K (2004) Integrating user preferences into evolutionary multi-objective optimization. In: Yaochu Jin (editor), Knowledge Incorporation in Evolutionary Computation, pp 461–478.

Branke J, Deb K, Dierolf H, Osswald M (2004) Finding knees in multiobjective optimization. In: Proceedings of 8th international conference on Parallel Problem Solving from Nature (PPSN VIII), pp 722–731.

Branke J, Greco S, Slowinski R, Zielniewski P (2010) Interactive evolutionary multiobjective optimization driven by robust ordinal regression. Bulletin of the Polish Academy of Science, 58(3):347–358.

Branke J, Kaussler T, Schmeck H (2001) Guidance in evolutionary multi-objective optimization. Advances in Engineering Software, 32(6):499–507.

Buchanan J T (1997) A naive approach for solving MCDM problems: The GUESS method. Journal of the Operational Research Society, 48(2):202–206.

Cahon J L (1978) Multiobjective programming and planning. Academic Press, New York.

Charnes A, Cooper W W, Ferguson R (1955) Optimal estimation of executive compensation by linear programming. Management Science, 1(2):138–151.

Chevaleyre Y, Endriss U, Lang J, Maudet N (2007) A short introduction to computational social choice. In: Proceedings of the 33rd conference on current trends in theory and practice of computer science (SOFSEM'07), pp 51–69.

Coello Coello C A (2000) Handling preference in evolutionary multiobjective optimization: A survey. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC'00), pp 30–37.

Coello Coello C A, Lamont G B, Van Veldhuizen D A (2007) Evolutionary algorithms for solving multi-objective problems (second edition). Springer, New York.

Cohen W W, Schapire R E, Singer Y (1999) Learning to order things. Journal of Artificial Intelligence Research, 10(1):243–270.

Conitzer V (2006) Computational aspects of preference aggregation. PhD thesis, Carnegie Mellon University, USA.

Corne D W, Jerram N R, Knowles J D, Oates M J (2001) PESA-II: Region-based selection in evolutionary multiobjective optimization. In: Proceedings of the 3rd Genetic and Evolutionary Computation COnference (GECCO'01), pp 283–290.

Corne D W, Knowles J D (2000) The Pareto-envelope based selection algorithm for multiobjective optimization. In: Proceedings of the 6th international conference on Parallel Problem Solving from Nature (PPSN VI), pp 839–848.

Cvetkovic D, Parmee I C (2002) Preferences and their application in evolutionary multiobjective optimisation. IEEE Transactions on Evolutionary Computation, 6(1):42–57.

Das I (1999) On characterizing the knee of the Pareto curve based on normal-boundary intersection. Structural Optimization, 18(2–3):107–115.

Das I, Dennis J E (1998) Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. SIAM Journal of Optimization, 8(3):631–657.

Da Fonseca V G, Fonseca C M, HALL A O (2001) Inferential performance assessment of stochastic optimisers and the attainment function. In: Proceedings of the 1st international conference on Evolutionary Multi-Criterion Optimization (EMO'01), pp 213–225.

Deb K (1999) Multi-objective evolutionary algorithms: Introducing bias among Pareto optimal solutions. KanGAL Report Number 99002, Kanpur Genetic Algorithms Laboratory (KanGAL), Indian Institute of Technology Kanpur, India.

Deb K (2000) An efficient constraint handling method for genetic algorithms. Computer methods in applied mechanics and engineering, 186(2):311–338.

Deb K (2001) Multi-objective optimization using evolutionary algorithms. John Wiley and Sons, Ltd, New York, USA.

Deb K (2011) Single and multi-Objective dynamic optimization: Two tales from an evolutionary perspective. KanGAL Report Number 2011004, Kanpur Genetic Algorithms Laboratory (KanGAL), Indian Institute of Technology Kanpur, India.

Deb K, Agrawal S (1995) Simulated binary crossover for continuous search space. Complex Systems, 9(1):115–148.

Deb K, Chaudhuri S, Miettinen K (2006b) Towards estimating nadir objective vector using evolutionary approaches. In: Proceedings of the 8th Genetic and Evolutionary Computation COnference (GECCO'O6), pp 643–650.

Deb K, Gupta H (2006) Introducing robustness in multi-objective optimization. Evolutionary Computation, 14(4):463–494.

Deb K, Kumar A (2007a) Interactive evolutionary multi-objective optimization and decision making using reference direction method. In: Proceedings of the 9th Genetic and Evolutionary Computation COnference (GECCO'07), pp 781–788.

Deb K, Kumar A (2007b) Light beam search based multi-objective optimization using evolutionary algorithms. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC'07), pp 2125–2132.

Deb K, Miettinen K (2008) A review of nadir point estimation procedures using evolutionary approaches: A tale of dimensionality reduction. KanGAL Report Number 2008004, Kanpur Genetic Algorithms Laboratory (KanGAL), Indian Institute of Technology Kanpur, India.

Deb K, Miettinen K, Chaudhuri S (2009a) An estimation of nadir objective vector using a hybrid evolutionary-cum-local-search procedure. Working paper W–470, Helsinki School of Economics, Helsinki, Finland.

Deb K, Miettinen K, Sharma D (2009b) A hybrid integrated multiobjective optimization procedure for estimating nadir point. In: Proceedings of the 5th international conference on Evolutionary Multi-criterion Optimization (EMO'09), pp 569–583.

Deb. K, Pratap A, Agarwal S, Meyarivan T (2000). A fast and elitist multi-objective genetic algorithm: NSGA-II. KanGAL Report Number 2000001, Kanpur Genetic Algorithms Laboratory (KanGAL), Indian Institute of Technology, Kanpur, India.

Deb K, Pratap A, Agarwal S, Meyarivan T (2002a) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation, 6(2):182–197.

Deb K, Sinha A, Korhonen P, Wallenius J (2010) An interactive evolutionary multiobjective optimization method based on progressively approximated value functions. IEEE Transactions on Evolutionary Computation, 14(5):723–739.

Deb K, Steuer R E, Tewari R, Tewari Ra (2011) Bi-objective portfolio optimization using a customized hybrid NSGA-II procedure. In: Proceedings of the 6th international conference on Evolutionary Multi-criterion Optimization (EMO'11), pp 358–373.

Deb K, Sundar J, Bhaskara U, Chaudhuri S (2006a) Reference point based multi-objective optimization using evolutionary algorithms. International Journal of Computational Intelligence Research, 2(3):273–286.

Deb K, Thiele L, Laumanns M, Zitzler E (2002b) Scalable multiobjective optimization test problems. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC'02), pp 825–830.

Dessouky M I, Ghiassi M, Davis W J (1986) Estimates of the minimum nondominated criterion values in multiple-criteria decision making. Engineering Costs and Production Economics, 10(2):95–104.

d'Anjou J, Fairbrother S, Kehn D, Kellerman J, McCarthy P (2004) The Java developer's guide to Eclipse. Addison-Wesley, USA.

Ehrgott M, Tenfelde-Podehl D (2003) Computation of ideal and nadir values and implications for their use in MCDM methods. European Journal of Operational Research, 151(1):119–139.

Farina M, Amato P (2004) A fuzzy definition of 'optimality' for manycriteria optimization problems. IEEE Transactions on Systems, Man and Cybernetics, 34(3):315–326.

Fernandez E, Lopez E, Bernal S, Coello Coello C A, Navarroa J (2010) Evolutionary multiobjective optimization using an outranking-based dominance generalization. Computers & Operations Research, 37(2):390–395.

Fernandez E, Lopez E, Lopez F, Coello Coello C A (2011) Increasing selective pressure towards the best compromise in evolutionary multiobjective optimization: The extended NOSGA method. Information Sciences, 181(1):44–56.

Figueira J, Greco S, Slowinski R (2008) Identifying the "most representative" value function among all compatible value functions in the GRIP method. In: Proceedings of the 68th meeting of the EURO Working Group on MCDA, CD-ROM.

Figueira J, Greco S, Slowinski R (2009) Building a set of additive value functions representing a reference preorder and intensities of preference: GRIP method. European Journal of Operational Research, 195(2):460–486

Fleischer M (2003) The measure of Pareto optima: Applications to multi-objective metaheuristics. In: Proceedings of the 2nd international conference on Evolutionary Multi-Criterion Optimization (EMO'03), pp 519–533

Fonseca C (1995) Multiobjective genetic algorithms with application to control engineering problems. PhD thesis, Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, UK.

Fonseca C M, Fleming P J (1993) Genetic algorithms for multi-objective optimization: Formulation, discussion and generalization. In: Proceedings of the 5th International Conference on Genetic Algorithms (ICGA'93), pp 416–423.

Fowler J W, Gel E S, Köksalan M, Korhonen P, Marquis J L, Wallenius J (2010) Interactive evolutionary multi-objective optimization for quasi-concave preference functions. European Journal of Operational Research, 206(2):417–425.

Freschi F, Coello Coello C A, Repetto M (2009) Multiobjective optimization and artificial immune systems: A review. In: Hongwei M (editor), Handbook of Research on artificial immune systems and natural computing: Applying complex adaptive technologies, pp 1–21.

Gibbard A (1973) Manipulation of voting schemes: a general result. Econometrica, 41(4):587–601.

Greco S, Mousseau V, Slowinski R (2008) Ordinal regression revisited: Multiple criteria ranking with a set of additive value functions. European Journal of Operational Research, 191(2):415–435.

Greco S, Slowinski R, Figueira J, Mousseau V (2010) Robust ordinal regression. In: Ehrgott M, Figueira J and Greco S (editors), Trends in multiple criteria decision analysis, pp 241–283.

Greenwood G W, Hu X S, D'Ambrosio J G (1997) Fitness functions for multiple objective optimization problems: combining preferences with Pareto rankings. In: Belew R K and Vose M D (edtiors) Foundations of genetic algorithms, pp 437–455.

Hajela P, Lin C Y (1992) Genetic search strategies in multicriterion optimal design. Structural Optimization, 4(2):99–107.

Harrington J (1965) The desirability function. Industrial Quality Control, 21(10):494–498.

Herrera-Viedma E, Alonso S, Chiclana F, Herrera F (2007) A consensus model for group decision making with incomplete fuzzy preference relations. IEEE Transactions on Systems, Man and Cybernetics, Part B, 37(1):176–189.

Holland J H (1975) Adaptation in natural and artificial systems. University of Michigan Press, Ann Arbor, USA.

Horn J, Nafpliotis N (1994) A niched Pareto genetic algorithm for multiobjective optimization. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC'94), pp 82–87.

Hu X, Huang Z, Wang Z (2003) Hybridization of the multi-objective evolutionary algorithms and the gradient-based algorithms. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC'03), pp 870–877.

Hughes E J (2005) Evolutionary many-objective optimization: Many once or one many? In: Proceedings of IEEE Congress Evolutionary Computation (CEC'05), pp 222–227.

Iserman H, Steuer R E (1998) Computational experience concerning payoff tables and minimum criterion values over the efficient set. European Journal of Operational Research, 33(1):91–97.

Jaszkiewicz A, Slowinski R (1999) The light beam search approach–An overview of methodology and applications. European Journal of Operation Research, 113(2):300–314.

Jin Y, Okabe T, Sendhoff B (2001) Adapting weighted aggregation for multiobjective evolution strategies. In: Proceedings of the 1st international conference on Evolutionary Multicriterion Optimization (EMO'01), pp 96–110.

Jin Y, Sendhoff B (2002) Incorporation of fuzzy preferences into evolutionary multiobjective optimization. In: Proceedings of the 4th Asia-Pacific conference on Simulated Evolution And Learning (SEAL'02), pp 26–30.

Jones D (2001) A taxonomy of global optimization methods based on response surfaces. Journal of Global Optimization, 21(4):345–383.

Karahan I, Köksalan M (2010) A territory defining multiobjective evolutionary algorithm and preference incorporation. IEEE Transactions on Evolutionary Computation, 14(4):636–664.

Klamroth K, Miettinen K (2008) Integrating approximation and interactive decision making in multicrietria optimization. Operations Research, 56(1):222–234.

Knowles J D, Corne D W (1999) The Pareto Archived Evolution Strategy: A New Baseline Algorithm for Pareto Multiobjective Optimisation. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC'99), pp 98–105.

Knowles J D, Corne D W (2000) Approximating the nondominated front using the Pareto archived evolution strategy. Evolutionary Computation, 8(2), pp 149–172.

Köksalan M, Karahan I (2010) An interactive territory defining evolutionary algorithm: iTDEA. IEEE Transactions on Evolutionary Computation, 14(5):702–722.

Korhonen P (1987) The specification of a reference direction using the analytic hierarchy process. Mathematical Modelling, 9(3–5):361–368.

Korhonen P, Laakso J (1986a) A visual interactive method for solving the multiple criteria problem. European Journal of Operational Research, 24(2):277–287.

Korhonen P, Laakso J (1986b) Solving generalized goal programming problems using a visual interactive approach. European Journal of Operational Research, 26(3):355–363.

Korhonen P, Salo S, Steuer R E (1997) A heuristic for estimating nadir criterion values in multiple objective linear programming. Operations Research, 45(5):751–757.

Korhonen P, Yu G Y (1997) A reference direction approach to multiple objective quadratic-linear programming. European Journal of Operational Research, 102(3):601–610.

Kumar A, Sharma D, Deb K (2007) A hybrid multi-Objective optimization procedure using PCX based NSGA-II and sequential quadratic programming. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC'07), pp 25–28.

Kursawe F (1991) A variant of evolution strategies for vector optimization. In: Proceedings of the 1st international conference on Parallel Problem Solving from Nature (PPSN I), pp 193–197.

Li X (2003) A non-dominated sorting particle swarm optimizer for multiobjective optimization. In: Proceedings of the 5th Genetic and Evolutionary Computation COnference (GECCO'03), pp 37–48.

Li X (2004) Better spread and convergence: Particle swarm multiobjective optimization using the maximin fitness function. In: Proceedings of the 6th Genetic and Evolutionary Computation COnference (GECCO'04), pp 117–128.

Loukil T, Teghem J, Fortemps P (2007) A multi-objective production scheduling case study solved by simulated annealing. European Journal of Operational Research, 179(3):709–722.

López Jaimes A, Coello Coello C A, Barrientos J E U (2009) Online objective reduction to deal with many-objective problems. In: Proceedings of the 5th international conference on Evolutionary Multicriterion Optimization (EMO'09), pp 423–437.

Markowitz H M (1952) Portfolio selection. The Journal of Finance, 7(1):77–91.

Martello S, Toth P (1990) Knapsack problems: Algorithms and computer implementations. John Wiley and Sons, Ltd, New York, USA.

Mattson C A, Mullur A A, Messac A (2004) Smart Pareto filter: Obtaining a minimal representation of multiobjective design space. Engineering Optimization, 36(6):721–740.

Metev B, Vassilev V (2003) A method for nadir point estimation in MOLP Problems. Cybernetics and Information Technologies, 3(2):15–24.

Miettinen K (1999) Nonlinear multiobjective optimization. Kluwer Academic Publishers, Boston, USA.

Miettinen K, Mäkelä M M (2006) Synchronous approach in interactive multiobjective optimization. European Journal of Operational Research, 170(3):909–922.

Miettinen K, Mäkelä M M, Kaario K (2006), Experiments with classification-based scalarizing functions in interactive multiobjective optimization. European Journal of Operational Research, 175(2):931–947.

Molina J, Santana L V, Hernández-Díaz A G, Coello Coello C A, Caballero R (2009) g-dominance: Reference point based dominance for multiobjective metaheuristics. European Journal of operational Research, 197(2):685–692.

Morge M, Beaune P (2004) A negotiation support system based on a multi-agent system: Specificity and preference relations on arguments. In: Proceedings of the 19th ACM Symposium on Applied Computing (SAC'04), pp 474–478.

Moscato P (1989) On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms. Caltech concurrent computation program, C3P Report 826.

Mousseau V, Dias L C (2004) Valued outranking relations in ELECTRE providing manageable disaggregation procedures. European Journal of Operational Research, 156(2):467– 482.

Pfeiffer J, Golle U, Rothlauf F (2008) Reference point based multi-objective evolutionary algorithms for group decisions. In: Proceedings of Genetic and Evolutionary Computation COnference (GECCO'08), pp 697–704.

Rachmawati L (2008) Incorporating decision maker preference in multi-objective evolutionary algorithms. PhD thesis, National university of Singapore, Singapore.

Rachmawati L, Srinivasan D (2006a) A multi-objective evolutionary algorithm with weighted sum niching for convergence on knee regions. In: Proceedings of the 8th Genetic and Evolutionary Computation COnference (GECCO'06), pp 749–750.

Rachmawati L, Srinivasan D (2006b) A multi-objective genetic algorithm with controllable convergence on knee regions. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC'06), pp 1916–1923.

Rachmawati L, Srinivasan D (2009) Multiobjective evolutionary algorithm with controllable focus on the knees of the Pareto front. IEEE Transactions on Evolutionary Computation, 13(4):810–824.

Rachmawati L, Srinivasan D (2010) Incorporating the notion of relative importance of objectives in evolutionary multiobjective optimization. IEEE Transactions on Evolutionary Computation, 14(4):530–546.

Ramesh R, Karwan M H, Zionts S (1988) Theory of convex cones in multicriteria decision making. Annals of Operations Research, 16(1):131–147.

Reyes-Sierra M, Coello Coello C A (2006) Multi-objective particle swarm optimizers: A survey of the state-of-the-art. International Journal of Computational Intelligence Research, 2(3):287–308.

Richardson J T, Palmer M R, Liepins G, Hilliard M (1989) Some guidelines for genetic algorithms with penalty functions. In: Proceedings of the 3rd International Conference on Genetic Algorithms (ICGA'89), pp 191–197.

Roy B (1990) The outranking approach and the foundations of ELECTRE methods. In: Bana e Costa C A (editor), Reading in multiple criteria decision aid, pp 155–183.

Roy B (1996) Multicriteria methodology for decision aiding, Kluwer Academic Publishers, Dordrecht, Netherlands.

Saint S, Lawson J R (1994) Rules for reaching consensus: a modern approach to decision making. Jossey-Bass, San Francisco, CA, USA.

Schaffer J D (1985) Multiple objective optimization with vector evaluated genetic algorithms. In: Proceedings of the 1st International Conference on Genetic Algorithms (ICGA'85), pp 93–100.

Schütze O, Laumanns M, Coello Coello C A (2008) Approximating the knee of an MOP with stochastic search algorithms. In: Proceedings of the 10th international conference on Parallel Problem Solving from Nature (PPSN X), pp 795–804.

Sharma D, Kumar A, Deb K, Sindhya K (2007) Hybridization of SBX based NSGA-II and sequential quadratic programming for solving multiobjective optimization problems. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC'07), pp 25–28.

Shawe-Taylor J, Cristianini N (2004) Kernel methods for pattern analysis. Cambridge University Press, New York, USA.

Shukla P K (2007) On Gradient-based local search methods in unconstrained evolutionary multi-objective optimization. In: Proceedings of the 4th international conference on Evolutionary Multicriterion Optimization (EMO'07), pp 569–583.

Smedley R, Wiseman (2001) Introducing pure mathematics. Oxford University Press, Oxford, UK.

Spall J C (1998) Implementation of the simultaneous perturbation algorithm for stochastic optimization. IEEE Transactions on Aerospace and Electronic Systems, 34(3):817–823.

Srinivas N, Deb K (1994) Multiobjective optimization using nondominated sorting in genetic algorithms. Evolutionary Computation, 2(3):221–248.

Stein M, Branke J, Schmeck H (2008) Efficient implementation of an active set algorithm for large-scale portfolio selection. Computers & Operations Research, 35(12):3945–3961.

Streichert F, Tanaka-Yamawaki M (2006) The effect of local search on the constrained portfolio selection problem. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC'06), pp 2368–2374.

Szczepanski M, Wierzbicki A P (2003) Application of multiple criterion evolutionary algorithms to vector optimization, decision support and reference point approaches. Journal of Telecommunications and Information Technology, 3(1):16–33.

Tan K C, Khor E F, Lee T H (2003) An evolutionary algorithm with advanced goal and priority specification for multi-objective optimization. Journal of Artificial Intelligence Research, 18(1):183–215.

Tan K C, Lee T H, Khor E F (1999) Evolutionary algorithms with goal and priority information for multi-objective optimization. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC'99), pp 106–113.

Tan K C, Lee T H, Khor E F (2001) Evolutionary algorithm with dynamic population size and local exploration for multiobjective optimization. IEEE Transactions on Evolutionary Computation, 5(6):565–588.

Tiwari S, Fadel G, Koch P, Deb K (2009) Performance assessment of the hybrid archive-based micro genetic algorithm (AMGA). In: Proceedings of IEEE Congress on Evolutionary Computation (CEC'09), pp 1935–1942.

Thiele L, Miettinen K, Korhonen P, Molina J (2009) A preference-based evolutionary algorithm for multiobjective optimization. Evolutionary Computation, 17(3):411–436.

Van Veldhuizen D A, Lamont G B (2000) On measuring multiobjective evolutionary algorithm performance. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC'00), pp 204–211.

Wagner T, Trautmann H (2010) Integration of preferences in hypervolume-based multiobjective evolutionary algorithms by means of desirability functions. IEEE Transactions on Evolutionary Computation, 14(5):688-701.

Wickramasinghe W R M U K, Li X (2008) Integrating user preferences with particle swarms for multi-objective optimization. In: Proceedings of the 10th Genetic and Evolutionary Computation COnference (GECCO'08), pp 745–752.

Wierzbicki A P (1980) The use of reference objectives in multiobjective optimization. In: Fandel G and Gal T (editors), Multiple criteria decision making theory and applications, pp 468–486.

Wilson R B (1963) A simplicial algorithm for concave programming. PhD thesis, Graduate School of Business Administration, Harvard University, USA.

Wooldridge M, Jennings N R (1994) Agent theories, architectures, and languages: A survey. ECAI Workshop on agent theories, architectures, and languages, pp 1–39.

Xia L, Conitzer V (2008) A sufficient condition for voting rules to be frequently manipulable. In: Proceedings of the 8th of ACM conference on Electronic Commerce (EC'08), pp 99–108.

Zhang Q, Li H (2007) MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE Transactions on Evolutionary Computation, 11(6):712–731.

Zhang Q, Liu W, Tsang E, Virginas B (2010) Expensive multiobjective optimization by MOEA/D with Gaussian process model. IEEE Transactions on Evolutionary Computation, 14(3):456–474.

Zitzler E (1999) Evolutionary algorithms for multiobjective optimization: Methods and applications. PhD thesis, ETH Zurich, Switzerland.

Zitzler E, Brockhoff D, Thiele L (2007) The hypervolume indicator revisited: On the design of Pareto-compliant indicators via weighted integration. In: Proceedings of the 4th international conference on Evolutionary Multi-criterion Optimization (EMO'07), pp 862–876.

Zitzler E, Deb K, Thiele L (2000) Comparison of multiobjective evolutionary algorithms: Empirical results. Evolutionary Computation, 8(2):173–195.

Zitzler E, Künzli S (2004) Indicator-based selection in multiobjective search. In: Proceedings of the 8th international conference on Parallel Problem Solving from Nature (PPSN VIII), pp 832–842.

Zitzler E, Laumanns M, Thiele L (2001) SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. In: Proceedings of Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems (EUROGEN), pp 95–100.

Zitzler E, Thiele L (1999) Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. IEEE Transactions on Evolutionary Computation, 3(4):257–271.

Zitzler E, Thiele L, Laumanns M, Fonseca C M, Grunert da Fonseca G (2003) Performance assessment of multiobjective optimizers: An analysis and review. IEEE Transactions on Evolutionary Computation, 7(2):117–132.

# Appendix A
# ZDT and DTLZ Benchmark Suites

## A.1 Zitzler-Deb-Thiele (ZDT) benchmark suite

The following table gives the definitions of the ZDT test functions (Zitzler et al. 2000):

**Table A.1 ZDT benchmark suite.**

| Name | Description | Pareto front |
|---|---|---|
| ZDT1 | $f_1(x) = x_1$<br>$f_2(x) = g(x)\left(1-\sqrt{\frac{f_1(x)}{g(x)}}\right)$<br>$g(x) = 1+\frac{9}{n-1}\sum_{i=2}^{n} x_i$<br>$x_i \in [0,1] \quad \forall i \in \{1,...,n\}, n = 30$ | The Pareto front corresponds to $g(x) = 1.$ |
| ZDT2 | $f_1(x) = x_1$<br>$f_2(x) = g(x)\left(1-\left(\frac{f_1(x)}{g(x)}\right)^2\right)$<br>$g(x) = 1+\frac{9}{n-1}\sum_{i=2}^{n} x_i$<br>$x_i \in [0,1] \quad \forall i \in \{1,...,n\}, n = 30$ | The Pareto corresponds to $g(x) = 1.$ |
| ZDT3 | $f_1(x) = x_1$<br>$f_2(x) = g(x)\left(1-\sqrt{\frac{f_1(x)}{g(x)}}-\frac{f_1(x)}{g(x)}\sin(10\pi\ f_1(x))\right)$<br>$g(x) = 1+\frac{9}{n-1}\sum_{i=2}^{n} x_i$<br>$x_i \in [0,1] \quad \forall i \in \{1,...,n\}, n = 30$ | The Pareto corresponds to $g(x) = 1.$ |
| ZDT4 | $f_1(x) = x_1$<br>$f_2(x) = g(x)\left(1-\sqrt{\frac{f_1(x)}{g(x)}}\right)$<br>$g(x) = 1+10(n-1)+\sum_{i=2}^{n}\left(x_i^2-10\cos(4\pi x_i)\right)$<br>$x_1 \in [0,1], \quad x_i \in [-5,5] \quad \forall i \in \{2,...,n\}, n = 30$ | The Pareto front corresponds to $g(x) = 1.$ |

| | | |
|---|---|---|
| ZDT5 | $f_1(x_1) = 1 + u(x_1)$<br>$f_2(x_2) = g(x)\left(\frac{1}{f_1(x)}\right)$<br>$g(x) = \sum_{i=2}^{n} v(u(x_i))$<br>$u(x_i)$ gives the number of ones in the bit vector $x_i$<br>$v(u(x_i)) = \begin{cases} 2 + u(x_i) & if\ \ u(x_i) < 5 \\ 1 & if\ \ u(x_i) = 5 \end{cases}$<br>$n = 11, x_1 \in \{0,1\}^{30}, x_i \in \{0,1\}^5\ \ \forall i \in \{2,...,n\}$ | The Pareto front corresponds to $g(x) = 10$.<br>The best deceptive local front corresponds to $g(x) = 11$. |
| ZDT6 | $f_1(x) = 1 - \exp(-4x_1)\sin^6(6\pi x_1)$<br>$f_2(x) = g(x)\left(1 - \left(\frac{f_1(x)}{g(x)}\right)^2\right)$<br>$g(x) = 1 + 9\left(\frac{1}{9}\sum_{i=2}^{n} x_i\right)^{0.25}$<br>$x_i \in [0,1]\ \ \forall i \in \{1,...,n\}, n = 30$ | The Pareto front corresponds to $g(x) = 1$. |

# A.2 Deb-Thiele-Laumans-Zitzler (DTLZ) benchmark suite

The following table gives the definitions of the DTLZ test functions (Deb et al. 2002b):

**Table A.2 DTLZ benchmark suite.**

| Name | Description | Pareto front |
|---|---|---|
| DTLZ1 | $f_1(x) = \frac{1}{2}x_1x_2...x_{M-1}(1 + g(X_M))$<br>$f_2(x) = \frac{1}{2}x_1x_2...(1 - x_{M-1})(1 + g(X_M))$<br>$\vdots$<br>$f_{M-1}(x) = \frac{1}{2}x_1(1 - x_2)(1 + g(X_M))$<br>$f_M(x) = \frac{1}{2}(1 - x_1)(1 + g(X_M))$<br>$g(X_M) = 100\left[\lvert X_M\rvert + \sum_{x_i \in X_M}(x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5))\right]$<br>$k = 5, n = M + k - 1, x_i \in [0,1]\ \ \forall i \in \{1,...,n\},$<br>$X_M = (x_M, x_{M+1},..., x_n)$ | The Pareto front corresponds to $X_M = 0$ and $\sum_{m=1}^{M} f_m = 0.5$. |

| | | |
|---|---|---|
| DTLZ2 | $f_1(x) = (1+g(X_M))\cos(x_1\pi/2)...\cos(x_{M-2}\pi/2)\cos(x_{M-1}\pi/2)$<br>$f_2(x) = (1+g(X_M))\cos(x_1\pi/2)...\cos(x_{M-2}\pi/2)\sin(x_{M-1}\pi/2)$<br>$\vdots$<br>$f_{M-1}(x) = (1+g(X_M))\cos(x_1\pi/2)\sin(x_2\pi/2)$<br>$f_M(x) = (1+g(X_M))\sin(x_1\pi/2)$<br>$g(X_M) = \sum_{x_i \in X_M} (x_i - 0.5)^2$<br>$k = 10, n = M + k - 1, x_i \in [0,1] \quad \forall i \in \{1,...,n\},$<br>$X_M = (x_M, x_{M+1},..., x_n)$ | The Pareto front corresponds to $x_i = 0.5$ $\forall x_i \in X_M$ and $\sum_{m=1}^{M} (f_m(x))^2 = 1.$ |
| DTLZ3 | Same as DTLZ2 except for a new $g$ function:<br>$g(X_M) = 100\left[\lvert X_M\rvert + \sum_{x_i \in X_M} (x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5))\right]$ | The Pareto front corresponds to $x_i = 0.5$ $\forall x_i \in X_M$. |
| DTLZ4 | $f_1(x) = (1+g(X_M))\cos(x_1^\alpha\pi/2)...\cos(x_{M-2}^\alpha\pi/2)\cos(x_{M-1}^\alpha\pi/2)$<br>$f_2(x) = (1+g(X_M))\cos(x_1^\alpha\pi/2)...\cos(x_{M-2}^\alpha\pi/2)\sin(x_{M-1}^\alpha\pi/2)$<br>$\vdots$<br>$f_{M-1}(x) = (1+g(X_M))\cos(x_1^\alpha\pi/2)\sin(x_2^\alpha\pi/2)$<br>$f_M(x) = (1+g(X_M))\sin(x_1^\alpha\pi/2)$<br>$g(X_M) = \sum_{x_i \in X_M} (x_i - 0.5)^2$<br>$k = 10, n = M + k - 1, x_i \in [0,1] \quad \forall i \in \{1,...,n\}, \alpha = 100$<br>$X_M = (x_M, x_{M+1},..., x_n)$ | The Pareto front corresponds to $x_i = 0.5$ $\forall x_i \in X_M$. |
| DTLZ5 | Same as DTLZ2 where $x_i$ is replaced by<br>$\theta_i = \frac{\pi}{4(1+g(r))}(1+2g(r)x_i)$ | The Pareto front corresponds to $x_i = 0.5$ $\forall x_i \in X_M$. |
| DTLZ6 | Same as DTLZ5 except for new $g$ function:<br>$g(X_M) = \sum_{x_i \in X_M} x_i^{0.1}$ | The Pareto front corresponds to $x_i = 0.5$ $\forall x_i \in X_M$. |
| DTLZ7 | $f_1(x) = x_1$<br>$f_2(x) = x_2$<br>$\vdots$<br>$f_{M-1}(x) = x_{M-1}$<br>$f_M(x) = (1+g(X_M)) + h(f_1, f_2,..., f_{M-1}, g)$<br>$g(X_M) = 1 + \frac{9}{\lvert X_M\rvert}\sum_{x_i \in X_M} x_i$<br>$h(f_1, f_2,..., f_{M-1}, g) = M - \sum_{m=1}^{M-1}\left[\frac{f_m}{1+g}(1+\sin(3\pi f_m))\right]$<br>$k = 20, n = M + k - 1, x_i \in [0,1] \quad \forall i \in \{1,...,n\},$<br>$X_M = (x_M, x_{M+1},..., x_n)$ | The Pareto front corresponds to $X_M = 0.$ |

| | | |
|---|---|---|
| DTLZ8 | $f_m(x) = \frac{1}{\lfloor \frac{n}{M} \rfloor} \sum_{i=\lfloor (j-1)\frac{n}{M} \rfloor}^{\lfloor \frac{n}{M} \rfloor} x_i \quad \forall j \in \{1,...,M\}$<br>Subject to $g_m(x) = f_M(x) + 4 f_m(x) - 1 \geq 0 \quad \forall j \in \{1,...,M-1\}$<br>$g_M(x) = 2 f_M(x) + \min_{\substack{i,m=1 \\ i \neq m}}^{M-1} [f_i(x) + f_m(x)] \geq 0$<br>$x_i \in [0,1] \quad \forall i \in \{1,...,n\}, \; n = 10M$ | The Pareto front is a combination of a straight line and a hyper-plane. The straight line is the intersection of the first (*M-1*) constraints with $f_1 = f_2 = ... = f_{M-1}$ and the hyper-plane is represented by the constraint $g_M$. |
| DTLZ9 | $f_m(x) = \frac{1}{\lfloor \frac{n}{M} \rfloor} \sum_{i=\lfloor (j-1)\frac{n}{M} \rfloor}^{\lfloor \frac{n}{M} \rfloor} x_i \quad \forall j \in \{1,...,M\}$<br>Subject to $g_m(x) = f_M^2(x) + f_m^2(x) - 1 \geq 0 \quad \forall j \in \{1,...,M-1\}$<br>$x_i \in [0,1] \quad \forall i \in \{1,...,n\}, \; n = 10M$ | The Pareto front is a curve with $f_1 = f_2 = ... = f_{M-1}$. |

# Appendix B
# Knee-based Test Functions

## B.1 Knee-based test problems

The following table gives the definitions of the knee-based test problems used throughout this thesis (Branke et al. 2001, Rachmawati and Srinivasan 2009):

**Table B.1 Knee-based test problems.**

| Problem name | Problem definition |
|---|---|
| DO2DK | $f_1(x) = g(x)\,r(x)\left[\sin\left(\frac{\pi x_1}{2^{s+1}}\right)+\pi\left(1+\frac{2^s-1}{2^{s+2}}\right)\right]$ ; $f_2(x) = g(x)\,r(x)\left[\cos\left(\frac{\pi x_1}{2}+\pi\right)+1\right]$<br>$g(x) = 1+\frac{9}{n-1}\sum_{i=2}^{n} x_i$ ; $r(x) = 5+10(x_1-0.5)^2+\frac{2^{s/2}}{K}\cos(2K\pi x_1)$<br>$x_i \in [0,1], \forall\, i = 1,\ldots,30$ |
| DO2DK-1 | $f_1(x) = g(x)\,r(x)\left[\sin\left(\frac{\pi x_1}{2^{s+1}}\right)+\pi\left(1+\frac{2^s-1}{2^{s+2}}\right)\right]$ ; $f_2(x) = g(x)\,r(x)\left[\cos\left(\frac{\pi x_1}{2}+\pi\right)+1\right]$<br>$g(x) = 1+\sum_{i=2}^{n} x_i^{0.1}$ ; $r(x) = 5+10(x_1-0.5)^2+\frac{2^{s/2}}{K}\cos(2K\pi x_1)$<br>$x_i \in [0,1], \forall\, i = 1,\ldots,30$ |
| DEB2DK | $f_1(x) = g(x)\,r(x)\sin\left(\frac{\pi x_1}{2}\right)$ ; $f_2(x) = g(x)\,r(x)\cos\left(\frac{\pi x_1}{2}\right)$<br>$g(x) = 1+\frac{9}{n-1}\sum_{i=2}^{n} x_i$ ; $r(x) = 5+10(x_1-0.5)^2+\frac{1}{K}\cos(2K\pi x_1)$<br>$x_i \in [0,1], \forall\, i = 1,\ldots,30$ |
| DEB2DK-1 | $f_1(x) = g(x)\,r(x)\sin\left(\frac{\pi x_1}{2}\right)$ ; $f_2(x) = g(x)\,r(x)\cos\left(\frac{\pi x_1}{2}\right)$<br>$g(x) = 1+\sum_{i=2}^{n} x_i^{0.1}$ ; $r(x) = 5+10(x_1-0.5)^2+\frac{1}{K}\cos(2K\pi x_1)$<br>$x_i \in [0,1], \forall\, i = 1,\ldots,30$ |
| DEB2DK-2 | $f_1(x) = g(x)\,r(x)\sin^3\left(\frac{\pi x_1}{2}\right)$ ; $f_2(x) = g(x)\,r(x)\cos\left(\frac{\pi x_1}{2}\right)$<br>$g(x) = 1+\frac{9}{n-1}\sum_{i=2}^{n} x_i$ ; $r(x) = 2.5+10(x_1-0.5)^2+\frac{1}{K}\cos(2K\pi x_1)$<br>$x_i \in [0,1], \forall\, i = 1,\ldots,30$ |

| | |
|---|---|
| DEB3DK | $f_1(x) = g(x)\,r(x)\sin\left(\frac{\pi x_1}{2}\right)\sin\left(\frac{\pi x_2}{2}\right)$ ; $f_2(x) = g(x)\,r(x)\sin\left(\frac{\pi x_1}{2}\right)\cos\left(\frac{\pi x_2}{2}\right)$ ;<br>$f_3(x) = g(x)\,r(x)\cos\left(\frac{\pi x_1}{2}\right)$ ; $g(x) = 1+\frac{9}{n-1}\sum_{i=2}^{n} x_i$ ; $r(x) = \frac{r_1(x_1)+r_2(x_2)}{2}$ ;<br>$r_i(x_i) = 5+10\left(x_i-0.5\right)^2+\frac{2}{K}\cos(2K\,\pi\,x_i)$<br>$x_i \in [0,1], \forall\, i = 1,\ldots,12$ |
| DEB3DK-1 | $f_1(x) = g(x)\,r(x)\sin\left(\frac{\pi x_1}{2}\right)\sin\left(\frac{\pi x_2}{2}\right)$ ; $f_2(x) = g(x)\,r(x)\sin\left(\frac{\pi x_1}{2}\right)\cos\left(\frac{\pi x_2}{2}\right)$ ;<br>$f_3(x) = g(x)\,r(x)\cos\left(\frac{\pi x_1}{2}\right)$ ; $g(x) = 1+\sum_{i=2}^{n} x_i^{0.1}$ ; $r(x) = \frac{r_1(x_1)+r_2(x_2)}{2}$ ;<br>$r_i(x_i) = 5+10\left(x_i-0.5\right)^2+\frac{2}{K}\cos(2K\,\pi\,x_i)$<br>$x_i \in [0,1], \forall\, i = 1,\ldots,12$ |