UNIVERSIDADE FEDERAL FLUMINENSE

DANIEL LOPES CINALLI

Integrating Collective Intelligence into Multi-Objective Optimization Evolutionary Algorithms: Interactive Preferences and Reference Points for a Facility Location Problem

NITERÓI

2017

UNIVERSIDADE FEDERAL FLUMINENSE

DANIEL LOPES CINALLI

Integrating Collective Intelligence into Multi-Objective Optimization Evolutionary Algorithms: Interactive Preferences and Reference Points for a Facility Location Problem

Defesa de Proposta de Tese de Doutorado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Doutor em Computação. Área de concentração: Inteligência Arificial

Orientador:
Profa. Dr. Ana Cristina Bicharra Garcia

Co-orientador:
Prof. Dr. Luis Martí Orosa

NITERÓI

2017

to Raphaela, Murilo and Caio.

Acknowledgments

to be written...

Abstract

Some organizations have to assign and manage facilities in an optimized way. Those activities involve many stakeholders with multiple conflicting objectives. Multi-objective optimization evolutionary algorithms have been successfully applied to several complex synthetic and real-world multi-objective problems (MOPs). Although these algorithms have proved themselves as a valid approach to the MOP, there is still need for improvements on the performance of the search process. This work introduces a novel approach meant for bringing collective intelligence methods into the optimization process carried out by evolutionary multi-objective optimization algorithms. In particular, it describes the extension of some well-known algorithms (Non-dominated Sorting Genetic Algorithm-II, S-metric Selection Evolutionary Multi-objective Algorithm, Strength Pareto Evolutionary Algorithm 2) to include collective online preferences into the optimization process. In these new methods —called CI-NSGA-II, CI-SMS-EMOA and CI-SPEA2—, groups of decision makers can highlight the regions of the Pareto frontier that are more relevant to them as to focus the search process mainly on those areas. Additionally, the integration of interactivity and cooperation into the evolutionary algorithms refines users' preferences and improves the reference points throughout the evolutionary progress. Rather than a unique or small group of decision makers with unilateral preferences, the application of dynamic group preferences aggregates consistent collective reference points and creative solutions to enhance multi-objective results. In order to analyse the results, three new performance indicators based on preferences are introduced to evaluate the quality of approximations set. As part of this work, the algorithms' performances are tested when faced with some synthetic problems as well as a real-world case of facility location. The experiments demonstrate the advantages of a collective intelligence operator integrated into the multi-objective evolutionary algorithm.

Keywords: collective intelligence; preferences; reference points; evolutionary multi-objective optimization algorithms; facility location problem.

Resumo

Certas organizações têm a necessidade de gerenciar e distribuir recursos de maneira otimizada. Essas atividades, em geral, envolvem muitos *stakeholders* com múltiplos objetivos conflitantes. A otimização evolutiva multi-objetivo foi aplicada com sucesso em diversos problemas complexos multi-objetivo (MOP) do mundo real. Embora esses algoritmos tenham uma abordagem válida para resolução de MOP, ainda existe necessidade de melhorar a performance do processo de busca. Este trabalho introduz um novo método que visa integrar inteligência coletiva ao processo de otimização conduzido pelos algoritmos evolucionários multi-objetivo. Em particular, a pesquisa descreve a extensão de alguns algoritmos clássicos (*Non-dominated Sorting Genetic Algorithm-II, S-metric Selection Evolutionary Multi-objective Algorithm, Strength Pareto Evolutionary Algorithm 2*) para a inclusão de preferências coletivas no processo de otimização em tempo real. Nesses métodos —chamados CI-NSGA-II, CI-SMS-EMOA e CI-SPEA2—, os grupos de tomadores de decisão podem ressaltar as regiões da frente de Pareto que sejam mais relevantes a eles e, por conseguinte, focar o processo de busca principalmente nessas áreas. Além disso, a combinação de interatividade e cooperação nos algoritmos evolucionários refinam as preferências dos usuários e aperfeiçoam os pontos de referência durante o progresso evolutivo. Em lugar de um único ou pequeno grupo de tomadores de decisão com preferências unilaterais, a aplicação dinâmica de preferências agregam pontos de referência consistentes e soluções criativas para melhorar os resultados multi-objetivos. A fim de analisar os resultados, três novos indicadores de performance baseados em preferências são apresentados para avaliar a qualidade das frents de aproximação. Como parte desse trabalho, as performances dos algoritmos são testadas com problemas de *benchmark* e com um problema de alocação de recursos. Os experimentos demonstram as vantagens de um operador de inteligência coletiva integrado aos algoritmos evolucionários multi-objetivo.

Palavras-chave: inteligência coletiva; preferências; pontos de referência; algoritmos evolutivos multi-objetivo; problema de alocação de recursos.

List of Figures

List of Tables

List of Abbreviations

AOF	:	Aggregate Objective Function;
ASF	:	Achievement Scalarizing Function;
BIC	:	Bayesian Information Criterion;
$C2_R$	:	Reference Set Distance;
CF	:	Composite Front;
CFLP	:	Capacitated Facility Location Problem;
CH-MOEA	:	Convex Hull Multi-objective Evolutionary Algorithm;
CH-MOGP	:	Convex Hull Multi-objective Genetic Programming;
CHIM	:	Convex Hull of Individual Minima;
COIN	:	Collective Intelligence;
DEAP	:	Distributed Evolutionary Algorithms in Python;
DM	:	Decision Maker;
EA	:	Evolutionary Algorithms;
EC	:	Evolutionary Computation;
EF	:	Efficient Front;
EM	:	Expectation Maximization;
ER	:	Error Ratio;
FDP	:	Field Development Planning;
FLP	:	Facility Location Problem;
FPSO	:	Floating Production Storage Offloading;
GA	:	Genetic Algorithm:
GD	:	Generation Distance;
GM	:	Gaussian Mixture;
HV	:	Hypervolume;
IEA	:	International Energy Agency;
IGA	:	Interactive Genetic Algorithms;
IGD	:	Inverted Generational Distance;
IT	:	Information Technology;
KM	:	K-means;

MCDM	:	Multi-Criterion Decision-Making;
MOEA	:	Multi-Objective Evolutionary Algorithms;
MOP	:	Multi-Objective Optimization Problems;
MPFE	:	Maximum Pareto Front Error;
MVT	:	Model-View-Template;
NA	:	Not Applicable;
NASA	:	National Aeronautics and Space Administration;
NBI	:	Normal Boundary Intersection;
NSGA-II	:	Non-dominated Sorting Genetic Algorithm II;
ONVG	:	Overall Non-dominated Vector Generation;
ONVGR	:	Overall Non-dominated Vector Generation Ratio;
OR	:	Operations Research;
PAAS	:	Platform as a Service;
REST	:	Representational State Transfer;
R-NSGA-II	:	Reference-Point-Based NSGA-II;
RNI	:	Ratio of Non-dominated Individuals;
ROI	:	Region of Interest;
SOA	:	Service-Oriented Architecture;
SPEA2	:	Strength Pareto Evolutionary Algorithm;
SMS-EMOA	:	S-metric selection EMOA;
UD	:	Uniform Distribution;
UFLP	:	Uncapacitated Facility Location Problem;
UPCF	:	User-Preference Composite Front;

Contents

Chapter 1

Introduction

1.1 Multi-Objective Problem

Many real-life decision problems require dealing with trade-offs between multiple conflicting objectives. Those problems are known as multi-objective optimization problems (MOPs) as they consider more than one criteria to be simultaneously optimized [73]. Frequently, the solution of a MOP is not a single point that optimizes all the objectives at the same time, but, instead, a possibly infinite set of points that represent different trade-offs between the objectives known as Pareto-optimal set (P_S). A decision maker (DM) has to select which of those solutions are the ones to be carried out in practice according to some *a priori* high-level preferences.

Several examples come from everyday life and permeate different areas like engineering, financial, logistics, personal routine, among others. Figure 1.1 illustrates one situation of decision-making involving conflicting objectives. It refers to the process of purchasing an automobile car [60]. The buyer must decide if he/she should pick the most comfort and expensive car, or the cheapest and less comfort vehicle. Actually, this decision process is not a single objective one. The buyer still can choose cars between these extreme solutions where a trade-off between these two criteria exists (points A, B and C). The DM will prioritize the objective cost and sacrifice the comfort, or vice-versa, according to his/her personal preferences.

When the National Aeronautics and Space Administration (NASA) began to study its post-Apollo human spaceflight programs in the seventies, these projects had to debate over the optimal shuttle design that best balanced operational cost, development of technology and capability. Facing a budget constraint at the time, NASA planners needed to decide over many shuttle orbiters and others system components characteristics to

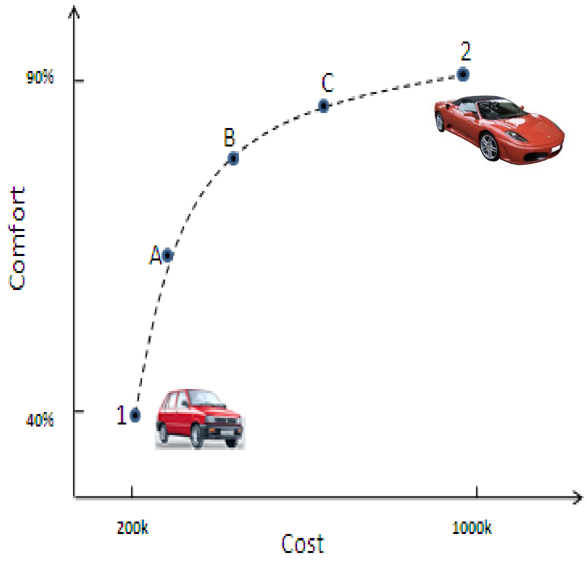


Figure 1.1: Multi-objective situation in real-life. It considers the cost and the comfort objectives to exemplify a trade-off set in a process of purchasing a car. Figure taken from [60].

lower the development costs of the resulting designs [115]: propulsion system, glide or power-assisted landing, primary structural material, extent of cross-range manoeuvrability, safety, aerodynamic heating and pressure. The designers, for example, had to find the optimum distribution of heat resistant materials based on the angle of atmospheric re-entry. If the angle is too steep, there is more friction on the nose and wing leading edges of the shuttle. If the angle is too shallow, the surface temperature will be concentrated on the spacecraft's underside. The designers and engineers wanted to decrease the angle in order to install the heat shield on the spacecraft's underside. But, at the same time, they wanted to preserve the cold as much as possible in that area to avoid premature deterioration of the heat shield. That conflicting situation made the entities involved disagree in many strategies to locate the required heat protection and choose the right angle to re-entry. Figure 1.2 shows the temperature distribution on a space shuttle according to the angle of atmospheric re-entry.

1.2 Facility Location Problem

The facility location problem [54, 106] is a branch of operations research concerned with the assignment of available facilities and resources to achieve the organization's strategic goals. This area has received significant attention due to the number of endeavours that must reduce costs and optimize their operations: manufacturing plants, storage facilities, public transport planning, equipment for oil spills, warehouses, vehicle routing,

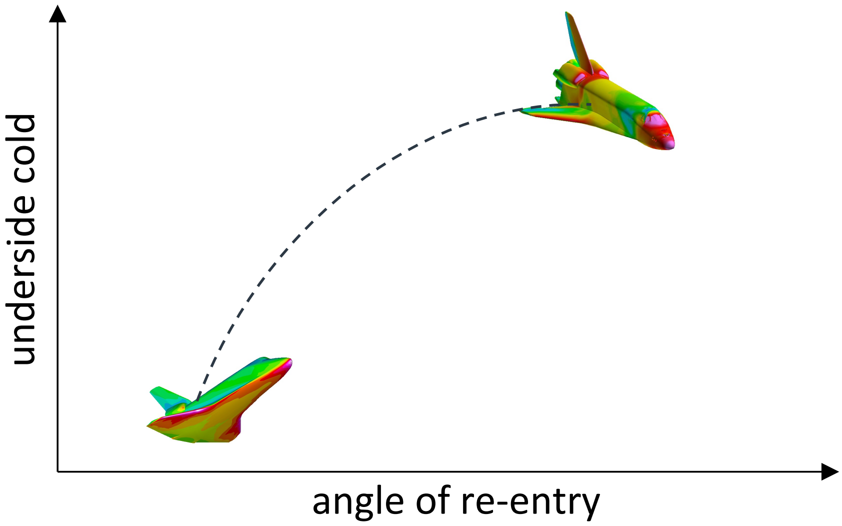


Figure 1.2: This figure shows the temperature distribution on a space shuttle during upper atmosphere re-entry. If the angle is too steep, there is more friction on the nose and wing leading edges of the shuttle (red); too shallow, and the surface temperature will be concentrated on the spacecraft's underside. The designers wanted to decrease the angle in order to keep the heat resistant materials on the spacecraft's underside. But, at the same time, they wanted to preserve the cold as much as possible on that area to avoid premature deterioration of the heat shield.

fire stations, hospitals, etc.

In this context, the petroleum industry manages an optimal placement and interconnection of extraction and transportation equipment to increase extraction, pumping, and generation of oil, while keeping costs and robustness at optimal levels. Offshore plant operation must balance the conflicting needs for materials and the application of machinery in an economic way to maximize the different aspects related to operational effectiveness. These circumstances describe a multi-objective optimization and decision making problem with many stakeholders looking for efficient approaches.

Figure 1.3 shows a fleet of specialized vessels for pipeline installation and subsea construction. Different decision targets can be considered in this resource placement problem: the transportation requirements and restrictions; the cost; the production; the safety of the environment and workers; the robustness of the production chain; the quality of the industrial assets and etc. But as more objectives compete among each other, the problem becomes harder to solve and comprehend.

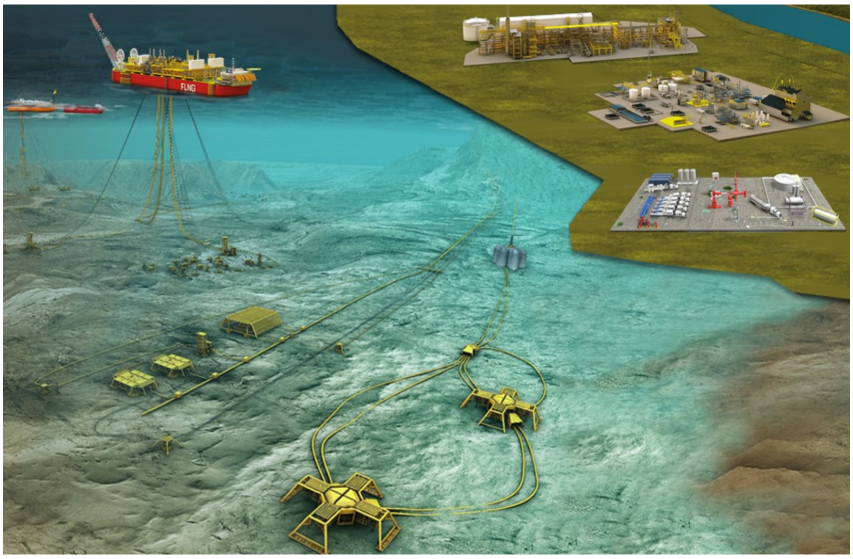


Figure 1.3: Multiple business activities of an oil & gas industry. The picture shows a fleet of specialized vessels for pipeline installation and subsea construction. Also, it illustrates offshore and onshore infrastructures responsible for processing the collect oil or gas [3].

1.3 Multi-Objective Optimization Methods

Multi-objective optimization refers to the process of finding a set of feasible solutions to a problem by trading off the optimal values of two or more functions. MOPs have been addressed with a wide variety of methods. The techniques are classified into two broad categories [50]: deterministic and heuristic.

Deterministic methods have been successfully applied in optimization. However, some problem characteristics may reduce the algorithm effectiveness. Many of the deterministic approaches that are based on gradient information of the parameters have difficulties with the local optima, plateaus or ridges in the search landscape. Gradient-based methods use the derivatives of the objective function to guide the search. They can perform well on unimodal functions and quickly converge to an optimal solution, but are not efficient in multimodal functions, non-differentiable or discontinuous problems. Branch and bound search techniques need heuristics to limit the search space. Depth-first and breadth-first require too much computational time with large-sized problems. Another very common approach to solve MOPs constructs a single aggregate objective function (AOF) which combines all of the objectives.

The scalarization of the vector-valued problem converts the original multiple-objective problem into a single-objective optimization. Prominent examples of this technique are the weighted-sum approach, ϵ-constraint [124], weighted metrics, Benson's method [27], lexicographic, min-max, among others. All of these methods try to find the optimal Pareto front using different approximation techniques. Note that scalarizing approaches are feasible if there are only a few objectives [158], because the number of combinations of weighting coefficients for objectives grows exponentially with the number of objectives.

The approximation of the entire Pareto-optimal frontier requires extensive time and computational resource. In the general case, optimization problems and, hence, MOPs, are NP-hard [15]. Moreover, some of those algorithms are sensitive to the shape or continuity of the Pareto front. For that reason deterministic search techniques are usually unsuitable to handle the complexity of this task.

Metaheuristic and stochastic approaches are frequently the only viable alternative to handle MOPs. Among existing metaheuristics there can be mentioned the evolutionary computation, simulated annealing, tabu search, Monte Carlo, ant colony and memetic algorithms [169]. Evolutionary algorithm (EAs) [16] is a nature-inspired computational approach that uses stochastic operators (variation, evaluation and selection) to work on optimization problems. The application of evolutionary algorithms to MOPs has prompted the creation of multi-objective optimization evolutionary algorithms (MOEAs) [50]. EAs can find a finite population of optimal solutions in one iteration run and disregard any particular shape of the underlying fitness landscape. They approximate the optimal Pareto front as a discrete set of points.

1.4 Problem Description

As mentioned in the previous section, multi-objective problems can be hard or even impossible to solve exactly. Most classical and optimization methods find difficulties to reach a global perspective and often converge to a locally optimal solution having inferior objective function values. Although MOEA is a valid approach for multi-objective optimization, in complex cases the computation of the entire Pareto-optimal frontier is still a time-consuming and onerous process.

Some MOP characteristics further complicate the final step of the optimization process. In the end of the optimization, the set of all non-dominated solutions may contain infinite points. The decision maker (DM) must identify which of those solutions are the ones that satisfy her/his preferences and would be realized in practice. Most of the time, the DM is not interested in finding solutions covering the entire trade-off set, but only a small sub-set within relevant regions of her/his preference. This task can be rather complex and requires in-depth knowledge of the problem being solved, something that is impossible in many practical situations.

In problems with a large number of objective functions to be optimized, the number of non-dominated points used to represent the whole Pareto front (P_F) increases exponentially. The solutions obtained in these higher order spaces do not dominate one another and, therefore, the set of optimal solutions becomes the entire solution space. This is not only an issue for the MOEAs due to the computational resources necessary to generate the trade-off set, but it also causes severe difficulties for the DMs to understand the obtained solutions and then to make decisions.

DM's preferences can be expressed as reference points (Section 2.7). Interactive techniques and reference points can be used to mitigate those inconveniences and steer the search for a suitable resolution in preferred areas of P_F. The interlace of the search process and DM preferences improves the population quality throughout the evolutionary process and leads to compromise solutions of practical interest. These approaches allow the optimization algorithm to reduce the search area and thus reaching satisfactory solutions at a lower computational cost.

On the other hand, it remains challenging to identify *a-priori* reference points for an unknown problem and elicit preferences to the optimization process. Despite the different manners to articulate the preferences in MOEAs, the approximation of the Pareto-optimal frontier still needs too much time and computational resource to calculate a great number of function evaluations.

1.5 Integrating Collective Intelligence in MOEAs

In practice, optimization problems pose difficulties in defining *a priori* reference points or preferences. Because of the lack of expert knowledge on the problem at hand, the DM preferences can be biased, unilateral, incomplete or even nonexistent.

Collective intelligence (COIN) methods [117] put forward a paradigm that allows elucidating knowledge from groups of (not necessarily expert) individuals. In this regard, collective reference points obtained by the interaction and aggregation of multiple opinions can be used to produce an accurate and unbiased representation of preferences and, hence, reference points. Built upon the subjectivity of the crowds and human cognition, the intelligence of participatory actions addresses dynamic collective reference points to overcome MOPs difficulties and guide the exploration of preferred solutions.

The integration of collective intelligence in MOEAs replaces a single reference point in the search process by a global preference of all the participants involved in the optimization. The collective reference point has an advantage over traditional interactive approaches because their results are not driven by a single DM, but a group of people that delimits their collective area of interest and preferences in the objective space. The combination of COIN into evolutionary algorithms is the main contribution of this work. It addresses the problem of finding the optimal solutions with less function evaluation and focusing the search on relevant regions of the Pareto front at a lower computational cost.

Thus, this work describes how to implement a collective intelligence operator to bias the search during the optimization phase and restrict the objective space. Besides the four main operators based on the theory of evolution: selection, crossover, mutation and elite-preservation; the COIN-based variation operator receives rational collaborations from the participants to improve the overall quality of evolutionary population and positively affect further generations. The underlying idea here is to submit some of the MOEA candidate solutions to be modified and expect that they are improved by the collective. The modified solutions are then re-injected to the population and, therefore, go through the rest of the steps of the evolutionary process. This calls for a special problem rendering that allows members of the collective to interact with and modify the solutions.

The synergy of actions and the heterogeneity inside collective environments develop creative solutions based on the crowds' subjectivity and cognition. From a larger perspective, the COIN operator brings a subjective input to the optimization engine and gives a new collective intelligence component to work along the random operators from stochastic methods.

Facility location problems can be used as the application domain for the collective intelligence operators. Most of the location problems are connected to a real-world case example and their objectives are meaningful to human participants. Hence, those problems can interact with crowd's cognition to obtain optimal solutions for complex problems of facility location. The proposed interactive COIN-MOEAs (Section 4) embeds human characteristics (strategic thinking, 3D spatial reasoning and orientation) into the optimization process. Some potential benefits are noted especially in cases of problems with multi-objective environments and spatial requirements for task execution, such as: logistics problems, oil field development planning and routing.

1.6 Objectives and Contributions

This thesis seeks to solve the problem:

- the approximation of the Pareto-optimal frontier needs extensive time and computational resource to calculate a great number of function evaluations;

In order to address this research problem, the thesis:

1. brings a new useful connection between different fields: multi-objective evolutionary algorithms and collective intelligence;
2. applies collective intelligence as a new operator to compute a fewer number of evaluation of alternatives in the evolutionary process;
3. iteratively refines the search parameters and finds out a preferred sub-set of the Pareto-optimal front (relevant regions);
4. implements 3 new performance indicators to evaluate the outcome sets of the preference-based interactive algorithms;
5. obtains optimal solutions for complex multi-objective facility location problems with spatial orientation;

With regard to the firsts objectives, the thesis puts forward the integration of real-time collective preferences and interactive behaviour into three existing MOEAs: Non-dominated Sorting Genetic Algorithm-II (NSGA-II) [64], S-metric Selection Evolutionary Multi-objective Algorithm (SMS-EMOA) [28], Strength Pareto Evolutionary Algorithm 2 (SPEA2) [179]; and, therefore, introduces a collective intelligence version of them. The new algorithms overcome difficulties derived from choosing *a priori* reference points and propose an online approach to define preferences with the support of collective environments. They produce better solutions in the sense that they iteratively refine the search parameters and generate more appropriated points for DM final choice.

While executing this task it became evident the lack of adequate performance indicators that take into account preferences. Therefore, the work also introduces three new performance indicators that are used to evaluate the quality of the optimal frontier approximation driven by the online collective preferences.

With regard to the last objective, experiments with the facility location problem demonstrated the effectiveness of the proposed algorithms as they yield better solutions at a lower computational cost. Based on the results, the COIN operator is a competitive advantage as it decreases the number of required function evaluations in the optimization process and provides faster analysis of preferred alternatives only.

1.7 Structure of the Thesis

The rest of this document is organized as follows. Chapter 2 covers some required formal definitions of multi-objective optimization, collective intelligence field and clustering algorithms. It outlines different techniques of reference points and interactive MOEAs. Chapter 3 describes the application domain for the collective intelligence operators. Chapter 4 presents the hypothesis of COIN as a genetic operator to bias the optimization search. Subsequently, it proposes the new algorithms based on interactive and collective intelligence techniques. Chapter 5 introduces three new performance indicators and shows the more appropriate ones from the classical literature to evaluate the quality of points around the collective reference points. After that, Chapter 6 analyses the performance of the algorithms when faced with benchmark problems and a facility location case study. Finally, Chapter 7 puts forward some conclusive remarks and future work directions.

For reference purposes, a number of complementary appendixes are included in this document. Appendix A lists all the publications related to this work so far. Appendix B lists the submitted materials for publication. Appendix C presents the formula of the scalable multi-objective test problems used for experiments in this thesis. Appendix D presents some collective decision-making techniques to combine singular inputs into a global perspective for decision. The tools used as part of the system architecture are presented in more detail in Appendix E. Appendix F describes the Conover-Inman hypothesis test.

Chapter 2

Foundations

2.1 Multi-Objective Optimization

MOP can be stated as follows:

Definition 2.1 (Multi-Objective Optimization Problem)**.** A MOP can be stated as follows:

$$\left.\begin{aligned} \text{minimize } \boldsymbol{F}(\boldsymbol{x}) &= \{f_1(\boldsymbol{x}), \ldots, f_k(\boldsymbol{x})\}\,, \\ \text{subject to } g_i(\boldsymbol{x}) &\leq 0\,, h_j(\boldsymbol{x}) = 0\,. \end{aligned}\right\} \tag{2.1}$$

where $\boldsymbol{x} \in \Omega$ is an n-dimensional decision variable. The solution to this problem can be expressed by relying on the Pareto dominance relationship.

Definition 2.2 (Pareto dominance relation)**.** A $\boldsymbol{x}$ is said to dominate $\boldsymbol{v}$ (denoted as $\boldsymbol{x} \prec \boldsymbol{v}$) iff $\forall f_i$, $f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{v}) \wedge \exists f_j$ s.t. $f_j(\boldsymbol{x}) < f_j(\boldsymbol{v})$.

There are other forms of Pareto dominance and two objective vectors may be incomparable (ie, not dominated by each other):

Definition 2.3 (Weak Pareto dominance)**.** A $\boldsymbol{x}$ weakly dominates $\boldsymbol{v}$ ($\boldsymbol{x} \preccurlyeq \boldsymbol{v}$) iff $\forall f_i$, $f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{v})$.

Definition 2.4 (Strict Pareto dominance)**.** The strictly dominance $\boldsymbol{x} \prec\!\!\prec \boldsymbol{v}$ stands only on iff $\forall f_i$, $f_i(\boldsymbol{x}) < f_i(\boldsymbol{v})$.

Definition 2.5 (Incomparable solutions)**.** The decision variables can also be incomparable ($\boldsymbol{x} \parallel \boldsymbol{v}$) when $\neg\,(\boldsymbol{x} \preccurlyeq \boldsymbol{v}) \wedge \neg\,(\boldsymbol{v} \preccurlyeq \boldsymbol{x})$.

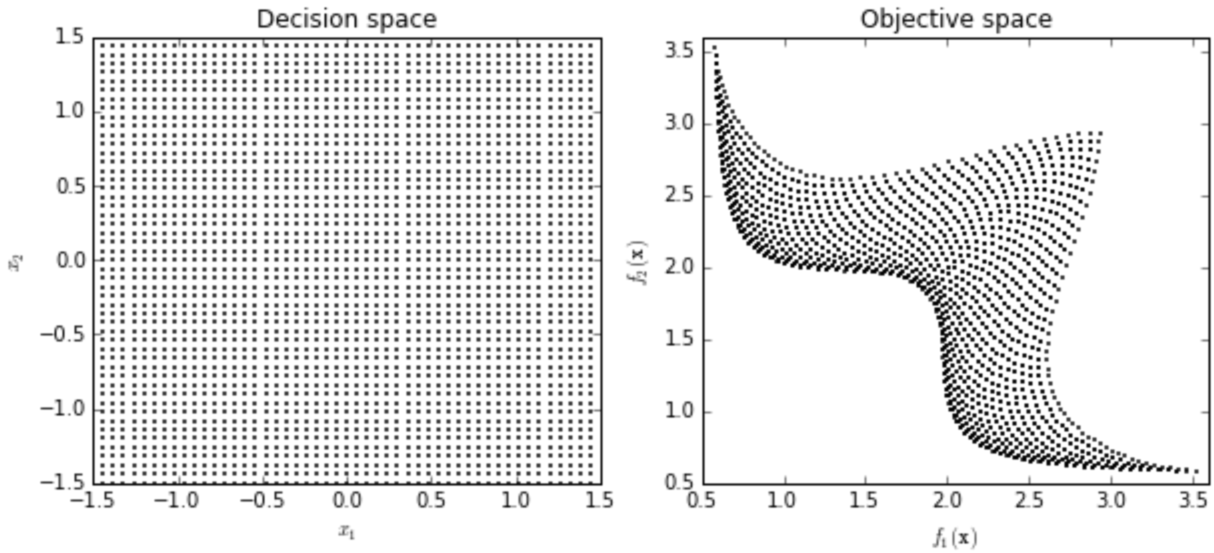


Figure 2.1: Decision and objective spaces in Dent problem. Figure taken from [120].

A solution $\boldsymbol{x} \in \Omega$ is Pareto optimal if there does not exist another solution $\boldsymbol{x'} \in \Omega$ such that $\boldsymbol{F}(\boldsymbol{x'}) \prec \boldsymbol{F}(\boldsymbol{x})$. Considering a set of points, the non-dominated subset can be constructed using the Pareto dominance relation:

Definition 2.6 (Pareto-optimal set)**.** The solution of a MOP is a (possibly infinite) Pareto-optimal set $P_S = \{\boldsymbol{x} \in \Omega, \nexists \boldsymbol{x'} \in \Omega \text{ such that } \boldsymbol{F}(\boldsymbol{x'}) \prec \boldsymbol{F}(\boldsymbol{x})\}$ that contains all the elements of Ω that are not Pareto-dominated ($\prec$) by any other element [51].

Definition 2.7 (Pareto-optimal front)**.** The projection of P_S through $\boldsymbol{F}()$ is known as the Pareto-optimal front, P_F.

The space $\mathbb{R}^k$ which contains the set of the attainable objective vectors $\mathcal{Z}$ is referred to as the objective space. The boundary of $\mathcal{Z}$ is denoted by $\partial\mathcal{Z}$. The mapping between decision space and objective space in a two dimensional minimization case is illustrated by the Dent problem [143] in Figure 2.1 [122]. This problem minimizes $\boldsymbol{F}(\boldsymbol{x}) = \{f_1(\boldsymbol{x}), f_2(\boldsymbol{x})\}$, such that $f_1(\boldsymbol{x}) = \frac{1}{2}\left(\sqrt{1+(x_1+x_2)^2}\sqrt{1+(x_1-x_2)^2}+x_1-x_2\right)+d$, $f_2(\boldsymbol{x}) = \frac{1}{2}\left(\sqrt{1+(x_1+x_2)^2}\sqrt{1+(x_1-x_2)^2}-x_1-x_2\right)+d$; where $d = \lambda e^{-(x_1-x_2)^2}$, $\lambda = 0.85$ and $\boldsymbol{x} \in [-1.5, 1.5]^2$.

Figure 2.2 demonstrates the set of individuals that are dominated (green dots) by a given individual (blue dot), the ones that dominate it (in red) and those that are incomparable.

In other contexts, MOPs solutions are called non-inferior, admissible or efficient solutions. The elements of this set are non-dominated with respect to all the others. The Dent problem has a connected convex-concave-convex front (see Figure 2.3). Although

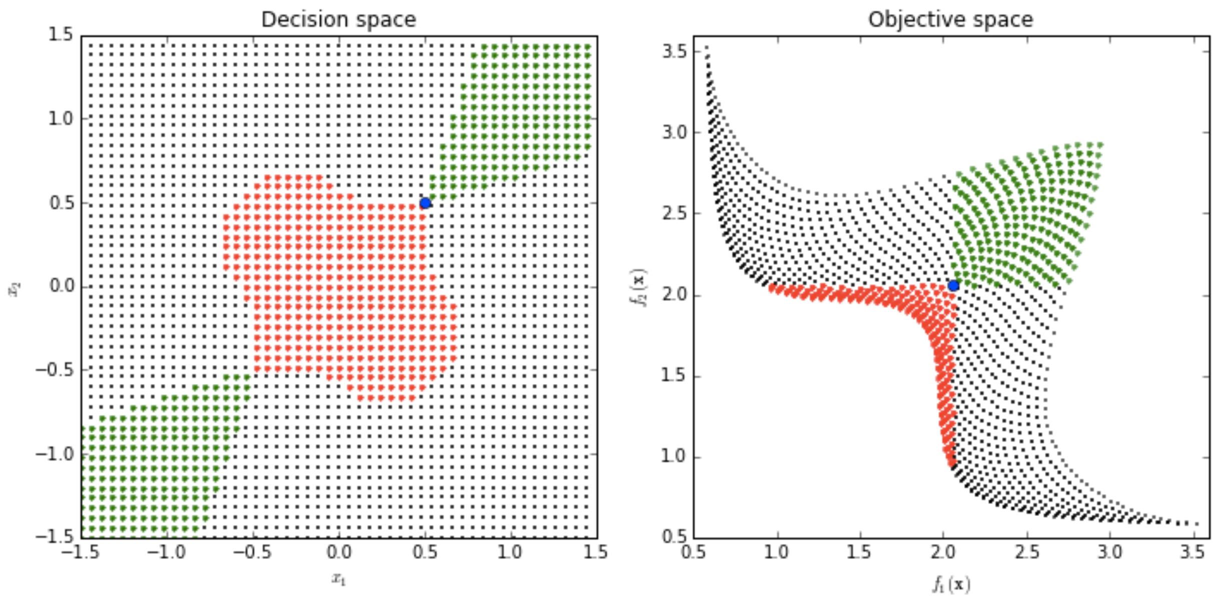


Figure 2.2: The dominance relation in Dent problem. Considering a given point (blue dot), in red are the points that dominate it and in green are the points that are dominated by the blue dot. Figure taken from [120].

elements of this solution set are equality optimal, it is required an interaction with a human decision maker to select the appropriated answer for the desired purpose.

On some optimization techniques is useful to know the lower and upper bounds of P_F [100]. The ideal point is sometimes used as a reference point which the method should optimize towards.

Definition 2.8 (Ideal point)**.** The ideal point denotes the array with the lower bound of all objective function and usually corresponds to a non-existent solution: $z_i^* = \min_{\boldsymbol{z} \in \mathcal{Z}} z_i, \forall i \in \{1, \ldots, k\}$. If all objective functions would be optimized individually, the composition of the values represent the ideal objective.

Definition 2.9 (Utopian point)**.** The utopian point is a vector strictly better than the ideal point in the search space: $z_i^{**} = z_i^* - \epsilon_i, \forall i \in \{1, \ldots, k\}$ and $\epsilon_i > 0$.

Definition 2.10 (Nadir point)**.** In turn, the nadir point is the upper bound of each objective function in the Pareto-optimal set: $z_i^{\text{nad}} = \max_{\boldsymbol{z} \in \mathcal{Z}} z_i, \forall i \in \{1, \ldots, k\}$. It is the worst value of every objective function of the P_S.

Figure 2.4 illustrates the lower (z_i^*, z_i^{**}) and upper bounds (z_i^{nad}) of P_F.

A subset X of $\mathbb{R}^n$ is convex if for any two pair of solutions $x^1, x^2 \in X$ and $\alpha \in [0, 1]$, the following condition is true: $\alpha x^1 + (1 - \alpha)x^2 \in X$ [141]. The intersection of all the convex sets containing a given subset X of $\mathbb{R}^n$ is called the convex hull of X. The convex hull of a set of points is the smallest convex set that contains the points.

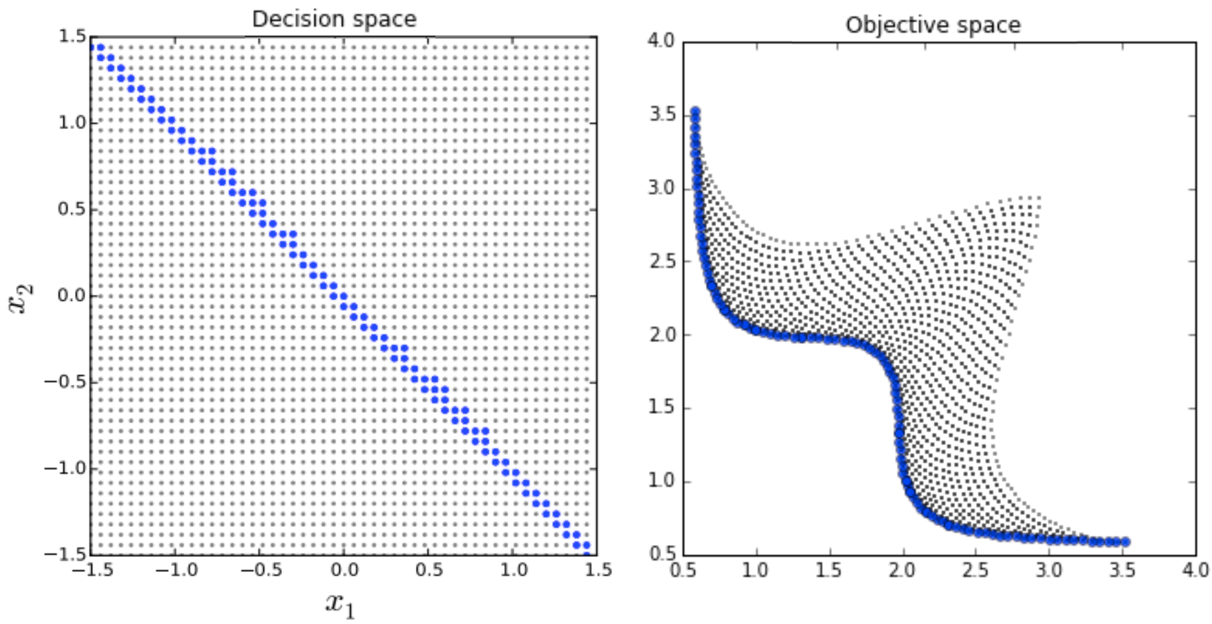


Figure 2.3: The non-dominated front in Dent problem. Figure taken from [120].

Definition 2.11 (CHIM)**.** The *convex hull of individual minima* (CHIM) [59] is the set of points in $\mathbb{R}^k$ that are convex combinations of $F_i^* - F^*$:

$$\mathcal{H} = \left\{ \Phi\beta : \beta \in \mathbb{R}^k, \sum_{i=1}^{k} \beta_i = 1, \beta_i \geq 0 \right\} \tag{2.2}$$

where x_i^* is the global minimizers of $f_i(\boldsymbol{x})$, $\forall i \in \{1, \ldots, k\}$. Let $F_i^* = F(x_i^*), \forall i \in \{1, \ldots, k\}$; and Φ is a pay-off matrix k x k whose the i^{th} column is $F_i^* - F^*$.

2.2 Evolutionary Computation

Scientists and engineers have frequently inspired in the nature and process of biological organisms to address those problems. There is a large set of approaches available for this purpose: evolutionary computation, ant-colony optimization, swarm intelligence, neural networks and molecular (DNA) computation. Their application can be seen across a growing number of disciplines, such as combinatorial optimization, complex data analysis and time-series prediction, trajectory planning in robotics, manufacturing and facility scheduling, aircraft design, resource allocation, evolution of rules for solving expert problems, communication network design, among others.

Probabilistic methods have proved to be capable of finding optimal or near-optimal solutions in a reasonable computational time. Evolutionary computation (EC) represents

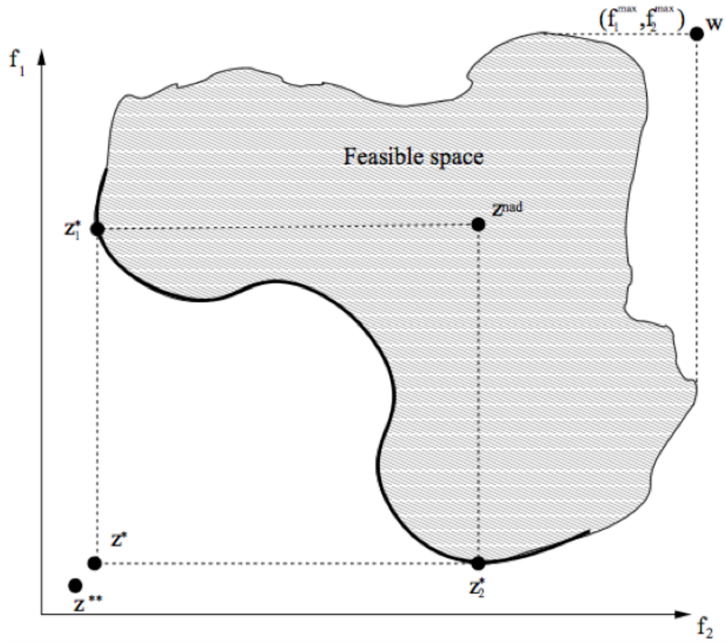


Figure 2.4: The ideal (z_i^*) and utopian (z_i^{**}) point representing the lower bounds of P_F. The nadir point (z_i^{nad}) representing the upper bound of P_F. Figure taken from [60].

a group of stochastic search techniques and optimization methods that simulate the natural evolution process. Genetic algorithm (GA) is a branch of EC. First developed by John Holland in the seventies [15, 93], GAs are inspired by Darwin's theory [58] about reproduction and natural selection – survival of the fittest. The genetic algorithms are adaptive heuristic search that not only exploit correlations in the search space, but also explore new and unknown areas to avoid local optimum points. In this way, they cover a wide range of problems like constraint satisfaction, combinatorial optimization, among others. GAs are often used to highly multimodal functions, discrete or discontinuous functions, NP-complete combinatorial problems and highly multimodal functions.

In the genetic algorithms, each candidate solution to a problem is represented as an individual with an associated fitness value. The population of candidate solutions P is evolved by successively applying four main operators: selection, crossover, mutation and elite-preservation [127]. At each iteration t, the best individuals are selected for survival and are included in the next generation population P_{t+1}. In this way solutions which are good can be used to generate better or similar solutions. The operation stops when some termination criterion is reached.

The standard GA has the following steps shown in Algorithm 2.1: 1) Choose initial population P_t; 2) Calculate the fitness function and associate the fitness score to each individual; 3) Loop until one or more pre-specified termination criteria are met.; 4) Per-

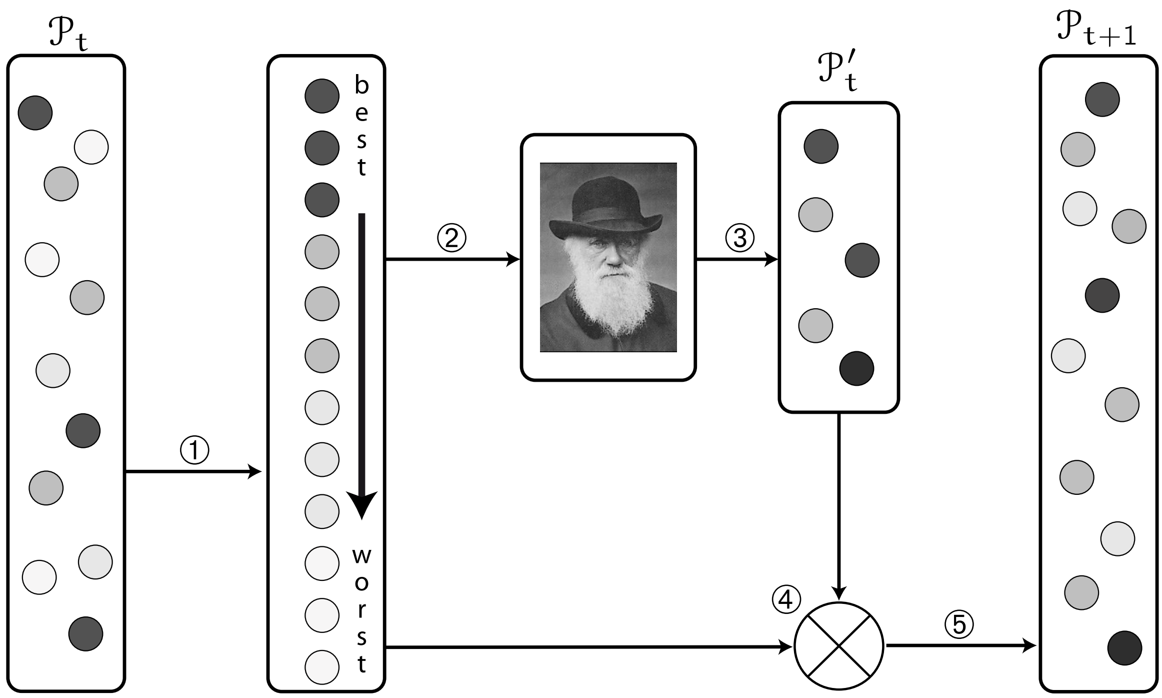


Figure 2.5: At generation t, individuals from population P_t are ranked according their fitness values (1). In (2) and (3), evolutionary operators modify the current population and create a new offspring population, $P_t^{'}$. Step (4) combines original and offspring populations to produce (5) the next population P_{t+1}. Figure taken from [119].

form selection; 5) Perform crossover; 6) Perform mutation; 7) Assign the fitness values to the new offspring population $P_t^{'}$ and perform elitism. The elitism keeps the good genetic characteristics for the next generation P_{t+1} and removes the individuals with worst fitness values.

Algorithm 2.1 Standard Genetic Algorithm.

```
choose initial population P_t
calculate the fitness function
while fitness value != termination criteria do
    selection
    crossover
    mutation
    calculate the fitness function
end while
```

The processes of a genetic algorithm iteration are represented in Figure 2.5. It is plainly shown on the diagram how the algorithm mimics the evolutionary process observed in nature.

The candidate solutions are encoded as finite-length string of values and often referred to as a chromosome or individuals. Each chromosome contains a set of characteristics (genes) that conveys the decision variables of the problem. The search space $\mathbb{G}$ of genetic algorithms usually represents a string chromosome as a fixed-length tuple of genes i_k:

$$\mathbb{G} = \{\forall\,(i[1], i[2], \ldots, i[n]) : i[k] \in \mathbb{G}_k \forall k \in 1..n\} \tag{2.3}$$

The stochastic operators are responsible for iteratively updating the current population. As already mentioned, the simplest form of genetic algorithm involves three types of operators: selection, crossover and mutation. The selection emphasizes the fitter individuals in the population. The main idea of selection is to prefer individuals for reproduction with high fitness over low-fitted ones. This will create offspring for the next generation with the belief that their individuals will have even higher fitness.

Different selection mechanisms are available in the literature [127,140]: roulette wheel selection, stochastic uniform sampling, tournament selection, rank-based selection, steady state selection, Boltzmann selection, sigma scaling, among others. A proper balance between the selection pressure and the variation operators must be maintained. A strong selection pressure may overestimate suboptimal highly fit individuals and compromise the diversity needed for evolution, whereas a week selection pressure will result in too slow progress (slow finishing).

Crossover is the primary instrument of variation in GAs. Basically, the crossover operator exchanges the genetic information of two individuals and produces one or two offspring. There is a crossover probability p_c that controls if the two selected individuals go through recombination or will be simply copied to the offspring population.

There are many variants of crossover [33, 127]: one-point crossover, two-point and N-point crossover, uniform crossover, linear crossover, arithmetic crossover, etc. Figure 2.6 presents the most popular crossover techniques. The one-point crossover chooses a single position at random and swaps the two tails after the position to form two new offspring. The two-point crossover chooses two positions at random and exchange the segments between them. In uniform crossover, every string value is exchanged with a certain probability, p_e, known as the swapping probability.

The crossover operator depends on the chromosome encoding, fitness function and other implementation details. It is still a very important open problem. There is not a definitive direction on the type of crossover that it is more appropriate for most of the problems.

Mutation randomly changes some of the values in a chromosome with a certain prob-

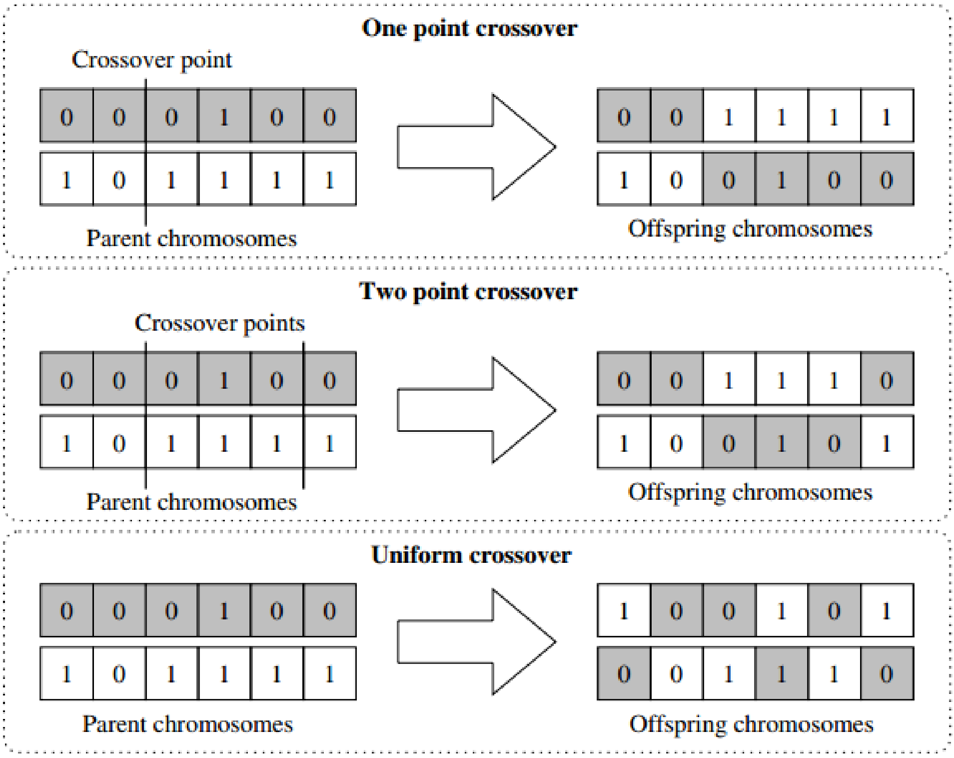


Figure 2.6: Variants of crossover operator [140]. The one-point crossover chooses a single position at random and swaps the two tails after the position. The two-point crossover chooses two positions at random and exchange the segments between them. The uniform crossover exchanges each string value with a certain probability (p_e).

ability. It operates on one parent individual and produces one offspring with changes. Currently there are numerous different types of mutation operator: random mutation, normally distributed mutation, polynomial mutation, flip bit, uniform and non-uniform mutation, etc. Figure 2.7 shows a mutated offspring after variation in its fifth position.

The mutation operator preserves diversity in the population. It allows the genetic algorithm to explore areas not explored by crossover, because it causes an unexpected movement in the search space. As a result, mutation prevents the population from stagnating at local optima.

2.3 Multi-Objective Evolutionary Algorithms

A non-dominated solution set may contain infinite points. Evolutionary algorithms (EAs) are a successful alternative for multi-objective optimization because of its search technique that relies on finite population of candidate solutions and the generation of many possible answers in every single run.

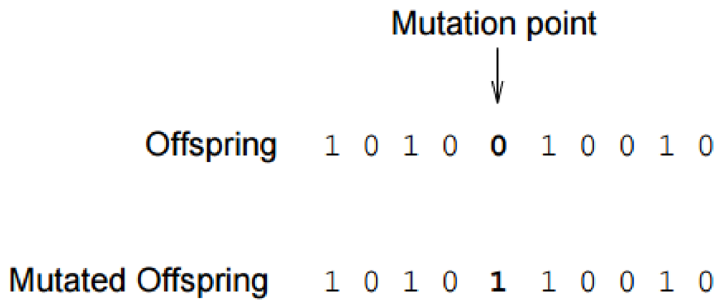


Figure 2.7: Mutated offspring after variation in its fifth position [23].

Multi-objective evolutionary algorithms (MOEAs) follow the common concepts of EAs. In every generation t, they find a set of individuals non-dominated by the rest of the population. The parent and the offspring population sizes are μ and λ, respectively [81]. A space of individuals $i \in I$ represents the candidate solutions of a population P: $P(t) = (i_1(t), \ldots, i_\mu(t)) \in I^\mu$. A problem-specific fitness function $F : I \rightarrow \mathbb{R}$ measures if certain solution satisfies the objective functions. Some operators in charge of reproduction (crossover and mutation) and selection create offspring generations until a termination criterion is reached, such as: a candidate with acceptable quality; a previous computational constraint; neither non-dominated solutions comes out. After running a MOEA, the final population detains an approximation set (S) of all non-dominated solutions with finite size that can be an appropriate representation of P_S.

The use of evolutionary algorithms has a number of practical advantages: (i) concept is simple to understand; (ii) inherently parallel and easily distributed; (iii) find multiple optimal solutions for multi-objective optimization; (iv) flexible building blocks for hybrid applications, like the algorithms proposed in this work.

2.4 Facility Location

This subsection presents the facility location problems in more details and, specially, a particular problem related to the Oil & Gas area named: field development planning (FDP).

Operations research (OR) is the application of advanced analytical methods to improve decision making and problem-solving. OR is often concerned with using techniques such as mathematical modeling to evaluate complex situations and determine the maximum or minimum of some real-world objective [35]. It addresses a wide range of problems, such as: optimal search, routing, supply chain management, transportation, scheduling, allocation, facility location, among others.

The facility location problem, also known as location analysis or k-center problem, is related to the process of assigning and managing facilities in an optimized way. It is concerned with how to place and efficiently utilize resources to achieve an organization's strategic goal, such as: minimization of operating cost, maximization of profit, fulfillment of demands, enlargement of sales coverage and market shares, among others. These problems have been studied extensively in operations research and management science field.

Many economical and logistic problems deal with a scenario where the competition for shared and scarce resources plays a major role in the decision process. Facility location decisions often involve large capital outlays and long-term planning horizons. Poor decisions in this domain may not be recovered without a large amount of money and time being expended. Consequently the expected result can get compromised if appropriate actions are not taken in an optimized manner. Examples are subsea layout design for the oil & gas exploration, vehicle routing, manufacturing plants, warehouse facilities, etc.

The facility location problem consists of a set of demand points D, a set of potential facility sites F, a fixed cost for opening each facility and a variable cost for each facility. The goal is to define positions for a subset S of facilities that should be opened and assigned to D. All the demand points must be serviced by a facility and the sum of fixed costs, variable costs, and transportation costs (usually modeled by distance) are minimized.

The properties assumed in the problem define different types of facility location. The number of new facilities that need to be located in the area of interest characterizes a single facility problem, with only one new facility to be settled, or a multi facilities problem with more than one facility to be located simultaneously. The capacity and services of the facility set two more types of problem. If facility can supply an infinite demand with unlimited capacity then it is denominated uncapacitated facility location problems (UFLP) and when facility's capacity of supply is limited then it is called capacitated facility problems (CFLP). An additional variation on the facility location problem is classified depending on number of services it is providing. A facility can supply only one type of service like in a food shop or can provide a group of services like in a general hospital.

There are three types of representation of space in location allocation problem: discrete, continuous and network-based. In discrete space model there are given a set of choices for the facility's location. The most commonly studied location problem in discrete model is the UFLP, also known as plant location problem or warehouse location problem. The continuous location consider any location within the space (usually Euclidean). The network-based model depends on graph-theoretic approach and representation. Regardless the space model, there can be some forbidden areas where the facility should not be placed or the designated route should not cross. For example new facility may not be built over water body and routes should avoid ineligible regions.

2.4.1 Classical Facility Location Problems

The Fermat-Weber problem is, historically, the first facility location problem studied as early as in the 17th century [30, 70]. The problem is defined with a given finitely distinct points $D_1, D_2, \ldots, D_m$ in $\mathbb{R}^n$ and positive multipliers $w_1, w_2, \ldots, w_m \in \mathbb{R}_+$. It has to find a point $F \in \mathbb{R}^n$ that minimizes the function:

$$\min f(F) = \sum_{i=1}^{m} w_i \|F - D_i\|, \tag{2.4}$$

here $\|F - D_i\|$ denotes the Euclidean distance of $X \in \mathbb{R}^n$. In other words, given m points, it requires finding a point P such that the sum of the Euclidean distances from P to the given points is minimum. The m points can be interpreted as customers location, P is the warehouse to be located and the weights w_i are the cost per unit distances of shipping the requirements to customers position.

Megiddo [125, 126] proved the Fermat-Weber problem is NP-hard. Tellier [153] proposed a geometrical solution and some iterative optimizing methods were used to address this problem. Kulin and Kuenne [109] suggested an algorithm based on iteratively reweighted least squares generalizing Weiszfeld's algorithm [159, 164] for the unweighted problem.

The set covering problem (SCP) is a classical complexity theory question presented in Karp's 21 NP-complete problems paper [102]. Given a ground set of n elements $U = x_1, x_2, \ldots, x_n$, a collection of m subsets $S = S_1, S_2, \ldots, S_m$ of that ground set where $S_i \subseteq U$, and an integer k. A set cover is a collection of subsets from S satisfied that every elements in U belongs to one of the subsets. A cost function $c : S \to \mathbb{Z}^+$ denotes the cost of a subset. The cost of the set cover is the sum of costs of each subsets in the collection of selected subsets. Formally, the goal is to find a set cover that minimizes the cost.

The set cover problem was studied extensively in the literature and applied to a wide range of problems, such as scheduling, delivery and routing problem, manufacturing, etc. The set cover is a NP-hard problem and many algorithms have been developed for solving it. Greedy algorithms were proposed by Chvatal [40] and Haouari [89]. Others heuristics such as those based on Lagrangian relaxation [38, 110] and meta-heuristics [11, 134] were developed based on the problem-specific information of the SCP.

2.4.2 Field Development Planning

There are many decision making problems whose information is spatial (geographical). In offshore field development, the equipment placement and submarine pipeline route for the hydrocarbons transportation demand a critical analysis. The Field Development Plan (FDP) is a key process of the Oil & Gas industry. FDPs evaluate multiple development options for a oil field and decide the appropriate scenario based on assessing trade-offs among numerous factors, such as: environmental impact, geophysics, geology, reservoir and production engineering, infrastructure, well design and construction, completion design, surface facilities, economics and risk assessment.

The FDPs comprise all process and activities required to manage an optimal placement of subsea equipment and the type of installations arrangements to develop an oil field. It is a complex engineering optimization problem. The mathematical model is characterized by a large number of design variables with objectives defined on high-dimensional spaces. It must comply with many constraints from the real-world. The routes have to overcome natural obstacles on the bottom of the sea and the marine environment (clime, depth, tides, corrosion, waves, etc) influences the choosing of subsea facilities.

Therefore, the selection of a submarine pipeline route that determines a good performance of the committed resources must be described and treated as a multi-objective facility problem. Figure 2.8 illustrates the conceptual study for the Ivar Aasen field located in the northern part of the North Sea. There are three oil well placed in a particular formation connected to two offshore platforms.

2.5 Collective Intelligence

Since the beginning of the 2000s, the development of social network technologies and interactive online systems has promoted a broader understanding of the “intelligence” concept. A new phenomenon appeared based not only on the cognition of one individual, but also placed on a network of relationships with other people and the external world.

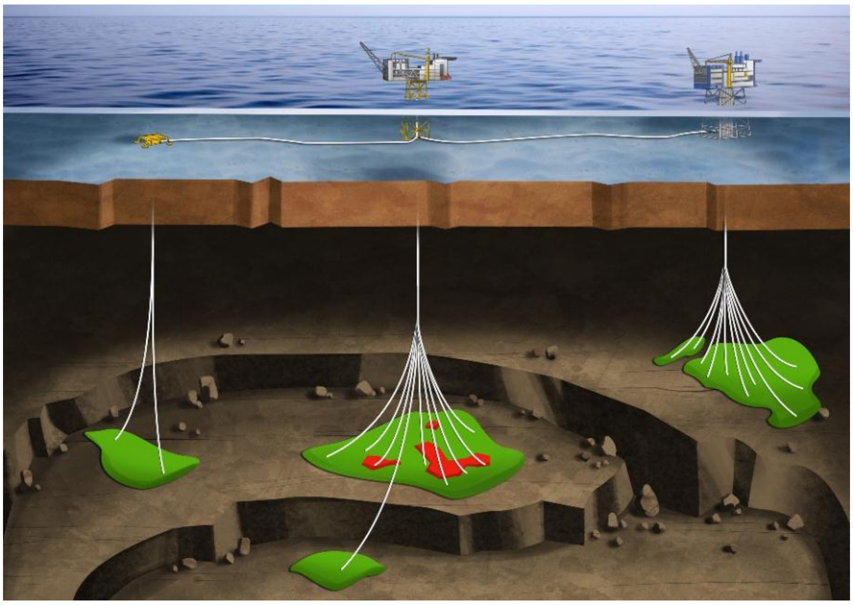


Figure 2.8: Conceptual study for the Ivar Aasen field with three oil well placed in a particular formation connected to two offshore platforms [1].

The field known as collective intelligence (COIN) [111,117] is defined as the self-organized group intelligence arisen from participatory and collaboration actions of many individuals. Shared tasks or issues are handled by singular contributions in such a manner that their aggregation process creates better results and solves more problems than each particular contribution separately [91,149]. This phenomenon develops a *sui generis* intelligence. It raises a global experience of collective attitudes without centralized control, bigger than its isolated pieces and sub-product of their combination.

COIN involves groups of individuals collaborating to create synergy and augment the intellectual processes of human beings. A decision-making process over the Internet has to manage users' interactions. It must get valuable knowledge concealed or dispersed in the group, even when the participants are not specialized in the subject. This environment includes large and heterogeneous audiences that are mostly independent among each others. Therefore, the problem must be decomposed in tasks that sustain diversity and transient members' attendances to align the interest of crowds.

Collective decision-making engages all the knowledge generated in the first stage of users' interaction and aggregates singular inputs into a global perspective for a decision. Some techniques to extract the group opinion or preferences are [163]: voting, recommendation systems, judgement aggregation, prediction markets and rating scales. While some initiatives collect the best information available from the crowd: wikis, document ranking and deliberation maps; others approaches like the elicitation of ideas assembles answers or suggestions to make all the inputs converge as a consensus.

Some COIN initiatives have proven their effectiveness over a wider range of areas. Amazon's Mechanical Turk [78] site outsources digital tasks that are difficult for computers, but not for humans, such as: tagging images, writing product descriptions, identifying performers on music and so on. InnoCentive site hosts companies' problems and offers a cash prize to the one who presents the most preferred solution. Both initiatives harness collective ideas, elaborate global preferences to hit the target and outperform a design expert. Affinova delivers a service to companies who want to improve their innovation and marketing rates in consumer packaged goods, retail, financial services and design. Its platform empowers teams to develop ideas, collect consumer feedback and predict the best execution plan for them. Danone, a global food company, used their services to launch the Activia product line in USA and the result beat the initial forecast by four times [6]. Another example is the puzzle game about protein folding: Foldit; it uses the human brain's natural three-dimensional pattern matching to solve the problem of protein structure prediction. The highest scoring solutions are analysed by researchers and validated if applicable in real problems or not. Users in Foldit has already helped to decipher the crystal structure of the Mason-Pfizer monkey virus (M-PMV) retroviral protease [104].

The free and easy-to-use application VizWiz [29] recruits web volunteers, including from Mechanical Turk marketplace, to help blind and visually impaired people. It sends photos with recorded questions about text labels, colors or icons and get answers back in real time from online sources. Duolingo is a platform for practice and learning of several languages. Its gamified background motivates the users to earn experience points as they progress on dictations and lessons. The site uses crowdsourcing to discuss or fix grammar topics and translate real content from the web. MatLab, a famous matrix-based language for fast numeric computation, launched a coding contest which entries are scored and ranked online [85]. The challenges are manifold, such as finding the n-th Fibonacci number as quickly as possible to plan or develop routes for the rovers in Mars. All the entries are visible and the contestants can modify an existing one and submit it again as their own entry. This strategy promotes a kind of *co-opetition* (collaboration plus competition) that makes the solutions evolve by the collaboration of many people. Xprize, a non-profit organization, defines itself as an innovation engine and a catalyst for the benefit of humanity. This institution stimulates prize competitions on subjects like: global development and sustainable solutions; energy and climate change; life sciences and education. There is a monetary rewards for the winners, but the real intention is to encourage the global collectivity to invest the intellectual capital required for difficult problems.

There are plenty of examples that promote the collaboration of many participants to achieve better outcomes. This advantage motivates the incorporation of collective intelligence within MOEAs.

2.6 Clustering Algorithms

Inside collective environment, contributions come from different people. Clustering algorithms distinguish the users with similar preferences to perform a cooperative evolution or decision making choice.

2.6.1 Gaussian Mixture Model

A mixture model is a probabilistic model to reveal distributions of observations in the overall population [19,146]. Given a data set $Y = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N\}$ where $\boldsymbol{y}_i$ is a d-dimensional vector measurement with the points created from density $p(\boldsymbol{y})$, a finite mixture model is defined as:

$$p\left(\boldsymbol{y}|\Theta\right) = \sum_{k=1}^{K} \alpha_k p_k\left(\boldsymbol{y}|z_k, \theta_k\right) \tag{2.5}$$

Let $K \geq 1$ be the number of components, $p_k\left(\boldsymbol{y}|z_k, \theta_k\right)$ be the mixture components where each k is a density or distribution over $p\left(\boldsymbol{y}\right)$ and parameters θ_k, $\boldsymbol{z} = \langle z_1, \ldots, z_k \rangle$ be a K-ary random variable defining the identity of the mixture component that produced $\boldsymbol{y}$ and $\alpha_k = p_k\left(z_k\right)$ are the mixture weights representing the probability that $\boldsymbol{y}$ was generated by component k. Hence, the parameters for a mixture model is $\Theta = \{\alpha_1, \ldots, \alpha_K, \theta_1, \ldots, \theta_K\}$, $1 \leq k \leq K$.

The Central Limit Theorem [84], explains why many applications that are influenced by a large number of random factors have a probability density function that approximates a Gaussian distribution. Let Y be a sequence of random variables that are identically and independently distributed, with mean μ and variance σ^2. The distribution of the normalised sum $S_n = \frac{1}{\sqrt{n}}(\boldsymbol{y}_1 + \ldots + \boldsymbol{y}_N)$ approaches the Gaussian distribution, $G(\mu, \sigma^2)$, as $n \to \infty$.

In a Gaussian mixture model, each of the K components is a Gaussian density with parameters $\theta = \{\boldsymbol{\mu_k}, \Sigma_k\}$, $\boldsymbol{y} \in \mathbb{R}^d$ and function as:

$$p_k(\boldsymbol{y}|\theta_k) = \frac{1}{(2\pi)^{d/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{\mu}_k)^t \Sigma_k^{-1}(\boldsymbol{y}-\boldsymbol{\mu}_k)} \tag{2.6}$$

The expectation maximization (EM) algorithm [68] for Gaussian mixture is a particular way of implementing the maximum likelihood estimation in probabilistic models with incomplete or missing data values. EM learns the parameters θ_k guessing a distribution for the unobserved data and finds the cluster to which a singular chromosome most likely belongs. It starts with an initial estimation of Θ and iterates between E-step and M-step of the algorithm to update Θ until convergence.

E-step estimates the posterior distribution of the latent variables taking into account the current parameters and the observed data. The membership weight w_{ik} computes the probability of all data points $\boldsymbol{y}_i$ to the mixture components k. In the M-Step, the algorithm uses the calculated membership weights to find new model parameters values. Let w_{ik} be the weights where $1 \leq k \leq K$ and $1 \leq i \leq N$.

$$w_{ik} = p(z_{ik} = 1|\boldsymbol{y}, \Theta) = \frac{p_k(\boldsymbol{y}|z_k, \theta_k)\,\alpha_k}{\sum_{m=1}^{K} p_m(\boldsymbol{y}|z_m, \theta_m)\,\alpha_m} \tag{2.7}$$

After E and M steps the convergence is computed using the value of the log-likelihood $\log l(\Theta)$. The algorithm stops when there are no significant changes in the convergence from one iteration to the next.

$$\log l(\Theta) = \sum_{i=1}^{N} \log p(\boldsymbol{y}|\theta) = \sum_{i=1}^{N} \left(\log \sum_{k=1}^{K} \alpha_k p_k(\boldsymbol{y}|z_k, \theta_k) \right) \tag{2.8}$$

2.6.2 K-means

K-means is another popular algorithm for cluster analysis [151]. It sets apart n observations $(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)$ into k clusters $S = \{S_1, S_2, \ldots, S_k\}$, where $k \leq n$ and each observation $\boldsymbol{y_i}$ is a d-dimensional real vector that belongs to the set S_j whose mean yields the least within-cluster sum of squares:

$$\arg\min_{S} \sum_{i=1}^{k} \sum_{\boldsymbol{y} \in S_i} \| \boldsymbol{y} - \mu_i \|^2 \tag{2.9}$$

The algorithm starts with k means $(\mu_1, \ldots, \mu_k)$ and iterates between two steps: a) assignment of $\boldsymbol{y}$ to the cluster with the nearest mean; b) recalculation of the clusters' centroids.

2.7 Reference Points and Interactive MOEAs

2.7.1 Preferences

The current state-of-the-art MOEAs are capable of obtaining reliable approximations of the Pareto optimal front. Many evolutionary frameworks have been applied to MOPs and broadly used to optimization. Some MOEAs like: SMS-EMOA [28]; MO-CMA-ES [95]; NSGA-II [64]; MOEA/D [174] or SPEA2 [179], exploit a set of solutions in parallel and supply a starting point to new developments on the field. They look for the Pareto optimal set, P_S, that ideally contains all non-dominated solutions uniformly distributed along the optimal frontier.

However, the optimal frontier (P_F) might be extremely large or possibly infinite and the DMs still must identify a final answer to their demands from this trade-off set. The challenge is no longer just to obtain a diversity of answers in the entire high-dimensional Pareto front, but also retrieve the expected solutions aligned to consistent preferences of the DMs. In most of the cases, their preferences are determined in the objective space.

Preferences are user-defined parameters and denote values or subjective impressions regarding the trade-offs points. It transforms qualitative feelings into quantitative values to bias the search during the optimization phase and restrict the objective space. In this sense, a reliable preference vector improves the trade-off answers obtained. Usually, preference is represented by a set of criteria I, where $i \in I$ corresponds to one preference information assigned to one attribute. The boundaries are usually set by the *ideal* ($\boldsymbol{z}^*$), *utopian* ($\boldsymbol{z}^{**}$) and *nadir* ($\boldsymbol{z}^{\mathrm{nad}}$) points [167].

Local preferences can be expressed as a vector of weights over the objectives [56], a lexicographic sorting of objectives [22, 74], a set of constrains [73], a trade-off information or reference points for the search [51, 60], among others representations.

MOEAs techniques are classified by their articulation of preferences: *a priori* approach performs decision before the searching process; interactive (progressive) method combines search and decision making; *a posteriori* technique searches before making decision.

Some traditional *a priori* techniques are dependent to the selection of objectives' weights or boundaries. The weighted sum method [56] associates a weight to each objective function and takes its sum to revert the problem to a mono-objective equation: $F(\mathbf{x}) =$

$\sum_{m=1}^{M} \lambda_m.f_m(\mathbf{x})$. Lexicographic ordering method [22, 74] prevents the manipulation of *a priori* weights by setting priorities to the objective functions and arrange them in order of importance. The ε-constraint approach selects one objective function to be optimized and restricts all the others within user-specified inequality constraints [73]: $min\ f_i(\mathbf{x})$; subject to $f_k(\mathbf{x}) \leq \varepsilon_k$ where $k = 1, \ldots, n$; $k \neq i$. Distance-to-a-reference method [51] compares a vector $\mathbf{f} \in \mathbb{R}^k$ correspondent to an ideal or reference objective chosen by the decision maker against a solution candidate $\mathbf{x}$. The Goal Attainment technique [60] also uses a vector $\mathbf{f} \in \mathbb{R}^k$ of reference objective function plus a weighted vector $\mathbf{w}$ to minimize a scalar coefficient Ψ in such a way that: $f_i(\mathbf{x}) - w_i.\Psi \leq \mathbf{x}_k$, where $i = 1, \ldots, k$.

The Guided Multi-Objective Evolutionary Algorithm (G-MOEA) proposed by Branke et al. [36] allows to specify linear trade-offs between objectives. G-MOEA uses these trade-off information to modify the definition of dominance and guide the search towards the more desired regions of the Pareto-optimal front. The DM specifies maximal and minimal acceptable weightings for one criterion over the other. For example, in a two-objectives problem the decision maker has to specify how many units of the first objective (f_1) he is willing to trade for one unit of the second objective (f_2).

$$
\begin{aligned}
\mathbf{x} \prec \mathbf{y} \Leftrightarrow (f_1(\mathbf{x}) + a_{12}f_2(\mathbf{x}) \leq f_1(\mathbf{y}) + a_{12}f_2(\mathbf{y})) \ \wedge \\
(a_{21}f_1(\mathbf{x}) + f_2(\mathbf{x}) \leq a_{21}f_1(\mathbf{y}) + f_2(\mathbf{y}))
\end{aligned} \tag{2.10}
$$

Reference points and interactive techniques aggregate different strategies to discover more relevant solutions in P_F. Instead of a computation of the whole front, these methods get suggestions or hints to highlight the regions and operate only on areas previously selected to get a preferred sub-set of the front.

2.7.2 Reference Point

The reference point approach [168] concentrates the search of Pareto non-dominated solutions in the vicinity of a set of selected preference points. It is based on the achievement scalarizing function that uses a reference point to capture the desired values of the objective functions. Let $\boldsymbol{z}^0$ be a reference point for an k-objective optimization problem of minimizing $\boldsymbol{F}(\boldsymbol{x}) = \{f_1(\boldsymbol{x}), ..., f_k(\boldsymbol{x})\}$, the reference point scalarizing function can be stated as follows:

$$\sigma\left(\boldsymbol{z},\boldsymbol{z}^0,\boldsymbol{\lambda},\rho\right)=\max_{i=1,\ldots,k}\left\{\lambda_i(z_i-z_i^0)\right\}+\rho\sum_{i=1}^{k}\lambda_i\left(z_i-z_i^0\right), \qquad (2.11)$$

where $\boldsymbol{z}\in\mathcal{Z}$ is one objective vector, $\boldsymbol{z}^0=\langle z_1^0,\ldots,z_k^0\rangle$ is a reference point vector, σ is a mapping from $\mathbb{R}^k$ onto $\mathbb{R}$, $\boldsymbol{\lambda}=\langle\lambda_1,\ldots,\lambda_k\rangle$ is a scaling coefficients vector, and ρ is an arbitrary small positive number. Therefore, the achievement problem can be rebuilt as: $\min\ \sigma\left(\boldsymbol{z},\boldsymbol{z}^0,\boldsymbol{\lambda},\rho\right)$.

Since the decade of 80's, there have been several works on interactive multi-objective methods using reference points and reference directions as preferences. Those approaches were applied mainly in the classical multi-objective programming field. But in the last 15 years they have also emerged in evolutionary multi-objective area.

Deb et al. [66] proposed a reference-point-based NSGA-II procedure (R-NSGA-II) to find a set of solutions in the neighbourhood of the corresponding Pareto-optimal front. A minimum weighted normalized Euclidean distance calculates the preference distance of a solution and uses that information to bias the selection operator. They extend the number of reference points to cover more than one region in Pareto front and reach more significant solutions. The Synchronous R-NSGA-II [76] is a similar approach, but uses three different scalarizing functions instead of the Euclidean distance. The Light Beam Search based EMO [63] modified the NSGA-II crowding operator by the light beam search to incorporate *a-priori* preferences and produce a set of solutions in the region of interest. The decision maker (DM) must supply an aspiration and a reservation point to determine the direction of the search. Another variation of the original NSGA-II, the RD-NSGA-II [62] let the user supply one or more reference directions to project efficient solutions on the Pareto-optimal frontier. The principle is the application of multiple achievement scalarizing functions (σ) to generate non-dominated fronts. Said et al. [25] presented a new dominance relation, named reference solution-based dominance (r-dominance), that replaces the Pareto dominance and favours solutions near the reference point indicated. Pfeiffer and others [137] rank the solutions according their Euclidean distances to each reference point. Those points close to the reference points are assigned with the lowest crowding distance, which favours its selection by the algorithm. The negotiation support system called W-NSS-GPA [24] takes the reference points and the decision makers' hierarchy levels as weights to calculate an aggregation point of all preferences.

Figure 2.9 shows five reference points and their respective points in Pareto frontier extracted from the well-known ZDT1 test suite [178].

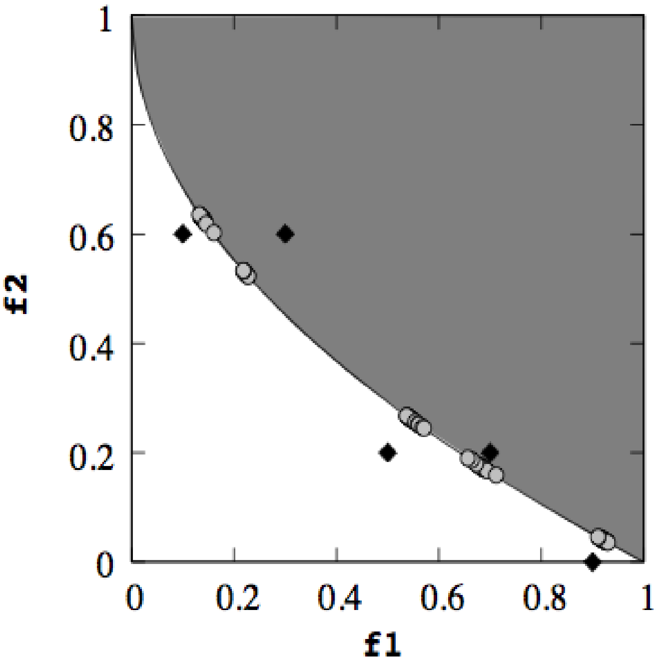


Figure 2.9: R-NSGA-II solution for ZDT1 problem with five reference points taken from [66].

In many-objective optimization problems, the high dimension of objective space makes a large number of individuals become non-dominated to each other. This condition may not produce enough selection pressure for a population to converge to a satisfactory Pareto front approximation. The NSGA-III [61] uses reference points on a hyperplane to overcome this issue and guide the search process through a diverse population. In DI-EMOA [20], decision makers specified region of interests of the objectives and defined the desirability functions (DF) accordingly. A desirability index (DI) was then defined as a scalarization operator to map multiple DF values to a single value. The DI was used as the secondary criterion of the non-dominated sorting procedure. A DI-based archiving mechanism was also proposed. The DF-SMS-EMOA [160] maps the objectives to desirability functions normalized in the domain $[0, 1]$, $d : Y \to [0, 1]$. Then, values of different objectives and units become comparable. The preferences are represented by the difference between the actual desirability and the maximum value of one: the smaller the difference is, the better the quality of the solution in the corresponding objective.

2.7.3 Interactive Evolutionary Multi-Objective

Interactive genetic algorithm (IGA) incorporates the evaluation of users on the candidates of evolutionary algorithms to solve problems whose optimization objectives are complex to be defined with exact functions [148]. Users' subjectivities are employed as fitness values to drive the search throughout the evolution process.

IGAs were successfully applied to get feedback of transitional results throughout the evolution process and steer the search towards preferred parts of the P_F. This behaviour gives the DMs a chance to extend their knowledge about the problem and review their preference model over time.

MOEAs can handle intermediate non-dominated solutions to the decision maker and improve the search with a reference point or fitness function adjustments. W-HYPE [37] applied the weighted hypervolume indicator in an interactive fashion to change the optimization goal of the algorithm. The DM indicates preferred solutions in the current population and, as a consequence, this affects the weight function used by the hypervolume indicator. iMOEA/D [82], an interactive version of the decomposition based MOEA, asks the DMs to analyse some current solutions and use their feedback to renew the preferred weight region in the following optimization. It converts the a MOP into a set of single-objective problems. PI-EMO [65] changes progressively the value function after every few generations to lead the optimization on more preferred solutions. After t generations, the user analyses η (≥ 2) well-sparse non-dominated solutions and provides a complete or partial ranking of preference information about them. The task is to construct a polynomial value function satisfying the given preference hint. Thiele et al. [155] introduced a preference-based interactive algorithm (PBEA) that adapts the fitness evaluation with an achievement scalarizing function to guarantee an accurate approximation of the desired area in Pareto-optimal front.

The necessity of multiple human interactions may cause user fatigation and deserves attention in order to avoid slow and expensive performance. Distinct approaches with fuzzy logic and machine learning concepts simulate users' preferences to prevent this issue and alleviate the constant interruption of the algorithm. The algorithm IGAMII [14] applies fuzzy logic to simulate the human decision maker and relieve the constant interaction during the evolution. In BC-EMO [131], the Support Vector Ranking algorithm is used to learn an approximation of the DM utility function. Based on the concept of coevolving a family of decision-maker preferences together with a population of candidate solutions, Wang et al. [18] proposed the PICEA-g algorithm. PICEA-g does not ask users to provide reference points; instead, it evolves these reference points (called goals in the algorithm) by preferring those dominated by fewer solutions. On the other hand, solutions are regarded as better if they dominate more goals, especially when the goals are dominated by few solutions.

Not only evolutionary algorithms, such as genetic algorithms, can be used to answer multi-objective problems. The bee colony optimization is a swarm intelligence based algorithm that explores the natural behaviour of honeybees to resolve multi-objective numerical functions [173]. Particle swarm optimization is another adaptive method based on a set of individual particles swarming spread in the parameter space of the problem. It can also be interactive and applied to MOPs (IMOPSO). The particles values are updated according its objective function to reach the best solution [90, 132, 165].

Yazdani [170], for instance, presents a particle swarm multi-objective optimization enhanced with a fuzzy logic-based controller to evaluate the search space throughout the iterations. Inspired by 'divide and conquer" strategy, the general idea is to break down the problem into several simpler ones and use fuzzy logic to select which part should be chosen for the next iteration. Wickramasinghe and Li [165,166] incorporated the reference point theory into two multi-objective particle swarm algorithms: non-dominated sorting PSO (NSPSO) [113] and maximinPSO [114]. Through multiples reference points, the decision maker guides the search and spreads the solutions along different areas of Pareto front. The distance metric updates particles positions and velocities according to their proximity to preferred regions in the objective space.

Chapter 3

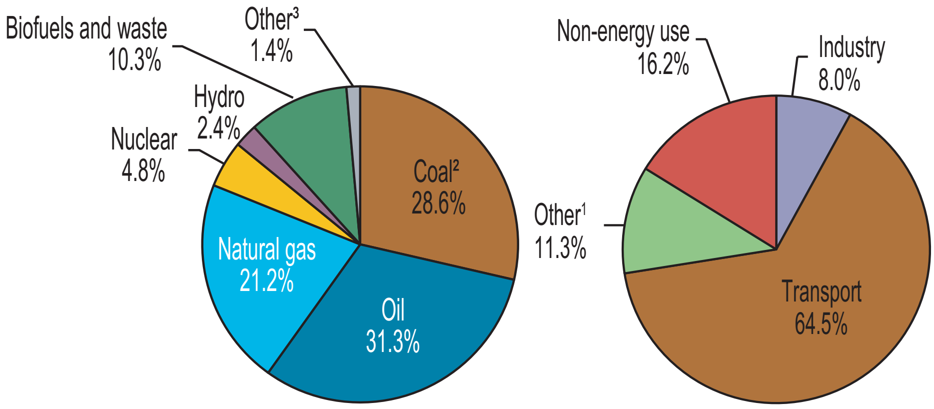
Application Domain

The domain application of this research is the multi-objective facility location problems with spatial orientation. The optimal placement of facilities is critical for activities where the balanced allocation of resources advises the right strategy for an enterprise. It involves many stakeholders that must consider multiple objectives and operational constraints simultaneously.

Some facility location problems have to evaluate geographical information to support decision-making. Besides choosing the available resources to be used, the spatial nature of the problem takes into account factors like avoiding obstacles, designing trajectories, and preventing proximity to competitors' facilities. The correct 2D/3D orientation of elements in the scenario directly influences the configuration of the best solutions and, therefore, affects the distribution of non-dominated points in the trade-off set.

The facility location problem is strategic in nature. The constraints on the location and placement of multiple facilities emulate real-life situations. These problems carry out money transactions. They charge large sums of capital resources and produce economic effects in the short and long run. For example, investors set oil company valuations based on the ability and positive experiences the company had in the previous field development projects. As a result, the Oil & Gas industry is under pressure from stakeholders to capitalize discoveries and optimal solutions as quickly as possible.

However, the complexity of these location problems is usually high and requires substantial computational effort to solve them. The time spent on these activities is associated with money. Then, it will be important to develop heuristic algorithms that lead to efficient, effective and fast solutions for practical multi-objective location problems.



(a) Energy supply by fuel (b) Energy consuption by sector

Figure 3.1: Total primary energy supply by fuel and the final consumption by sector. [1] Includes agriculture, commercial and public services, residential, and non-specified other. [2] Peat and oil shale are aggregated with coal. [3] Includes geothermal, solar, wind, heat, etc. Figure taken from [10].

3.1 Oil Field Development Problem

The oil market still represents a significant share of the world energetic matrix. According to the International Energy Agency (IEA), the oil and gas are the main source of energy with about 52.5% of the market share [10]. Figure 3.1 presents the total primary energy supply by fuel and the final consumption by sector.

One special case of facility location problem with spatial orientation is the field development planning. As presented in Section 2.4.2, the FDP performs feasibility and conceptual studies for pipeline installation and subsea construction. It is responsible for the definition and sizing of the main piping, drilling and subsea support equipment to deliver the infrastructure for a complete field development. Figure 3.2 illustrates the tools for pipelaying and subsea equipment installation.

This work focuses on the problem of selecting the optimal submarine pipeline routes employed to carry the oil & gas from wells to offshore platforms. For this purpose, multiple conflicting development objectives must be addressed: operational flexibility and scalability, capital versus operating costs, environmental impact, project risk and uncertainty, among others.

The submarine pipeline route for the oil has to overcome not only natural obstacles on the seafloor, but remained subsea equipment, pipelines and flowlines. Also, there may be constraints to access a specific area on the bottom of the sea, such as regions with corals or competitors oils fields. Figure 3.3 shows some oil & gas routes connected to one

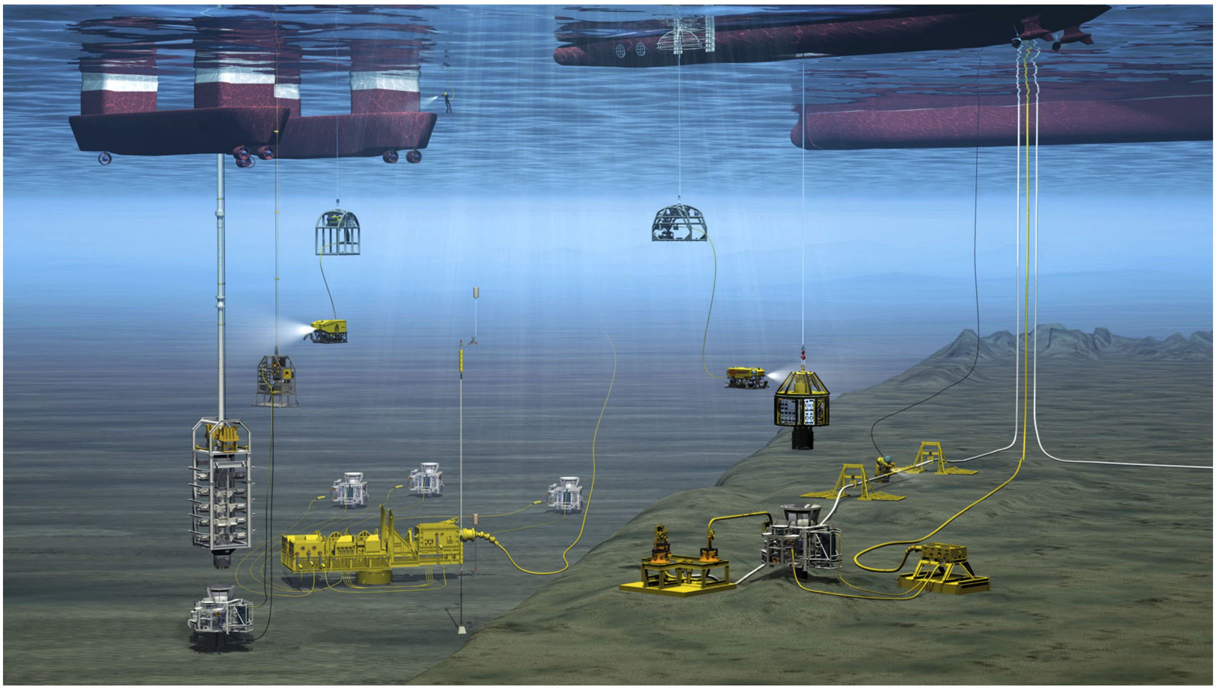


Figure 3.2: Tools for pipelaying and subsea equipment installation. Figure taken from [5].

Floating Production Storage Offloading (FPSO) unit.

The lifetime of an oil reservoir varies between 15 to 30 years and may be extended up to 50 years. The wells go through different stages during this cycle: production increase, stabilization period, injection of water to maintain a satisfactory volume and, the final stage, production decrease. The choosing of subsea facilities follows the technical constraints and uses innovation to tackle challenges such as extreme water depth and rugged sea-floor terrain. Pipes can be flexible with internal diameters ranging from 2" to 20" or rigid pipes up to 16" diameter. The offshore platforms have progressed in deepwater installation and today they can operate in water depths beyond 2.800 meters.

The early phases of a FDP conceptual study are the most important. It is where the most value is created or lost. Mistakes in this phase cannot be recovered in the project execution. Oil & Gas fields are rarely straightforward nowadays. There is difficult fluids, obscure location to drill and complex facilities to allocate in an optimal way. The combination of these factors needs special consideration. In the general case, the selection of a route is based on a specialist decision. The expert engineer manually investigates the seabed bathymetry and the available information regarding obstacles to validate the design of a FDP. But this is a very complex and expensive process to be based purely on the expertise of an engineer. It involves a large number of design variables with objectives defined on high-dimensional spaces. The field development project may contain around

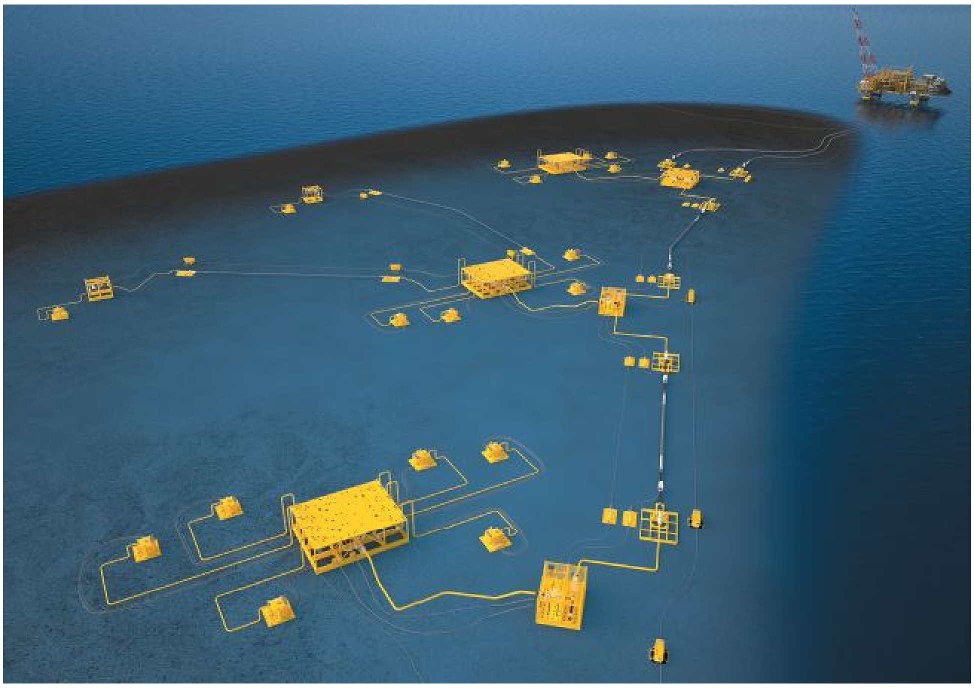


Figure 3.3: The subsea development of an Oil & Gas field. Different submarine pipeline routes connected to one FPSO. Figure taken from [2].

50 wells and 8 offshore platforms producing 100.000 oil barrels a day for 30 years. The flexible lines may cost $10.000 USD per meter and a large oil field may contain hundreds kilometres of lines. The platform cost ranges between 1 to 2 billion USD approximately.

The economy drives the success of this business. It is expected that the traditional approach with interactive and manual analysis should be replaced by a more robust procedure. In the current model, the design time needed to assess an optimal pipeline route is too long and it is subjected to interpretation misleads. The market leading company in planning and development phase of oil and gas projects takes months to draft a FDP project with the support of many experts. The general idea of this work is to use a different technique to provide more accurate results and minimize the costs associated with the installation and operation of subsea facilities.

3.2 Computational Representation and Time Complexity

In most of the models for location theory, the facilities are represented as points in the Euclidean plane (2D space) or vertices of a given graph. The routes can be represented as straight lines or curves. Topological relations like distance, connectivity, inclusion, adjacency and overlapping between spatial objects must be precisely defined to reflect all the real-life behaviour of the objects.

The location problems differentiate between continuous and discrete problems with respect to the decision space. In the discrete world, space can be represented as a surface divided into a regular grid of cells (raster model). A matrix $M(i, j)$ defines all the cells, named as pixels. Field development planning is a complex problem whether in discrete or continuous world simulation. But a continuous representation requires more computational effort in this type of problem due to the increase of facility positions available (x,y coordinates) and object boundaries.

Considering a simplified example to illustrate the complexity of FDP in discrete world, let n be the available positions (cells) in the 2D space and two distinct facilities (f_1 and f_2) to be connected through line segments (sequence of positions in n). The number of arrangements of any subset of n distinct objects is the number of one-to-one sequences that can be formed from any subset of n distinct objects. It takes the form of:

$$n! \sum_{k=0}^{n} \frac{1}{k!} \approx n!e \tag{3.1}$$

The asymptotic complexity of this simplified example is $\mathcal{O}(n!)$

Chapter 4

Proposal

This chapter presents the idea of integrating collective intelligence into a genetic operator to improve the overall quality of evolutionary population.

4.1 Proposal of COIN-based Selection and Variation Operators

Meta-heuristics approaches, like MOEAs, confront difficulties with complex scenarios of high-dimension and large problem space. The potential number of objectives necessary to describe the environment or the incapacity to comprehend and map all the variables to a correct fitness function can prevent a solution in a reasonable time and quality. On the other hand, human beings are used to multi-objective situations in their everyday lives. Those complex scenarios that are hard for a computer might be easier or natural to the human mind.

Persons are able to improve the multi-objective algorithms with cognitive and subjective evaluation to find better solutions. The human ability to see complex solutions is a motivation to involve people in tasks that are currently difficult for computers. It is possible for non-experts to make useful contributions to problems and leverage the computer methods to create novel results.

COIN is a different level of abstraction and can be a special contribution to make MOEAs go beyond their reach. Human characteristics such as perception, spatial reasoning, strategy, weighting factors, agility, among others subjectivities might be introduced into the algorithm to generate a better pool of answers and enhance the optimization process. A group of people can understand conflicting situation involving multiples objectives and may use their collective intelligence to trump expert's abilities. The wisdom arisen from the diversity of many individuals is able to discover creative resolutions.

While some MOEAs techniques construct a partial order of preferences based on *a priori* reference points to give a stronger selection pressure among Pareto-equivalent solutions, others progressive methods combine simultaneously the preferences information and the search for solutions. But very few MOEAs consider more than one user for reference point selection or evolutionary interaction. They neglect a collective scenario where many users could actively interact and take part of the decision process throughout the optimization.

The association of collective intelligence features to multi-objective optimization field raises the understanding of preferences from an individual context to a collective perception and has yet to be properly addressed. This work presents a collective intelligence operator to bias the search during the optimization phase and restricts the objective space. The main idea underlying this method is to drive the DM's search towards relevant regions in Pareto-optimal set and, also, promote the usage of COIN as a creative search for new individuals. By means of people's heterogeneity and common sense, the COIN operator iteratively refine the search parameters with rational collaborations to improve the overall quality of evolutionary population. The suggested approach decreases the number of function evaluations, accelerate the convergence and achieve relevant regions of Pareto front at a lower computational cost.

As to the quality and performance aspects of the proposal, the measure indicated contains both the quality of the outcome as well as the computational resources spent to generate this result. Regarding the latter aspect, it is advisable to keep the number of fitness evaluations or the overall runtime under observation [182].

A collective reference point produced by the interaction and aggregation of multiple opinions may provide a more accurate reference point than designed by only one DM (unilateral). A unique decision maker carries the risk of having mistaken guidelines or poor quality in terms of search parameter. In addition to the risk of a biased preference, the lack or unavailability of expert decision makers increases the difficulties to elicit preferences that will explore interested areas of the objective space. Conversely, the synergy of actions and the heterogeneity inside collective environments develop creative resolutions based on the subjectivity and cognition of the crowds.

In practical terms, the COIN-based operator is responsible for aggregating the users' contributions to the current population P at generation t. After the reproduction operators (crossover and mutation), the COIN operator chooses the best individuals and submits them to the collective intelligence. There are two types of collective intelligence operators proposed: COIN-based Selection operator and COIN-based Variation operator.

The COIN-based Selection operator performs a voting process where the current individuals from P_F are handed to collective members for judgement. Participants select the best ones of the group in a pairwise comparison manner. In the end of the process, the selected individuals will be used as samples to discover the collective preferences. The comparison of individuals defines a pattern of preference that guides the exploitation of points in areas of interest.

The COIN-based Variation operator, by contrast, submits some of the current individuals from P_F to the members of the collective. Then, participants can use their cognition and reasoning skills to improve or produce new individuals to be placed back in the population. They can manipulate and change the values of intermediate MOEAs solutions. The COIN variation operator is responsible for aggregating the users' rational contributions into the current population P. Both operators return a rational input to the current population.

Exploration consists of probing a larger area of the search space. This global search operation tries to find new and unknown regions with promising solutions. It diversifies the search in order to avoid the local optimum. Exploitation, on the other hand, operates in a limited area of the search space. This local search operation tries to improve a promising solution already at hand by intensifying the search in its vicinity.

In this regard, the creation of new EA individuals built under rational supervision may reveal interesting effects on the optimization search. The first is the exploitation of points on a limited area of the search space. The user contributions not only give a new individual to the population, but also indicate an area of interest to perform the search. This local search operation tries to improve a promising solution already at hand by intensifying the search in its vicinity.

Another positive effect is the exploration of new areas in the search space. The contributed individuals may bias the search to new and unknown regions with promising solutions. It diversifies the search in order to avoid the local optimum and explore areas driven by the collectivity.

The collective intelligence variation operator can be seen as a sort of local search as it can be presumed to provide improvements over currently existing solutions.

The collective reference points are defined based on the overall contributions. Whether they are a complete new individual or a set of users' votes, the contributions represent the collective preferences and are used to delimitate areas of interest on the approximation set during the optimization process.

Some types of facility location problems request complex spatial reasoning and orientation to obtain a satisfactory solution. For example, if one have to trace many routes avoiding obstacles and minimizing the setup cost of resources allocation. This kind of problem is still difficult for computers to fully understand and produce acceptable alternatives. This question becomes even more complex if there are more objectives in place, like shortest distance and client demands. Usually, computers inspect many intermediate solutions of poor quality that are easily recognized as unreasonable by humans. The approaches in this work use the human natural three-dimensional pattern matching to produce rational improvements to the intermediate solutions of MOEAs.

4.2 Algorithms

Problems with a large optimal frontier cause extreme difficulties for the DMs to make decisions. Due to the great number of solutions in the trade-off set, the DM has to identify those points that satisfy his preferences. Hence, there is an issue regarding the computational resources necessary to generate the complete trade-off set, but also there is a concern to choose one of the available solutions from the Pareto-optimal front.

The articulation of preferences during the optimization phase of MOEAs helps to retrieve the DM's expected solutions at the end of the process. Instead of computing the whole front, the preferences can bias the search to a small sub-set within relevant regions of DM interest.

Considering many different approaches to represent the DM preferences in the evolution process, the reference point is more suitable for driving the optimization on areas previously selected by the user. Reference points are simple to understand and operate. When a point is defined in the objective space, the closest region in P_F to the reference point is the preferred region. The reference points can be applied before the searching process (*a priori*) or interactively.

The *a priori* reference point requires a previous knowledge of the problem being solved, which is impractical in many situations. On the other hand, the interactive approach allows constant adjustments in the reference point during the optimization. It calls for a lower degree of cognitive complexity because the DMs can improve their knowledge about the problem and review their preferences at each interaction with the evolution mechanism.

For that reason, the new algorithms are extensions of the classical MOEAs: NSGA-II [64], SPEA2 [179] and SMS-EMOA [28]. The main changes on the original methods are the incorporation of COIN as an operator; the transformation of the continuous evolutionary process into an interactive one; and the adoption of collective reference points to drive the search towards relevant regions in Pareto-optimal front.

All the references points are discovered online with the support of a genuine collective intelligence of many users. The algorithms allow the DMs to choose multiple reference points simultaneously whether they are feasible (deducible from a solution vector) or infeasible points.

4.2.1 CI-NSGA-II

The NSGA-II [64] is a non-domination based genetic algorithm for multi-objective optimization. It adopts two main concepts: a density information for diversity and a fast non-dominated sorting in the population. The crowding distance uses the size of the largest cuboid enclosing two neighbouring solutions to estimate the density of points in the front. Solutions with higher values of this measure are preferred rather than points in a more crowded region (smaller values) because they are better contributors to a uniformly spread-out Pareto-optimal front. The non-dominated sorting places each individual into a specific front such that the first front τ_1 is a non-dominant set, the second front τ_2 is dominated only by the individuals in τ_1 and so on. Each solution inside the front τ_n receives a rank equal to its non-domination level n. The selection operator prefers a minor domination rank (i_{rank}) and higher values of crowding distance (i_{dist}).

The selection operator uses the rank (i_{rank}) and crowding distance (i_{dist}) in a binary tournament. The partial order $\prec_c$ between two individuals i and j, for example, prefers the minor domination rank if they are from different fronts or otherwise, the one with higher values of crowding distance. Then, crossover and mutation are applied to generate an offspring population.

$$i \prec_c j := i_{\text{rank}} < j_{\text{rank}} \vee (i_{\text{rank}} = j_{\text{rank}} \wedge i_{\text{dist}} > j_{\text{dist}}) \tag{4.1}$$

In algorithm 4.1, the new CI-NSGA-II converts the original NSGA-II into an interactive process. The variable *MAXgeneration* receives the maximum number of generations for evolution. The variable *block*, used in the inner while loop, represents the iteration interval which means the number of generations to run without external user interruption.

Until the first interaction step with the participants through the *CollectiveContributions()* procedure, the algorithm uses the standard crowding distance in order to come up with a good spread of the solutions. After the *CollectiveContributions()* procedure, however, the COIN operator (*COIN Selection()*) starts focusing on preferred areas of the search space based on the user contributions.

Algorithm 4.1 The Collective Intelligence NSGA-II.

$generation \leftarrow MAXgeneration$
$block \leftarrow SUBSETgeneration$
while $i < generation$ **do**
  while $block$ **do**
    $offspring \leftarrow$ **Tournament**(pop)
    $offspring \leftarrow$ **Crossover**$(offspring)$
    $offspring \leftarrow$ **Mutation**$(offspring)$
    $pop \leftarrow$ **COIN Selection**$(pop + offspring)$
    $i{+}{+}$
  end while
  $front \leftarrow$ **PF**(pop)
  $contributions \leftarrow$ **CollectiveContributions**$(front)$
  $pop \leftarrow contributions$
  $\Theta \leftarrow$ **ExpectationMaximization**$(contributions)$
  $pop \leftarrow$ **ReferencePointDistance**(pop, Θ)
end while

After crossover and mutation, the subroutine *CollectiveContributions()* suspends the evolution progress and submits some individuals from the population to the users' evaluation. The objective here is to address some of the MOEA intermediate solutions to be modified and improved by the collective intelligence participants. The modified solutions are then re-introduced to the population and, therefore, follow the remaining steps of the evolutionary process.

This approach allows members of the collective to interact with and modify the solutions. As presented in Section 4.1, the collective intelligence are engaged in two different manners: a selection operator that compares the individuals and chooses the best candidate, or a variation operator that improves current individuals from the population. Both approaches discover online collective reference points with the support of a genuine collective intelligence of many users. But, the variation operator offers the users a chance for rational improvement on the individuals and the evolution process.

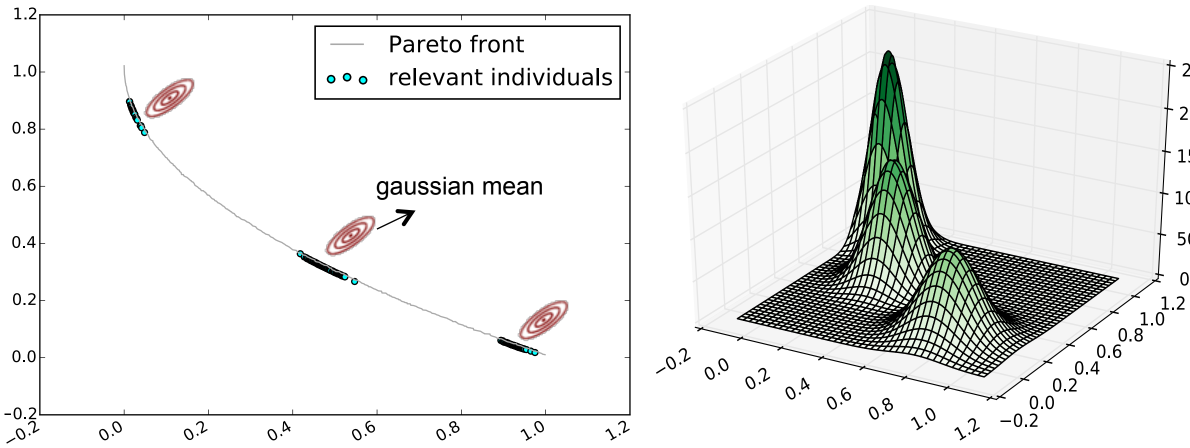


Figure 4.1: Three online collective reference points and their Gaussian distribution applied to the ZDT1 test.

Enhanced by collective subjectivity and cognition, the successive stages of evolution are improved via group's preferences in a direct crowdsourcing fashion. It allows the collective to act as DMs to choose multiple reference points simultaneously.

Inside collective environment, the contributions come from different people. Assuming the Central Limit Theorem [84], the inputs have a distribution that is approximately Gaussian. Therefore, after each collective interaction, the subroutine *ExpectationMaximization()* gets the users' collaboration as a Gaussian Mixture model to emulate the evaluation landscape of all participants' preferences. It discovers a pattern of rational preference among the participants' contributions that leads the exploration of areas of interest.

The expectation maximization approach (see Chapter 2) creates online reference points (Θ) for search optimization. Whether the user's collaboration is a simple vote on the best individual presented to him (pairwise comparison) or a complete re-edited individual, the clustering algorithm distinguishes the users with similar preferences to perform a cooperative evolution and a decision making choice through the collective reference points. Figure 4.1 shows an example of three online reference points and the Gaussian distribution of their points from the well-known ZDT1 test suite [178].

Finally, the procedure *ReferencePointDistance()* calculates the minimum distance from each point in the population to the nearest collective reference points in Θ. This way, the point near the reference point is favoured and stored in the new population. CI-NSGA-II develops a partial order similar to the NSGA-II procedure, but replaces the crowding distance operator by the distance to collective reference points (i_{ref}). The partial order $\prec_c$ between two individuals i and j, for example, prefers the minor domination rank if they are from different fronts or, otherwise, the one with lower values of reference point distance.

$$i \prec_c j := i_{\text{rank}} < j_{\text{rank}} \vee (i_{\text{rank}} = j_{\text{rank}} \wedge i_{\text{ref}} < j_{\text{ref}}) \tag{4.2}$$

This algorithm performs the *COIN Selection()* operation based on the new partial order. Like NSGA-II, individuals with minor domination rank are preferred. But if they belong to the same front, the one with the closest reference point distance is used instead. Considering the same population P and the partial order method, Figure 4.2 illustrates the individuals from NSGA-II and CI-NSGA-II last front for ZDT1 problem with one reference point. The green dots represent individuals selected by the partial order rule and the rejected individuals are in red. It is clear that the former algorithm chooses a more diverse distribution of individuals, whereas the latter concentrates on the solutions closest to the reference points and tends to reject the more distant ones.

CI-NSGA-II prioritizes the points close to the online collective reference point. The algorithm consumes preference information to explore satisfactory solutions for DMs.

4.2.2 CI-SMS-EMOA

The SMS-EMOA [28] is a steady-state algorithm that applies the non-dominated sorting as a ranking criterion and the hypervolume performance measure (S) as a selection operator.

After the non-domination ranking, the next step is to update the last front population, P_{worst}. It replaces the member with the minimum contribution to P_{worst} hypervolume by a new individual that increases the hypervolume covered by the population.

In algorithm 4.2, the new CI-SMS-EMOA converts the original SMS-EMOA into an interactive process. The *CollectiveContributions()* and *ExpectationMaximization()* subroutines have the same purpose and work as the CI-NSGA-II. In the *COIN Selection()* operation, individuals with minor domination rank (i_{rank}) are preferred. If they belong to the same front, the one with the maximum contribution to the hypervolume of the set and the closest reference point distance (i_{ref}) is selected.

The procedure *Hype-RefPoint Distance()* gets the hypervolume contribution (S) and calculates the minimum distance from each solution in the population to the nearest collective reference points in Θ. This way, the solution with high hypervolume values and short reference point distance is favoured and stored in the new population.

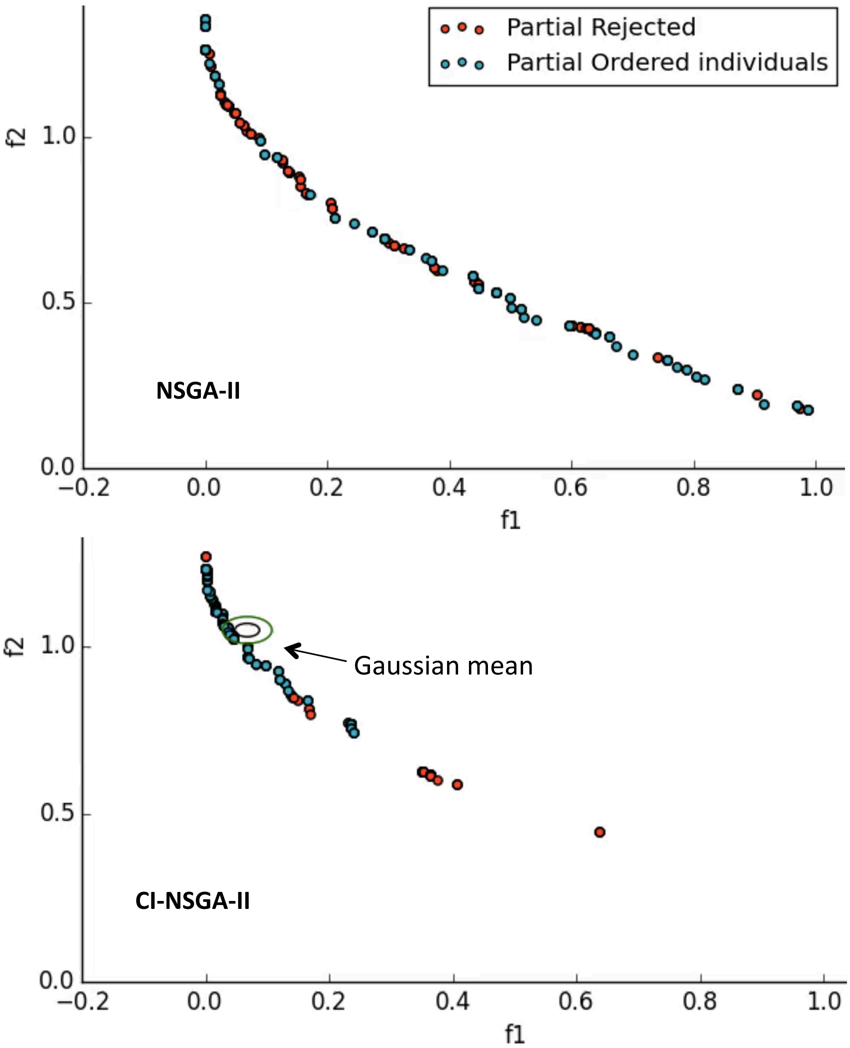


Figure 4.2: NSGA-II and CI-NSGA-II last front selection, respectively. The green dots represent individuals selected by the partial order rule and the red ones are those rejected.

4.2.3 CI-SPEA2

The strength Pareto evolutionary algorithm 2 (SPEA2) [179] developed a fitness assignment strategy based on the number of individuals one solution dominates and it is dominated by. SPEA2 implements elitism by keeping an external population (archive) of size N. The archive preserves the best solutions since the beginning of the evolution.

The strength $ST(i)$ for each individual i is the number of population members it dominates: $ST(i) = |\{j : j \in P_t \oplus \bar{P}_t \wedge i \prec j\}|$; where $\oplus$ is the multiset union, P_t and $\bar{P}_t$ are the population and archive population at generation t, respectively. The fitness $F(i)$ for a individual i is given by the strength of its dominators: $F(i) = \sum ST(j)$; where $j \in P_t \vee \bar{P}_t$, $j \prec i$. High values of $F(i)$ means the individual i is dominated by many others and $F(i) = 0$ corresponds to a non-dominated individual.

Algorithm 4.2 The Collective Intelligence SMS-EMOA.

$generation \leftarrow MAXgeneration$
$block \leftarrow SUBSETgeneration$
while $i < generation$ **do**
  while $block$ **do**
    $offspring \leftarrow$ **Tournament**(pop)
    $offspring \leftarrow$ **Crossover**$(offspring)$
    $offspring \leftarrow$ **Mutation**$(offspring)$
    $pop \leftarrow$ **COIN Selection**$(pop + offspring)$
    $i{+}{+}$
  end while
  $front \leftarrow$ **PF**(pop)
  $contributions \leftarrow$ **CollectiveContributions**$(front)$
  $pop \leftarrow contributions$
  $\Theta \leftarrow$ **ExpectationMaximization**$(contributions)$
  $pop \leftarrow$ **Hyper-RefPoint Distance**(pop, Θ, S)
end while

SPEA2 uses a nearest density estimation technique, adapted from the kNN method [145], to distinguish individuals having the same fitness values. This density function is a function of the distance to the k-th nearest data point and it is added to the fitness function F.

After fitness assignment, the algorithm implements an archive truncation method to guarantee a good spread of non-dominated solutions. All the non-dominated solutions are stored in $\bar{P}_t$. If the archive is too small ($\mid \bar{P}_t \mid \leq N$), then the $N- \mid \bar{P}_t \mid$ dominated individuals are selected according to their fitness values F. In turn, if the archive is too large the individuals with the worst density are removed from $\bar{P}$.

In the new CI-SPEA2 (algorithm 4.3), the subroutine *COIN Selection()* computes the strength of all individuals and the non-dominated members are copied to the archive $\bar{P}_t$. The k-th nearest data point used to calculate the original density function in SPEA2 was substituted by the collective reference points Θ. If the archive $\mid \bar{P}_t \mid \leq N$, the algorithm chooses the nearest individuals to the collective reference point until the archive size is reached. Otherwise, if $\mid \bar{P}_t \mid > N$, it removes the more distant ones proportionally to the number of individuals in each reference point cluster. This way, the archive keeps the same distribution of points around its reference points.

The procedure *RefPointDistance()* sets the minimum distance between each solution in the population $(P_t + \bar{P}_t)$ and the collective reference points in Θ.

Algorithm 4.3 The Collective Intelligence SPEA2.

$generation \leftarrow MAXgeneration$
$block \leftarrow SUBSETgeneration$
while $i < generation$ **do**
  while $block$ **do**
    $archive \leftarrow$ **COIN Selection**$(pop + archive)$
    $offspring \leftarrow$ **Tournament**$(archive)$
    $offspring \leftarrow$ **Crossover**$(offspring)$
    $offspring \leftarrow$ **Mutation**$(offspring)$
    $pop \leftarrow offspring$
    $i++$
  end while
  $front \leftarrow$ **PF**(pop)
  $contributions \leftarrow$ **CollectiveContributions**$(front)$
  $pop \leftarrow contributions$
  $\Theta \leftarrow$ **ExpectationMaximization**$(contributions)$
  $pop, archive \leftarrow$ **RefPoint Distance**$(pop, archive, \Theta)$
end while

4.2.4 Projection of Reference Points

The necessity of multiple human interactions to address the collective reference points may cause user fatigation. In addition to that, the participants may not be available to take part on each interruption of the evolution process. The question of how to alleviate the users' burden or assistance in the new algorithms is important.

In practice, a distinct approach can be included into the three algorithms proposed: the projection of the reference points. At each generation t, the reference points can be repositioned based on the current approximation set S. The approach maps the trends of user preferences and automatically selects new reference points without the need for a user's interaction.

The proposed algorithms are extended with the reference point projection approach. The new versions of the algorithms are named as: CI-NSGA-II-P, CI-SMS-EMOA-P and CI-SPEA2-P. They reduce the user burden in interactive MOEAs to create the collective reference points and keep finding the most preferred solutions.

Let P_t be the population at generation G_t with size N and Q_t be the offspring. The new approximation set S_t for this generation t are clustered closely around the J collective reference points $\boldsymbol{z}_j$ ($j \in 1, \ldots, J$). The points are clustered based on the closest distance to one of the collective reference points and associated to them.

For each cluster C_j, the ideal point is determined by identifying the array with the

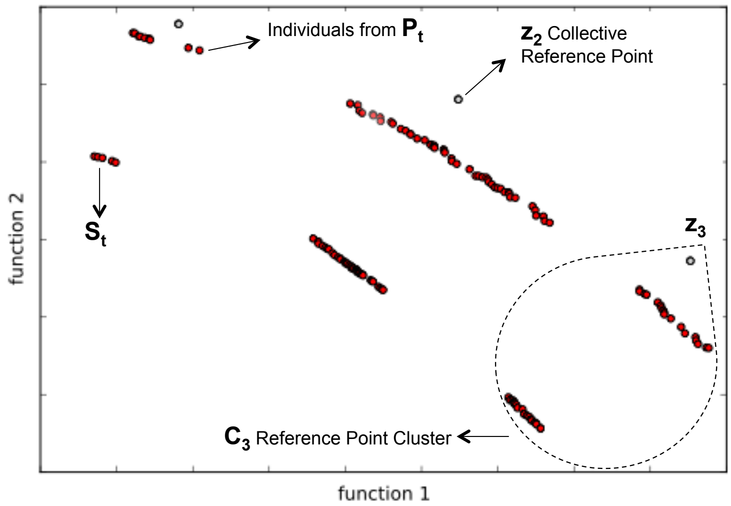


Figure 4.3: Reference point and its corresponding cluster

lower bound of all objective function z_j^*. Then, a reference line is defined by joining the collective reference point z_j with the ideal point z_j^*. The perpendicular distance of each point in C_j from the corresponding reference line is calculated. In two-dimension, the distance of a point (x_0, y_0) from the line that passes through two points $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ is defined as follows:

$$\text{distance}(P_1, P_2, (x_0, y_0)) = \frac{|(y_2 - y_1)x_0 - (x_2 - x_1)y_0 + x_2y_1 - y_2x_1|}{\sqrt{(y_2 - y_1)^2 + (x_2 - x_1)^2}} \tag{4.3}$$

The solution with the closest perpendicular distance is named representative-point. Finally, the coordinate of the point in the reference line which is closest to the representative-point become the new reference point to the generation G_{t+1}. Figure 4.3 illustrates a reference point z_3 and its cluster C_3 of individuals in S_t. Figure 4.4 shows the reference line for three different clusters and ideal points.

4.2.5 Complexity of the Algorithms

The new algorithms are based on the classical MOEAs: NSGA-II, SPEA2, and SMS-EMOA. The overall worst-case complexity of the NSGA-II is $\mathcal{O}(MN^2)$ [64], where M is the number of dimensions and N is the population size. The SPEA2 has the worst runtime complexity of the truncation operator: $\mathcal{O}(K^3)$, where $\bar{N}$ is the size of the archive and $K = N + \bar{N}$. But on average the SPEA2 has the complexity of $\mathcal{O}(K^2 \log K)$ [179]. The bottleneck of SMS-EMOA is the high time complexity for computing the values of the hypervolume. In a two-dimensional space the complexity is $\mathcal{O}(N^2)$. But in the case

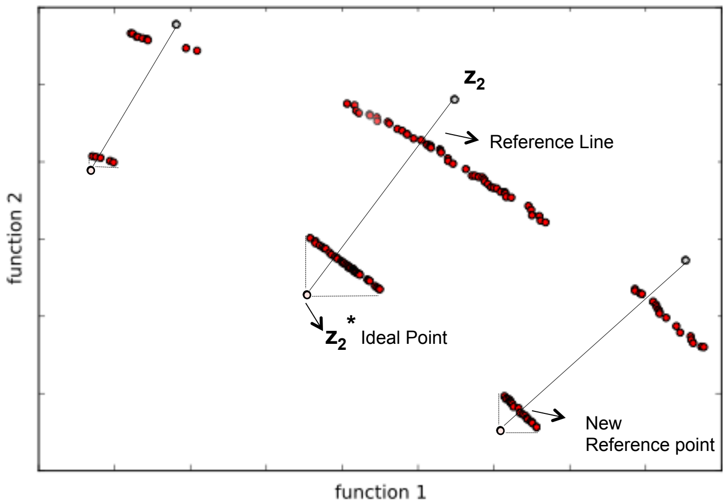


Figure 4.4: Reference line for three different clusters and ideal points

of high-dimensional MOPs, the complexity grows exponentially: $\mathcal{O}(N^{M+1})$ [77].

The algorithms proposed in this work: CI-NSGA-II; CI-SPEA2; CI-SMS-EMOA; CI-NSGA-II-P; CI-SPEA2-P; CI-SMS-EMOA-P do not increase the asymptotic complexity of the original algorithms. The Gaussian Mixture procedure in the new algorithms, used to reveal distributions of observations in the overall population and determine the collective reference point, has the complexity of $\mathcal{O}(MN+GN^2)$, where G is the number of mixtures.

Other operations have smaller complexity. The ideal point and the association of the points to a cluster C_j on the reference point projection for the algorithms: CI-NSGA-II-P; CI-SMS-EMOA-P; CI-SPEA2-P; requires a total of $\mathcal{O}(MN)$ and $\mathcal{O}(MNH)$ computations, respectively.

Chapter 5

Assessing Performance in Preference-Based MOEAs

5.1 Multi-Objective Performance Indicators

The multi-objective algorithms have two main goals with respect to the optimization process: a) a good convergence to the Pareto-optimal front; b) a good diversity in obtained points, even when preferred areas are appointed instead of the whole P_F. In the last decades, several metrics were implemented to evaluate the outcome sets of MOEAs and used to compare the performance of different evolutionary approaches. The analysis of two sets of trade-off solutions requires an appropriate study of the performance indicators available.

There are a variety of approaches that analyse the distribution of points in the objective function space and the accuracy in terms of convergence. Considering an approximation set S, the performance indicators can be grouped as cardinality, accuracy and diversity metrics [101, 138]. The cardinality of S refers to the number of non-dominated solutions that exists in S. Accuracy metrics evaluate the convergence or closeness of S to the known Pareto-optimal front. Finally, the diversity metrics exam the distribution and extent of the approximation set S.

A performance indicator is said unary if it receives as parameter only one approximation set S to be evaluated. Formally, an unary metric is denoted as: $I(S) : T \to \mathbb{R}$, where T is the set of all approximation sets. If the indicator receives as parameter two approximation sets, S_0 and S_1, the metric is said to be binary and denoted as $I(S_0, S_1) : T^2 \to \mathbb{R}$.

It is worth noting that unary metrics assign a quality value to an approximation set, while binary metrics take into account the relationship between two approximation sets

(S_0, S_1) in terms of dominance to discover which one is better. For that reason, the unary quality indicators are most commonly used in the literature [182].

5.1.1 Cardinality Indicators

The cardinality indicators quantify the number or ratio of non-dominated solutions in S.

Usually, these indicators are kept as the prerequisite ahead of the distribution or accuracy metrics. Because if the number of non-dominated solutions in any two approximation sets (S, S') for comparison differs or is too small in size, the distribution and accuracy indicators would become statistically insignificant.

The optimal solution (P_S) set obtained by the optimizers is termed as A. A finite number of non-dominated solutions that approximates the true P_F is termed as B.

5.1.1.1 Overall Non-dominated Vector Generation

The Overall Non-dominated Vector Generation (ONVG) gives the number of the non-dominated solutions in the optimal solution set (A) [157]:

$$ONVG(A) = |A| \tag{5.1}$$

The Overall Non-dominated Vector Generation Ratio (ONVGR) gives the ratio of the optimal solution set (A) with respect to the true Pareto-optimal front (B):

$$ONVGR(A, B) = \frac{|A|}{|B|} \tag{5.2}$$

5.1.1.2 Error Ratio

Similar to the ONVGR indicator, the Error Ratio (ER) considered the solution intersections between A and B instead [156]. It is defined as:

$$ER(A, B) = 1 - \frac{|A| \cap |B|}{|B|} \tag{5.3}$$

5.1.1.3 Ratio of Non-dominated Individuals

The Ratio of Non-dominated Individuals (RNI) verifies the proportion of non-dominated individuals in the approximation set S [150]:

$$RNI(S) = \frac{|\bar{S}|}{n}, \tag{5.4}$$

where $\bar{S}$ denotes the set of non-dominated individuals in population S and n is the number of point in S.

5.1.1.4 Coverage of Two Sets Indicator

The Coverage of Two Sets [180, 181] is another measure widely used:

$$C(X', X'') = \frac{|\{\boldsymbol{a'} \in X''; \exists \boldsymbol{a'} \in X' : \boldsymbol{a'} \prec \boldsymbol{a''}\}|}{|X''|}, \tag{5.5}$$

where X', $X'' \subseteq X$ are two sets of decision vectors and function C maps the percentage of domination from one set to another in the interval $[0, 1]$. But it does not express how much better one set is over the other. Although convex regions may be preferred by the hypervolume indicator, the coverage of two sets technique has no restriction related to the shape of Pareto front.

5.1.2 Accuracy Indicators

The accuracy metrics evaluate the convergence or the proximity of S to the known Pareto-optimal front.

5.1.2.1 Pareto-optimal Front Coverage Indicator

Pareto-optimal Front Coverage indicator, $D_{S \to P_F}$, is a proximity indicator [34] that defines the distance between an achieved approximation set S and their closest counterpart in the current Pareto-optimal front:

$$D_{S \to P_F}(S) = \frac{1}{|S|} \sum_{\boldsymbol{x} \in S} \min_{\boldsymbol{x'} \in P_S} \{d(\boldsymbol{x}, \boldsymbol{x'})\}, \tag{5.6}$$

where d is the Euclidean distance between two points. It is also known as Generational

Distance (GD). If the Pareto-optimal front is continuous, a correct formulation of this indicator calls for a line integration over S. Small values of $D_{S \to P_F}$ indicate proximity to the Pareto-optimal front. This measure of quality does not necessarily represent a diverse solution, because it describes on average how far away the elements of S are from the trade-off front P_F. Its main drawback is the obligation to previously know the true front, what may be unfeasible for real applications.

5.1.2.2 ϵ-Indicator

Epsilon indicators rely on the epsilon dominance concept. In this performance, ϵ defines the value required to translate/scale the optimal solution set P_S such that P_S dominates P_F [107, 182]. It takes the form of:

$$I_{\epsilon+}(A, B) = \inf_{\epsilon \in \mathbb{R}} \{\forall \boldsymbol{y} \in B, \exists \boldsymbol{x} \in A \text{ such that } \boldsymbol{x} \preccurlyeq \epsilon + \boldsymbol{y}\} \tag{5.7}$$

5.1.2.3 Maximum Pareto Front Error

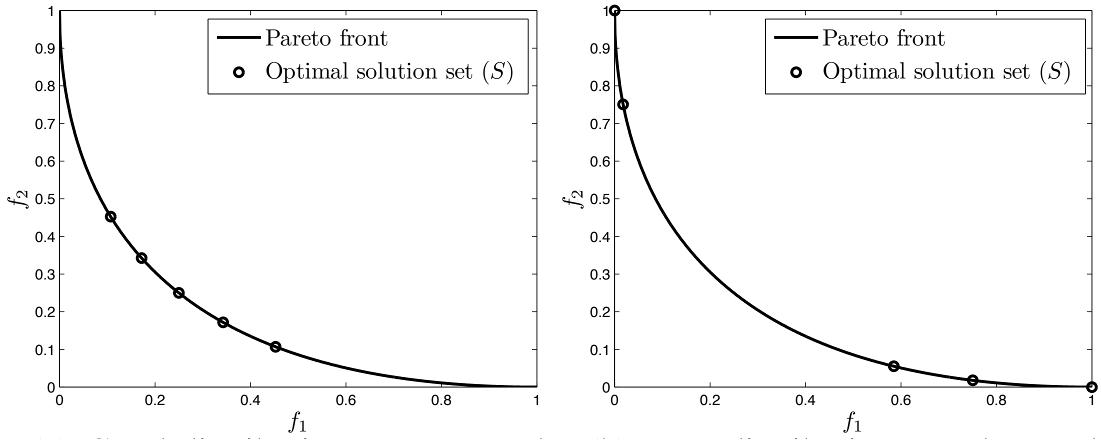
The Maximum Pareto Front Error (MPFE) evaluates the largest distance in the objective space between any individual $\boldsymbol{x_i}$ in the approximation front and the corresponding closest vector $\boldsymbol{x_j}$ in the true Pareto front P_F [156, 171].

$$MPFE(S) = \max_i d_i, \tag{5.8}$$

where $d_i = \min_j \|f(\boldsymbol{x}_i) - P_F(\boldsymbol{x}_j)\|$ is the distance in objective space between individual $\boldsymbol{x_i}$ and the nearest member in P_F.

5.1.3 Diversity Indicators

Diversity in the pool of final solutions guarantees good representations of alternatives to decision makers. The diversity indicators focus on the distribution and spread of the solutions. Figure 5.1 illustrates the the difference between distribution and spread. The points in 5.1a are well distributed but they have a poor spread, because the approximation set S does not contain the extreme points (0,1),(1,0) of the P_F. Figure 5.1b, on the other hand, have a good spread but poor distribution.



(a) Good distribution, poor spread. (b) Poor distribution, good spread.

Figure 5.1: Distribution and spread of the solutions [101].

5.1.3.1 Spread

The spread indicator (Δ) [49, 60] provides information related to the extent of the spread of the obtained Pareto front. It is defined as follows:

$$\Delta = \frac{\sum_{m=1}^{M} d_m^e + \sum_{i=1}^{|S|} |d_i - \bar{d}|}{\sum_{m=1}^{M} d_m^e + |S|\bar{d}}, \tag{5.9}$$

where d_i is a neighbouring distance measure, $\bar{d}$ is the mean value of this distance measure, d_m^e is the distance between the extreme solutions of S, $|S|$ is the quantity of the obtained non-dominated objective vectors. The smaller the value is, the more diverse is the front.

5.1.3.2 Maximum Spread Indicator

This indicator measures the length of the diagonal of a hyperbox formed by the extreme function values found in the approximation set [176]:

$$D = \sqrt{\sum_{m=1}^{M} \left(\max_{i=1}^{|S|} f_m^i - \min_{i=1}^{|S|} f_m^i \right)^2} \tag{5.10}$$

5.1.3.3 Spacing

The spacing indicator analyses the distribution of the obtained non-dominated set. It measures the spacing with a relative distance measure between the consecutive solutions in S [60]:

$$SP = \sqrt{\frac{1}{|S|}\sum_{i=1}^{|S|}\left(d_i - \bar{d}\right)^2} \tag{5.11}$$

where $d_i = min_{k\in S\wedge k\neq i}\sum_{m=1}^{M}|f_m^i - f_m^k|$ and $\bar{d}$ is the mean value $\bar{d} = \sum_{i=1}^{|S|} d_i/|S|$.

5.1.3.4 Uniform Distribution

The Uniform Distribution (UD) measures the distribution of non-dominated points on the trade-off set S [150]:

$$UD(S) = \frac{1}{1 + D_{nc}}, \tag{5.12}$$

where $D_{nc} = \sqrt{\sum_{\boldsymbol{x}_i\in S}\left(nc(\boldsymbol{x}_i) - \bar{nc}(\bar{\boldsymbol{x}})\right)^2 / \left(|S|-1\right)}$ is the standard deviation of niche count of the overall set of non-dominated individuals in S, $nc(\boldsymbol{x}_i) = |\{\boldsymbol{x}_j \in S : \|\boldsymbol{x}_i - \boldsymbol{x}_j\| < \alpha\}| - 1$ and $\bar{nc}(\bar{\boldsymbol{x}})$ is the average of $nc(\boldsymbol{x}_i)$.

5.1.4 Accuracy-Diversity Indicators

Accuracy-Diversity indicators measure both the convergence and diversity of S on a single scale.

5.1.4.1 Hypervolume Indicator

The Hypervolume (HV) or S-metric indicator [75, 181] calculates the volume of the union of hypercubes a_i defined by a non-dominated point m_i and a reference point x_{ref} defined as:

$$\begin{aligned} S(M) &= \Lambda(\{\textstyle\bigcup_i a_i | m_i \in M\}) \\ &= \Lambda(\textstyle\bigcup_{m\in M}\{x | m \prec x \prec x_{ref}\}) \,. \end{aligned} \tag{5.13}$$

It is a quantitative metric that computes the region space covered by all non-dominated points. This performance indicator can be used independently to evaluate the efficiency of different multi-objective algorithms and does not require knowledge of the true Pareto-optimal front on beforehand, which is an advantage for real-world problems rather than

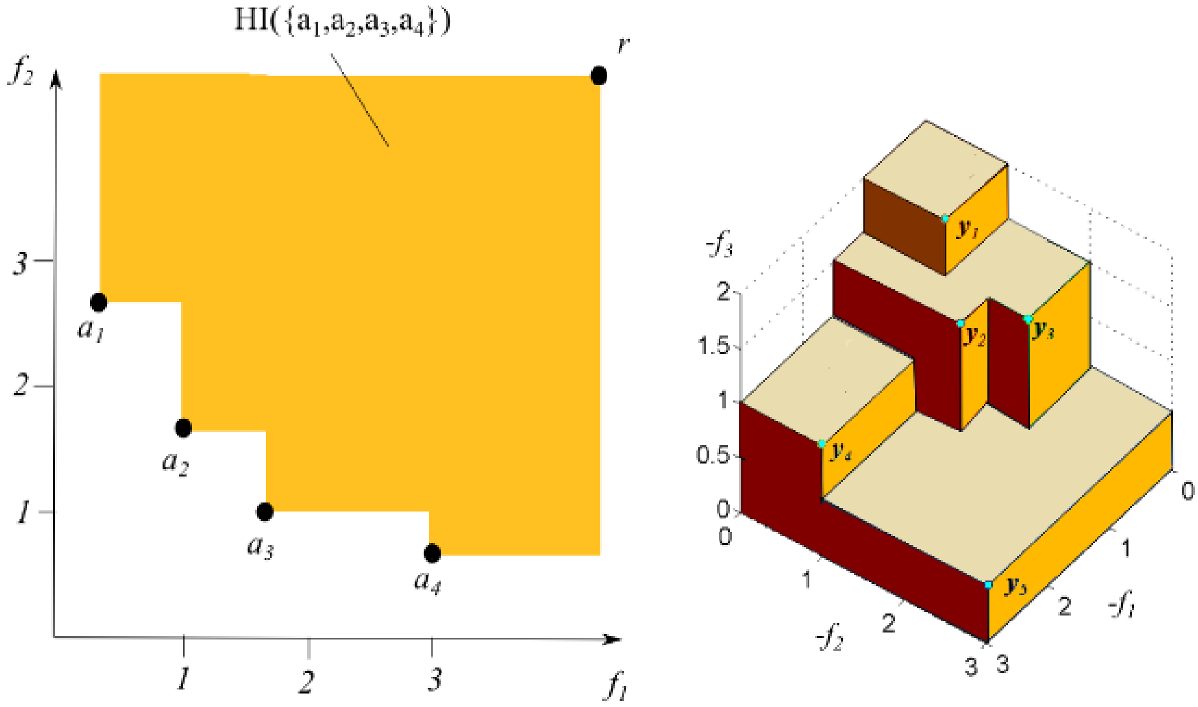


Figure 5.2: Dominated Hypervolume in two and three dimensions. The left figure shows the covered area of a minimization case whereas the right one shows the covered volume of a maximization case. Figure taken from [55].

synthetic ones like ZDT test suite [178], DTLZ [67] or WFG [94].

Figure 5.2 shows the dominated Hypervolume in two and three dimensions for a set $A = \{a_1, \ldots, a_4\} \subset \mathbb{R}^2$ (minimization case) and $Y = \{y_1, \ldots, y_5\} \subset \mathbb{R}^3$ (maximization case), respectively.

The understanding of performance concerns not only the quality of the approximation set S in terms of diversity and convergence. But also includes the computational resources needed to generate this outcome. Due to the the high computational complexity of the Hypervolume, the runtime of this indicator is exponential and become intractable when the number of objectives is large ($m \geqslant 4$). Some studies have been addressed to reduce the computational burden by estimating the Hypervolume indicator with Monte-Carlo sampling [13, 177].

5.1.4.2 Averaged Hausdorff Distance Indicator

The Averaged Hausdorff Distance indicator or Δ_p [121, 139, 142] is a metric that analysis both the proximity to the true Pareto front and the distribution of points along it. Δ_p combines variations of two performance indicators already described in this chapter: Generational Distance [34] and Inverted Generational Distance [57]. Let $A, B \subset \mathbb{R}^M$ be non-empty finite sets, the Δ_p value for $p > 0$ is termed the averaged Hausdorff distance between sets A and B. It takes the form of:

$$\begin{aligned} \Delta_p(A,B) &= \max\left(GD_P\left(A,B\right), IGD_P\left(A,B\right)\right) with \\ GD_P(A,B) &= \left(\frac{1}{|A|}\sum_{\boldsymbol{x}\in A} d(\boldsymbol{x},B)^P\right)^{1/P} \quad and \\ IGD_P(A,B) &= \left(\frac{1}{|B|}\sum_{\boldsymbol{y}\in B} d(\boldsymbol{y},A)^P\right)^{1/P}, \end{aligned} \tag{5.14}$$

where $d(\boldsymbol{u},A) = \inf\{\|\boldsymbol{u}-\boldsymbol{v}\| : \boldsymbol{v}\in A\}$ for $\boldsymbol{u},\boldsymbol{v}\in\mathbb{R}^M$ and some vector norm $\|.\|$.

5.1.4.3 Inverted Generational Distance

The Inverted Generational Distance (IGD) measures both convergence and diversity of the approximation set S [57]. It is defined as:

$$IGD(S) = \frac{\sum\limits_{\boldsymbol{v}\in B} d(\boldsymbol{v},\bar{S})}{|B|}, \tag{5.15}$$

where $\bar{S}$ denotes the set of non-dominated individuals in population S, $d(\boldsymbol{v},\bar{S})$ represents the minimum Euclidean distance between $\boldsymbol{v}$ and the points in $\bar{S}$.

5.2 Performance Indicators for Preference-Based MOEAs

The performance indicators evaluate the quality of different results and allow comparison between the algorithms. However, the current state-of-the-art diversity indicators presented in the previous section cannot be employed in this study. That is because their computation depends on the spread of solutions in the whole Pareto front and, on contrary, the proposed preference-based algorithms here aim to obtain subsets of solutions close to the collective reference points. In this case, as the goal is not the spread and distribution of points along the trade-off set, the Hypervolume, Spread and Spacing indicator are inadequate.

Figure 5.5 illustrates a shortcoming of Inverted Generational Distance and Hypervolume metrics [112] when these indicators are applied to a trade-off set with preference

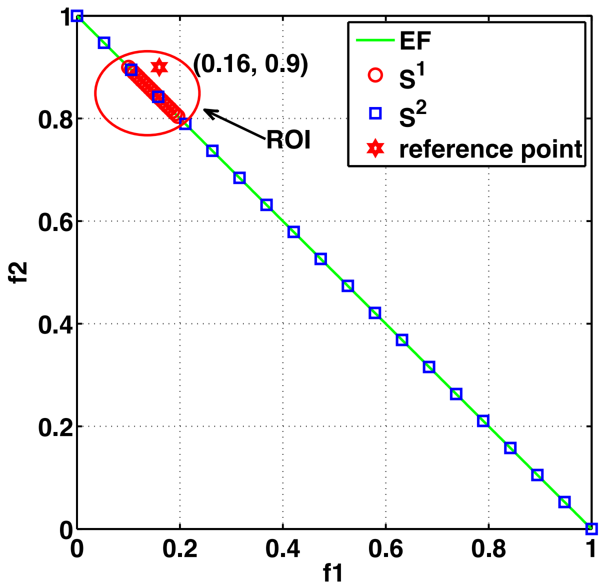


Figure 5.3: A shortcoming of IGD and HV indicators. S^2 has higher values of IGD and HV. But S^1 should be preferable than S^2 because their points are closer to the reference point $\boldsymbol{z}^0$ provided. Figure taken from [112].

information. Considering the function $f_2 = 1 - f_1$, the example shows the efficient front (EF) as a line having a intercept of 1.0 with each objective axis. The reference point $\boldsymbol{z}^0$ is $(0.16, 0.9)$. Two sets of points S^1 and S^2 have the same cardinality ($|S^1| = |S^2| = 20$). The approximation set S^1 is concentrated around $\boldsymbol{z}^0$ and S^2 is evenly distributed along the whole Pareto front. Due to the spread of S^2, its IGD and HV have higher values ($IGD(S^1) = 3.476E - 1; IGD(S^2) = 4.610E - 4; HV(S^1) = 0.2910; HV(S^2) = 0.6837$). But based on the reference point $\boldsymbol{z}^0$ provided, S^1 should be preferable than S^2.

The accuracy indicators are one available alternative to measure the performance of preference-based MOEAs. But relying only on the convergence properties gives just an one-sided analysis of the outcomes. It would be important to explore different aspects of the solutions such as the concentration of points around the reference points.

There is a lack of performance indicators that focus only on the proportion of occupied area in P_F. Recent studies have focused great attention upon this limitation. The following subsections present some indicators based on preference areas and three new ones adopted for the work reported here.

5.2.1 Filatovas Spread

This spread indicator proposed by Filatovas [76] modifies the Spread indicator Δ [49, 60] by removing the distance between the extreme objective vectors of the approximation set

S. The formula is defined as follows:

$$\Delta_P = \frac{1}{P} \sum_{r=1}^{P} \left(\frac{1}{|S|\bar{d}} \sum_{i=1}^{|S|} |d_i - \bar{d}| \right), \tag{5.16}$$

where P is the number of non-empty clusters, d_i is a neighbouring distance measure, $\bar{d}$ is the mean value of this distance measure.

5.2.2 User-Preference Composite Front

The User-Preference metric based on a Composite Front (UPCF) is an indicator for evaluating the performance of preference-based MOEAs [128]. The Composite Front (CF) is a type of reference set [129], which is a collection of candidate solutions that comprises a preferred region based on the location of a user-supplied reference point.

In practice, the CF is the collection of all non-dominated solutions from the merged solution sets of all algorithms that are to be compared. After that, a preferred region is defined on the composite front based on the location of a provided reference point. The solution with the least distance to the reference point is called mid-point. Then the points within a range of r from the mid-point are considered to be in the preferred region.

Once the preferred region is defined, the Hypervolume or Inverted Generational Distance (IGD) can be applied to evaluate the results of each algorithm which are within the preferred region. In Figure 5.4, the squares and circles represent the solution sets for two different preference-based MOEAs. The points in black (whether squares or circles) are the non-dominated solutions and they form the composite front. The points in gray are the dominated ones. The parameter r determines the size of the preferred region around the mid-point.

5.2.3 Ratio of Non-Dominated Points

The Ratio of Non-Dominated Points measures the quantity of non-dominated points found in the approximation set S that belongs to a known reference set R [88]. It is defined as:

$$C1_R(S) = \frac{|S \cap R|}{|R|} \tag{5.17}$$

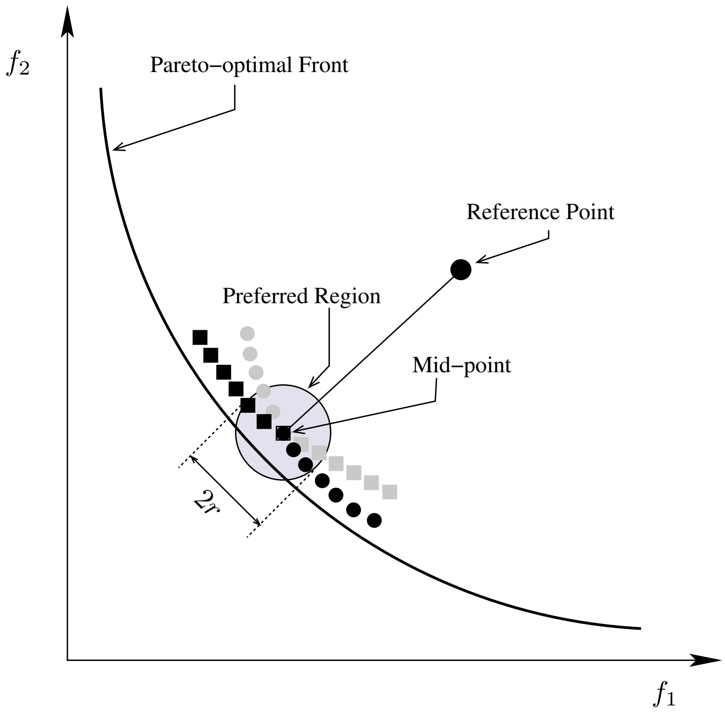


Figure 5.4: The composite front and its preferred region. Squares and circles in black represent the composite front. The parameter r determines the size of the preferred region around the mid-point. Figure taken from [128].

5.2.4 R-metric

Li and Deb [112] suggested an indicator to quantitatively evaluate the performance of a preference-based MOEA using reference points. The R-metric indicator pre-processes the preferred trade-off set according to a multi-criterion decision-making (MCDM) approach before applies the IGD or HV performance assessments.

In this method, three parameters are necessary: a) a reference point $\boldsymbol{z}^r$; b) a worst point $\boldsymbol{z}^w$; c) the extent of the preferred area, denoted as $\Delta(0 < \Delta \leq 1)$. First of all, the R-metric will merge all the L approximation sets $S^C = S^1, \ldots, S^L$ resulted from L different preference-based MOEAs. For each S^i, $i \in 1, \ldots, L$, only the non-dominated solutions, comparing to those in the composite set S^c, are retained.

Then, some points must be identified for further operation. The centroid of a given trade-off set S is denoted as $\boldsymbol{z}^c$. The representative point $\boldsymbol{z}^p$ is the closest point ($\boldsymbol{z}^p \in S^i$) to the centroid $\boldsymbol{z}^c$. The preferred area must have a delimited area Δ, so only the points inside this region are retained in S^i.

The final step computes the Achievement Scalarizing Function (ASF) [167] of $\boldsymbol{z}^p$ on the reference line connecting $\boldsymbol{z}^r$ and $\boldsymbol{z}^w$. The formula is defined as follows:

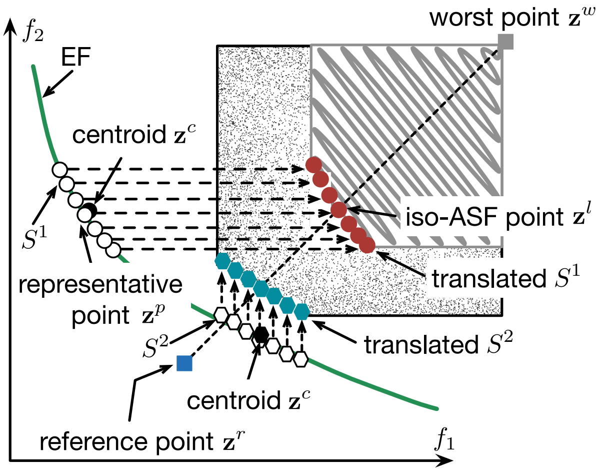


Figure 5.5: The R-metric solution translation of two sets of points (S^1 and S^2) with regard to $\boldsymbol{z}^r$ and $\boldsymbol{z}^w$. Figure taken from [112].

$$\begin{aligned} \text{minimize } ASF(\boldsymbol{x}|\boldsymbol{z}^r\boldsymbol{w}) = \max_{1\leq i\leq m} \frac{f_i(\boldsymbol{x}-\boldsymbol{z}_i^r)}{\boldsymbol{w}_i} \\ \text{subject to } \boldsymbol{x}\in\Omega, \end{aligned} \tag{5.18}$$

where $\boldsymbol{w}$ is the weight vector that implies the relative importance of objectives. Based on the ASF, each objective vector has a projection, called iso-ASF point, on the reference line. The iso-ASF point of the representative point $\boldsymbol{z}^p$ is denoted as $\boldsymbol{z}^l$. To conclude, all points from S^i are translated to the corresponding virtual position along the direction vector $\boldsymbol{z}^l - \boldsymbol{z}^p$ and a regular metrics can be applied for performance assessment. Figure 5.5 exhibits a solution translation for S^1 and S^2 with regard to $\boldsymbol{z}^r$ and $\boldsymbol{z}^w$.

5.2.5 Novel Performance Indicators for Preference-Based MOEAs

This work presents three new indicators based on preference areas.

5.2.5.1 Referential Cluster Variance Indicator

Instead of a good spread of solutions along P_F, the Referential Cluster Variance indicator κ, proposed in this work, wants to obtain subsets of solutions close to the collective reference point. In this context, a small cluster variance means the individuals from the sample $Y = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N\}$ are clustered closely around the population mean (μ) or the reference point ($\boldsymbol{z}^0$). A low dispersion for a group of preferred points in P_F denotes a better efficiency of the approach tested. The Referential Cluster Variance indicator κ is represented as follows:

$$\kappa = \frac{1}{N}\sum_{i=1}^{N}(\boldsymbol{y_i} - \mu)^2 \tag{5.19}$$

In cases with more than one collective reference point ($\boldsymbol{z}^j$), the points are clustered based on the closest distance to one of the reference points: $C_j = \{\boldsymbol{a} \in \mathbb{R}^k : \|\boldsymbol{a} - \boldsymbol{z}^j\| \leq \|\boldsymbol{a} - \boldsymbol{z}^i\|, \forall i\}$. Cluster C_j consists of all points for which $\boldsymbol{z}^j$ is the closest. The referential cluster variance is calculated to each cluster separately.

5.2.5.2 Convex Hull Volume Indicator

The convex hull of a set of points in k-dimensional space can be represented as a set of bounding facets and a collection of vertexes for each facet. A set C is convex if the line segment between any two points in C lies in C, i.e., for any $x_1, x_2 \in C$ and any Θ with $0 \leq \Theta \leq 1, \Theta x_1 + (1-\Theta)x_2 \in C$. The convex hull of C (*conv* C) for n points is then given by the expression:

$$conv\ C = \left\{\sum_{j=1}^{n}\lambda_j x_j : \lambda_j \geq 0\ for\ all\ j\ and\ \sum_{j=1}^{n}\lambda_j = 1\right\} \tag{5.20}$$

Convex hull is a well-known geometric object widely used in various fields such as shape analysis, pattern recognition, geographical information systems, image processing, etc. There are a few multi-objective evolutionary approaches designed to work with this geometric concept in the optimization.

The Normal Boundary Intersection method (NBI) [59] projects elements of the CHIM towards the boundary $\partial\mathcal{Z}$ of the objective space $\mathcal{Z}$ through a normal vector N. The intersection point between $\partial\mathcal{Z}$ and N the normal pointing is a Pareto optimal point, if the P_F surface is convex. Martínez and Coello [123,172] introduced an archiving strategy based on the CHIM to find evenly distributed points along the P_F. Their convex hull multi-objective evolutionary algorithm (CH-MOEA) uses an archiving mechanism that stores non-dominated solutions which are orthogonal to each point of CHIM ($h \in \mathcal{H}$). Likewise, Shan-Fan et al. [144]presented a MOEA where the non-dominated solutions are picked out from dominated solutions by the quick convex hulls algorithm.

Wang and Emmerich et al. proposed a convex hull-based multi-objective genetic programming (CH-MOGP) [161,162] that follows similar strategies than SMS-EMOA and

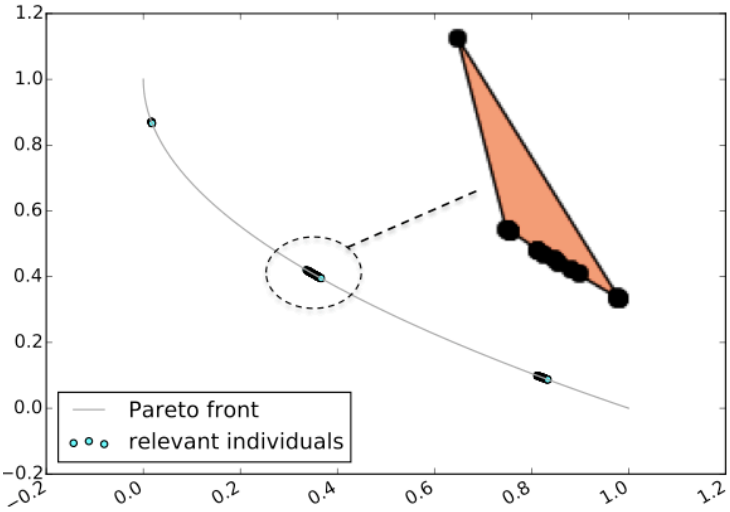
NSGA-II. But it uses convex hull-based sorting approach as an indicator based selection schema to rank the individuals into different levels. Monfared [130] also employs convex hull concepts to elaborate a geometric ranking procedure for non-dominated comparisons in NSGA-II. Another approach for classifiers comparisons brings the volume under the convex hull as a MOEA performance indicator [175].

The convex hull method can be extended to measure the quality of the non-dominated points in the desired region of interest. To best of the author's knowledge, there are no reports in the literature concerning the use of convex hull as a MOEA performance indicator. The idea behind this is to combine the points around each reference point to form a convex facet of the P_F preferred area. Thereafter, the volume of the convex hull is calculated and used as a scalar indicator for the distribution of points in P_F. Small values of the hull volume (Ψ) indicate concentrated points around the reference points.

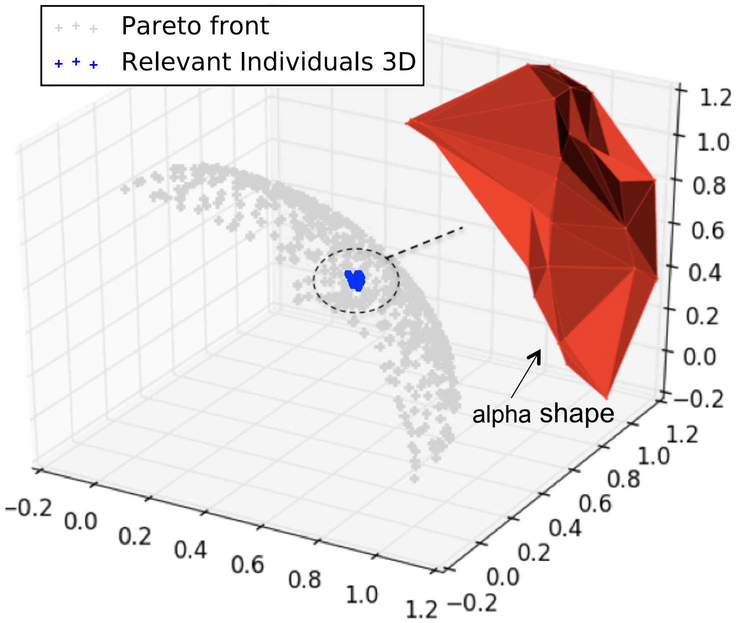
In order to compute the convex hull in the plane, some algorithms test all pair of points. These algorithms run in $\mathcal{O}(n^4)$ time. But there are several algorithms which attain the time complexity of $\mathcal{O}(n \log n)$. The *quickhull* method [18] uses a divide and conquer approach similar to *quicksort*. It has the average case complexity of $\mathcal{O}(n \log n)$, but may degenerate to $\mathcal{O}(n^2)$ in the worst case.

Non-convex problems can use alpha shapes [72] to determine a concave hull of their points in P_F. The alpha shape is a generalization of the convex hull and a subgraph of the Delaunay triangulation. As with the quickhull, the alpha shape of n points in the plane can be determined in time $\mathcal{O}(n \log n)$. The value of alpha (α) controls the geometric design of the shape. For large α values the shape approaches to the boundary of the convex hull. On the other hand, as α decreases the shape shows more cavities.

Figure 5.6 illustrates the convex and non-convex enclosure for the non-dominated points generated by the CI-NSGA-II algorithm. The test problems used were the ZDT1 [178] and DTLZ3 [67], respectively.



(a) ZDT1



(b) DTLZ3

Figure 5.6: Convex hull and alpha shape of non-dominated points after CI-NSGA-II iterations with one reference point.

5.2.5.3 Reference Set Distance

The Reference Set Distance ($C2_R$) indicates the mean distance between the non-dominated points from the approximation set S and the reference set R:

$$C2_R(S) = \sqrt{\left(\frac{1}{|S'|}\sum_{i=1}^{|S'|} d_i\right)^2} \tag{5.21}$$

where S' is the set of non-dominated points from S and d_i is the minimum distance between the i-th element in S' and the reference set R.

Chapter 6

Experimental Results

6.1 Multi-Objective Benchmark Problems

This section presents the results of tests with benchmark problems. As mentioned in Chapter 4, some traditional performance indicators (like: Hypervolume and Spread) do not evaluate subsets of solutions close to the collective reference points. For that reason, this work uses the new indicators to measure the quality of the algorithms: Referential Cluster Variance (κ), Convex Hull Volume (Ψ) and the Reference Set Distance ($C2_R$); along with the proximity-based Front Coverage indicator($D_{S \to P_F}$) and the appropriate preference-based performance indicators: R-metric and Filatovas Spread (Δ_P).

The CI-NSGA-II, CI-SPEA2 and CI-SMS-EMOA are compared with one another and with respect to the preference-based algorithms: R-NSGA-II, W-HYPE. The scalable multi-objective test problems ZDT, DTLZ and WFG [50] have a known optimal front and can be used to benchmark the outcome of all the algorithms. Their features cover different classes of MOPs and difficulties: convex P_F, non-contiguous convex parts, non-convex, multimodal, etc. For those reasons, the test problems submit the new algorithms to distinct optimization difficulties and compare their results.

To summarise, the benchmark problems are:

- ZDT1
- ZDT2
- ZDT3
- ZDT4
- ZDT6
- DTLZ1
- DTLZ2
- DTLZ3
- DTLZ4
- DTLZ5
- DTLZ6
- DTLZ7

- WFG1
- WFG2
- WFG3
- WFG4
- WFG5
- WFG6
- WFG7
- WFG8
- WFG9

The preference-based algorithms are:

- CI-NSGA-II
- CI-SMS-EMOA
- CI-SPEA2
- CI-NSGA-II-P
- CI-SMS-EMOA-P
- CI-SPEA2-P
- R-NSGA-II
- WHYPE

The preference-based performance indicators are:

- R-metric
- κ (Referential Cluster)
- Ψ (Convex Hull)
- $C2_R$ (Reference Set)
- $D_{S \to P_F}$ (Convergence)
- Δ_P (Filatovas Spread)

6.1.1 Simulated DMs

The R-NSGA-II and W-HYPE algorithms use *a priori* reference points: $\boldsymbol{z}^0 = (0.3, 0.3)$, $\boldsymbol{z}^0 = (0.3, 0.3, 0.3)$ and $\boldsymbol{z}^0 = (0.3, 0.3, 0.3, 0.3, 0.3)$; for the problems with two, three or five objectives. The proposed collective intelligence MOEAs, on contrary, create the reference points interactively based on the participants' preferences. The main barrier to such benchmark problems is the involvement of human decision makers into interactive solution processes, which makes the generation of collective reference points troublesome.

Some artificial DMs methods can be used as techniques of generating preference information. Because interactive methods change significantly in the way they handle preference information, distinct artificial DMs were created for different preference information types [116,147,183]. Ojalehto [135] was the first to develop a framework for artificial DMs based on reference points.

This research provides simulated DMs to emulate the collectivity and implement the selection operator. The human preference is replaced with an artificial DM that allows to repetitively evaluate two or more points in a controlled environment. Each simulated DM has to vote between two individuals (c_1 or c_2) from the approximation set. They have a predefined reference point ($\boldsymbol{z}^j$) in the objective space which will be used to bias the votes. The *a priori* reference points represent objective values the artificial DMs would

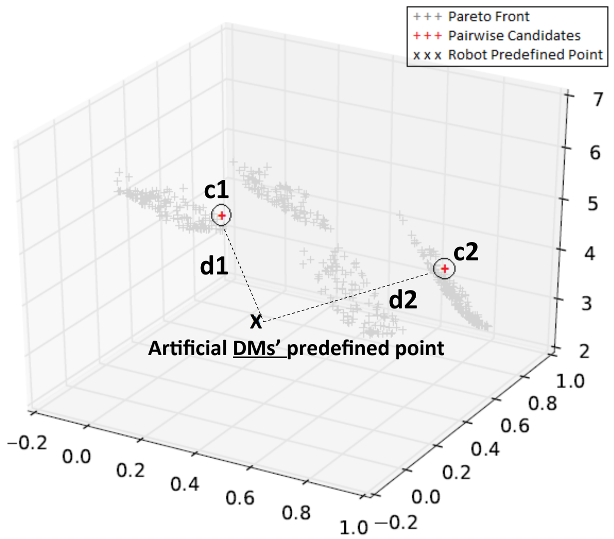


Figure 6.1: The artificial DM's predefined point will choose candidate $c1$ because the distance $d1 < d2$.

like to achieve. They choose an individual according to the closest distance between its predefined point $\boldsymbol{z}^j$ and each of the two candidates: $vote = \{c_1 : \|c_1 - \boldsymbol{z}^j\| \leq \|c_2 - \boldsymbol{z}^j\|\}$. Figure 6.1 illustrates candidates $c1$ and $c2$ with their respectively distances ($d1$ and $d2$) to the predefined point x. As $d1 < d2$, the artificial DM would vote on $c1$. A total of 100 artificial DMs provide a distribution of preferences that will be used to discover online reference points. It is important to notice that the collective reference point is built on the similarity of answers (votes) after the Gaussian Mixture approach.

In this experiment, the artificial DMs abstract the collectivity within a controlled environment. So the algorithms can be tested, compared and better understood in their working principles. The quantity of online reference points is directly related to the number of k clusters in the Gaussian Mixture model. In cases where k is not previously defined, the experiment used the X-means approach [136] to learn k from the data. This algorithm searches different values of k and scores each clustering model using the Bayesian Information Criterion (BIC): $BIC(M_j) = \iota_j(D) - (p_j/2)\log R$, where D is the dataset, M_j are models corresponding to solutions with different values of k, $\iota_j(D)$ is the log-likelihood of the dataset D according to model M_j, p_j is the number of parameters in M_j and R is the number of points in the dataset. X-means chooses the model with the best score.

In addition to the Gaussian Mixture model, the K-means algorithm was implemented to bring a different clustering technique into the analysis of the algorithms. But the performance of Gaussian Mixture for these benchmarking cases was consistently better.

6.1.2 Parameter and Experimental Settings

The benchmark problems that are used in this work are two-objective ZDT functions, three-objective DTLZ functions, three-objective WFG functions and five-objective WFG functions. The WHYPE algorithm was implemented on PISA framework [31]. PISA is a text-based interface for search algorithms. It is a library of ready-to-go modules that contains optimization problems (test and benchmark problems), selection modules (evolutionary multi-objective optimizers) and modules for performance assessment. All the others algorithms were implemented on a python evolutionary framework called DEAP [79].

The population size has been set to 200 for all the problems, except ZDT4 and ZDT6 with a population size of 100. The crossover and mutation probability are 0.9 and 0.1, respectively. Table 6.1 shows the number of iterations in each run and the number of variables for the benchmark problems.

As already mentioned in the previous subsection, the reference points for the simulated DMs voting and the *a priori* MOEAs are: $\boldsymbol{z}^0 = (0.3, 0.3)$, $\boldsymbol{z}^0 = (0.3, 0.3, 0.3)$ and $\boldsymbol{z}^0 = (0.3, 0.3, 0.3, 0.3, 0.3)$ for the problems with two, three or five objectives, respectively. These reference points are also used by the Reference Set Distance indicator ($C2_R$) to indicate the mean distance between the non-dominated points and the reference set R. Assuming the DM's expectation of the preferred region extension on the R-metric and $C2_R$ indicators is a concentrated area, Deb suggests the Δ is set to 0.1 for all test problems [112].

6.1.3 Box Plot Results

After 41 independent executions per algorithm on each test problem, the box plots were used to represent and support a valid judgment of the quality of the solutions and how different algorithms compare with each other. Figures 6.2 - 6.5 show the distribution of the performance indicators for the ZDT, DTLZ and WFG problems in the form of box plots. The box plots are grouped by performance indicators. Each group of box plots analyses and compares all the preference-based algorithms. For better visualization, all values were normalized.

Table 6.1: The number of iterations and variables for the benchmark problems.

Problem	Number Objectives	Number of Variables	Max Generations
ZDT1	2	30	200
ZDT2	2	30	250
ZDT3	2	30	330
ZDT4	2	30	600
ZDT6	2	30	900
DTLZ1	3	12	190
DTLZ2	3	12	200
DTLZ3	3	12	300
DTLZ4	3	12	200
DTLZ5	3	12	600
DTLZ6	3	12	600
DTLZ7	3	12	600
WFG1	3	34	400
WFG2	3	34	400
WFG3	3	34	400
WFG4	3	34	400
WFG5	3	34	400
WFG5	3	34	400
WFG6	3	34	400
WFG7	3	34	400
WFG8	3	34	400
WFG9	3	34	400
WFG1	5	34	400
WFG2	5	34	400
WFG3	5	34	400
WFG4	5	34	400
WFG5	5	34	400
WFG5	5	34	400
WFG6	5	34	400
WFG7	5	34	400
WFG8	5	34	400
WFG9	5	34	400

6.1.4 Statistical Hypothesis Test

Although box plots allow a visual comparison of the results, it is necessary to go beyond reporting the descriptive statistics of the performance indicators and apply a statistical hypothesis test. The Conover-Inman procedure [53] is a non-parametric method for testing equality of population medians. It can be implemented in a pairwise manner to determine if the results of one algorithm were significantly better than those of the other. A significance level, α, of 0.05 was used for all tests.

Tables 6.2-6.5 contain the results of the statistical analysis for ZDT, DTLZ and WFG test problems based on the mean values. Each table compares all the preference-based algorithms by the performance indicators used in the experiments.

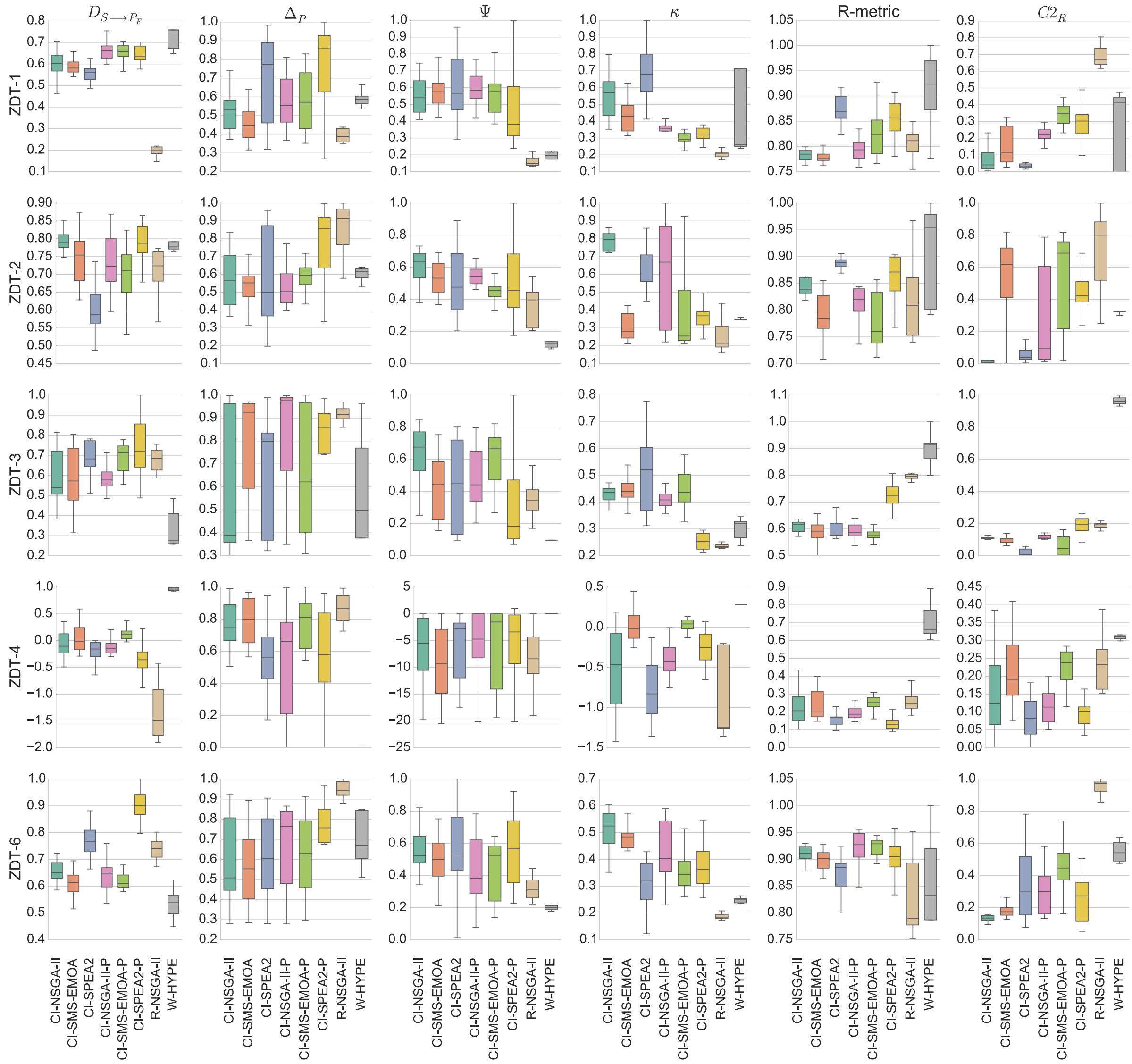


Figure 6.2: Distribution of the performance indicators for ZDT problems. Each group of box plots compares all the preference-based algorithms.

Figure 6.3: Distribution of the performance indicators for DTLZ problems. Each group of box plots compares all the preference-based algorithms.

Figure 6.4: Distribution of the performance indicators for WFG problems. Each group of box plots compares all the preference-based algorithms.

Figure 6.5: Distribution of the performance indicators for WFG problems with five objectives. Each group of box plots compares all the preference-based algorithms.

Table 6.2: ZDT results of the Conover-Inman statistical hypothesis [53] tests based on the mean values. The preference-based algorithms are compared by the performance indicators used in the experiments. Green cells (+) denote cases where the algorithm in the row statistically was better than the one in the column. Cells marked in red (−) are cases where the method in the column yielded statistically better results when compared to the method in the row. Cases where no significant difference was established are identified with a "∼".

$D_{S \to P_F}$

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
ZDT1	CI-NSGA-II	·	∼	+	−	−	−	+	−
	CI-SMS-EMOA	∼	·	+	−	−	−	+	−
	CI-SPEA2	−	−	·	−	−	−	+	−
	CI-NSGA-II-P	+	+	+	·	−	−	+	−
	CI-SMS-EMOA-P	+	+	+	+	·	∼	+	−
	CI-SPEA2-P	+	+	+	+	∼	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	+	+	+	+	+	+	+	·
ZDT2	CI-NSGA-II	·	+	+	+	+	∼	+	∼
	CI-SMS-EMOA	−	·	+	+	+	−	+	−
	CI-SPEA2	−	−	·	−	−	−	−	−
	CI-NSGA-II-P	−	−	+	·	+	−	+	−
	CI-SMS-EMOA-P	−	−	+	−	·	−	−	−
	CI-SPEA2-P	∼	+	+	+	+	·	+	∼
	R-NSGA-II	−	−	+	−	+	−	·	−
	WHYPE	∼	+	+	+	+	∼	+	·
ZDT3	CI-NSGA-II	·	+	−	∼	−	−	−	+
	CI-SMS-EMOA	−	·	−	−	−	−	−	+
	CI-SPEA2	+	+	·	+	∼	−	−	+
	CI-NSGA-II-P	∼	+	−	·	−	−	−	+
	CI-SMS-EMOA-P	+	+	∼	+	·	−	−	+
	CI-SPEA2-P	+	+	+	+	+	·	∼	+
	R-NSGA-II	+	+	+	+	+	∼	·	+
	WHYPE	−	−	−	−	−	−	−	·
ZDT4	CI-NSGA-II	·	+	+	−	+	−	−	+
	CI-SMS-EMOA	−	·	−	−	+	−	−	+
	CI-SPEA2	−	+	·	−	+	−	−	+
	CI-NSGA-II-P	+	+	+	·	+	−	−	+
	CI-SMS-EMOA-P	−	−	−	−	·	−	−	+
	CI-SPEA2-P	+	+	+	+	+	·	−	+
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
ZDT6	CI-NSGA-II	·	+	−	+	+	−	−	+
	CI-SMS-EMOA	−	·	−	−	∼	−	−	+
	CI-SPEA2	+	+	·	+	+	−	−	+
	CI-NSGA-II-P	−	+	−	·	+	−	−	+
	CI-SMS-EMOA-P	−	∼	−	−	·	−	−	+
	CI-SPEA2-P	+	+	+	+	+	·	+	+
	R-NSGA-II	+	+	+	+	+	−	·	+
	WHYPE	−	−	−	−	−	−	−	·

Δ_P

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
ZDT1	CI-NSGA-II	·	−	+	+	+	+	−	+
	CI-SMS-EMOA	+	·	+	+	+	+	−	+
	CI-SPEA2	−	−	·	−	−	+	−	−
	CI-NSGA-II-P	−	−	+	·	+	+	−	+
	CI-SMS-EMOA-P	−	−	+	−	·	+	−	+
	CI-SPEA2-P	−	−	−	−	−	·	−	−
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	−	+	−	−	+	−	·
ZDT2	CI-NSGA-II	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	+	−	−	+	−
	CI-SPEA2	−	+	·	+	+	+	+	−
	CI-NSGA-II-P	−	−	−	·	−	−	+	−
	CI-SMS-EMOA-P	−	+	−	+	·	+	+	−
	CI-SPEA2-P	−	−	−	−	−	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	+	+	+	+	+	+	·
ZDT3	CI-NSGA-II	·	∼	−	−	+	+	+	+
	CI-SMS-EMOA	∼	·	−	−	+	+	+	+
	CI-SPEA2	+	+	·	∼	+	+	+	+
	CI-NSGA-II-P	+	+	∼	·	+	+	+	+
	CI-SMS-EMOA-P	−	−	−	−	·	−	−	−
	CI-SPEA2-P	−	−	−	−	+	·	−	−
	R-NSGA-II	−	−	−	−	+	+	·	−
	WHYPE	−	−	−	−	+	+	+	·
ZDT4	CI-NSGA-II	·	+	−	−	−	−	+	
	CI-SMS-EMOA	−	·	−	−	−	−	−	
	CI-SPEA2	+	+	·	−	+	+	+	
	CI-NSGA-II-P	+	+	+	·	+	+	+	
	CI-SMS-EMOA-P	+	+	−	−	·	−	+	
	CI-SPEA2-P	+	+	−	−	+	·	+	
	R-NSGA-II	−	−	−	−	−	−	·	
	WHYPE								·
ZDT6	CI-NSGA-II	·	−	+	+	+	+	+	+
	CI-SMS-EMOA	+	·	+	+	+	+	+	+
	CI-SPEA2	−	−	·	+	−	+	+	+
	CI-NSGA-II-P	−	−	−	·	−	+	+	+
	CI-SMS-EMOA-P	−	−	+	+	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	+	+	·

Ψ

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
ZDT1	CI-NSGA-II	·	−	+	−	∼	+	+	+
	CI-SMS-EMOA	+	·	+	−	+	+	+	+
	CI-SPEA2	−	−	·	−	−	+	+	+
	CI-NSGA-II-P	+	+	+	·	+	+	+	+
	CI-SMS-EMOA-P	∼	−	+	−	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·
ZDT2	CI-NSGA-II	·	∼	+	∼	+	+	+	+
	CI-SMS-EMOA	∼	·	+	∼	+	+	+	+
	CI-SPEA2	−	−	·	−	−	+	+	+
	CI-NSGA-II-P	∼	∼	+	·	+	+	+	+
	CI-SMS-EMOA-P	−	−	+	−	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
ZDT3	CI-NSGA-II	·	+	+	−	−	+	+	+
	CI-SMS-EMOA	−	·	+	−	−	+	−	+
	CI-SPEA2	−	−	·	−	−	+	−	+
	CI-NSGA-II-P	+	+	+	·	∼	+	+	+
	CI-SMS-EMOA-P	+	+	+	∼	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	−	+
	R-NSGA-II	−	+	+	−	−	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
ZDT4	CI-NSGA-II	·	+	+	−	+	+	−	
	CI-SMS-EMOA	−	·	−	−	−	+	−	
	CI-SPEA2	−	+	·	−	+	+	−	
	CI-NSGA-II-P	+	+	+	·	+	+	∼	
	CI-SMS-EMOA-P	−	+	−	−	·	+	−	
	CI-SPEA2-P	−	−	−	−	−	·	−	
	R-NSGA-II	+	+	+	∼	+	+	·	
	WHYPE								·
ZDT6	CI-NSGA-II	·	−	+	+	+	−	+	+
	CI-SMS-EMOA	+	·	+	+	+	+	+	+
	CI-SPEA2	−	−	·	−	−	−	−	+
	CI-NSGA-II-P	−	−	+	·	−	−	−	+
	CI-SMS-EMOA-P	−	−	+	+	·	−	−	+
	CI-SPEA2-P	+	−	+	+	+	·	+	+
	R-NSGA-II	−	−	+	+	+	−	·	+
	WHYPE	−	−	−	−	−	−	−	·

κ

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
ZDT1	CI-NSGA-II	·	+	−	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	+	+	+	+	+
	CI-SPEA2	+	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	−	−	·	+	+	+	+
	CI-SMS-EMOA-P	−	−	−	−	·	+	+	−
	CI-SPEA2-P	−	−	−	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	+	−	+	·
ZDT2	CI-NSGA-II	·	+	−	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	−	+	−	+	−
	CI-SPEA2	+	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	+	−	·	+	+	+	+
	CI-SMS-EMOA-P	−	−	−	−	·	−	+	−
	CI-SPEA2-P	−	+	−	−	+	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	+	−	−	+	+	+	·
ZDT3	CI-NSGA-II	·	−	∼	+	−	+	+	+
	CI-SMS-EMOA	+	·	+	+	+	+	+	+
	CI-SPEA2	∼	−	·	+	−	+	+	+
	CI-NSGA-II-P	−	−	−	·	−	+	+	+
	CI-SMS-EMOA-P	+	−	+	+	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·
ZDT4	CI-NSGA-II	·	+	+	−	−	+	−	+
	CI-SMS-EMOA	−	·	+	−	−	−	−	+
	CI-SPEA2	−	−	·	−	−	−	−	−
	CI-NSGA-II-P	+	+	+	·	+	+	−	+
	CI-SMS-EMOA-P	−	+	+	−	·	+	−	+
	CI-SPEA2-P	−	+	+	−	−	·	−	+
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	−	+	−	−	−	−	·
ZDT6	CI-NSGA-II	·	−	+	+	+	+	+	+
	CI-SMS-EMOA	+	·	+	+	+	+	+	+
	CI-SPEA2	−	−	·	−	−	−	+	+
	CI-NSGA-II-P	−	−	+	·	+	+	+	+
	CI-SMS-EMOA-P	−	−	+	−	·	−	+	+
	CI-SPEA2-P	−	−	+	−	+	·	−	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·

R-metric

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
ZDT1	CI-NSGA-II	·	∼	+	+	+	+	+	+
	CI-SMS-EMOA	∼	·	+	+	+	+	+	+
	CI-SPEA2	−	−	·	−	−	−	−	+
	CI-NSGA-II-P	−	−	+	·	+	+	+	+
	CI-SMS-EMOA-P	−	−	+	−	·	+	−	+
	CI-SPEA2-P	−	−	+	−	−	·	−	+
	R-NSGA-II	−	−	+	−	+	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
ZDT2	CI-NSGA-II	·	−	+	−	−	+	−	+
	CI-SMS-EMOA	+	·	+	+	−	+	+	+
	CI-SPEA2	−	−	·	−	−	−	−	−
	CI-NSGA-II-P	+	−	+	·	−	+	∼	+
	CI-SMS-EMOA-P	+	+	+	+	·	+	+	+
	CI-SPEA2-P	−	−	+	−	−	·	−	+
	R-NSGA-II	+	−	+	∼	−	+	·	+
	WHYPE	−	−	+	−	−	−	−	·
ZDT3	CI-NSGA-II	·	−	−	−	−	+	+	+
	CI-SMS-EMOA	+	·	+	∼	−	+	+	+
	CI-SPEA2	+	−	·	−	−	+	+	+
	CI-NSGA-II-P	+	∼	+	·	−	+	+	+
	CI-SMS-EMOA-P	+	+	+	+	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
ZDT4	CI-NSGA-II	·	+	−	−	+	−	+	+
	CI-SMS-EMOA	−	·	−	−	+	−	+	+
	CI-SPEA2	+	+	·	+	+	−	+	+
	CI-NSGA-II-P	+	+	−	·	+	−	+	+
	CI-SMS-EMOA-P	−	−	−	−	·	−	−	+
	CI-SPEA2-P	+	+	+	+	+	·	+	+
	R-NSGA-II	−	−	−	−	+	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
ZDT6	CI-NSGA-II	·	−	−	+	+	−	−	−
	CI-SMS-EMOA	+	·	−	+	+	∼	−	−
	CI-SPEA2	+	+	·	+	+	+	−	−
	CI-NSGA-II-P	−	−	−	·	∼	−	−	−
	CI-SMS-EMOA-P	−	−	−	∼	·	−	−	−
	CI-SPEA2-P	+	∼	−	+	+	·	−	−
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	+	+	+	+	+	+	−	·

$C2_R$

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
ZDT1	CI-NSGA-II	·	+	−	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	+	+	+	+	+
	CI-SPEA2	+	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	−	−	·	+	+	+	+
	CI-SMS-EMOA-P	−	−	−	−	·	−	+	−
	CI-SPEA2-P	−	−	−	−	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	+	−	+	·
ZDT2	CI-NSGA-II	·	+	−	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	−	−	−	+	−
	CI-SPEA2	+	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	+	−	·	+	+	+	+
	CI-SMS-EMOA-P	−	+	−	−	·	−	+	−
	CI-SPEA2-P	−	+	−	−	+	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	+	−	−	+	+	+	·
ZDT3	CI-NSGA-II	·	−	−	+	−	+	+	+
	CI-SMS-EMOA	+	·	−	+	−	+	+	+
	CI-SPEA2	+	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	−	−	·	−	+	+	+
	CI-SMS-EMOA-P	+	+	−	+	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
ZDT4	CI-NSGA-II	·	+	+	−	+	−	−	−
	CI-SMS-EMOA	−	·	−	−	+	−	−	+
	CI-SPEA2	−	+	·	−	+	−	+	+
	CI-NSGA-II-P	+	+	+	·	+	−	+	+
	CI-SMS-EMOA-P	−	−	−	−	·	−	−	+
	CI-SPEA2-P	+	+	+	+	+	·	+	+
	R-NSGA-II	−	+	−	−	+	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
ZDT6	CI-NSGA-II	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	−	·	+	+	+	+	+	+
	CI-SPEA2	−	−	·	−	+	−	+	+
	CI-NSGA-II-P	−	−	+	·	+	−	+	+
	CI-SMS-EMOA-P	−	−	−	−	·	−	+	+
	CI-SPEA2-P	−	−	+	+	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·

Table 6.3: DTLZ results of the Conover-Inman statistical hypothesis [53] tests based on the mean values. The preference-based algorithms are compared by the performance indicators used in the experiments. Green cells (+) denote cases where the algorithm in the row statistically was better than the one in the column. Cells marked in red (−) are cases where the method in the column yielded statistically better results when compared to the method in the row. Cases where no significant difference was established are identified with a "∼".

$D_{S \to P_F}$

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
DTLZ1	CI-NSGA-II	·	−	+	+	−	+	+	+
	CI-SMS-EMOA	+	·	+	+	−	+	+	+
	CI-SPEA2	−	−	·	−	−	+	+	+
	CI-NSGA-II-P	−	−	+	·	−	+	+	+
	CI-SMS-EMOA-P	+	+	+	+	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·
DTLZ2	CI-NSGA-II	·	+	+	−	+	+	+	+
	CI-SMS-EMOA	−	·	+	−	−	−	−	−
	CI-SPEA2	−	−	·	−	−	−	−	−
	CI-NSGA-II-P	+	+	+	·	+	+	+	+
	CI-SMS-EMOA-P	−	+	+	−	·	+	+	−
	CI-SPEA2-P	−	+	+	−	−	·	+	−
	R-NSGA-II	−	+	+	−	−	+	·	−
	WHYPE	−	+	+	−	+	+	+	·
DTLZ3	CI-NSGA-II	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	−	·	+	−	+	+	+	+
	CI-SPEA2	−	−	·	−	−	−	−	+
	CI-NSGA-II-P	−	+	+	·	+	+	+	+
	CI-SMS-EMOA-P	−	−	+	−	·	+	+	+
	CI-SPEA2-P	−	−	+	−	−	·	−	+
	R-NSGA-II	−	−	+	−	−	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
DTLZ4	CI-NSGA-II	·	−	−	+	−	−	∼	+
	CI-SMS-EMOA	+	·	−	+	+	−	+	+
	CI-SPEA2	+	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	−	−	·	−	−	−	+
	CI-SMS-EMOA-P	+	−	−	+	·	−	+	+
	CI-SPEA2-P	+	+	−	+	+	·	+	+
	R-NSGA-II	∼	−	−	+	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
DTLZ5	CI-NSGA-II	·	−	+	∼	+	−	−	−
	CI-SMS-EMOA	+	·	+	+	+	+	−	−
	CI-SPEA2	−	−	·	−	−	−	−	−
	CI-NSGA-II-P	∼	−	+	·	+	−	−	−
	CI-SMS-EMOA-P	−	−	+	−	·	−	−	−
	CI-SPEA2-P	+	−	+	+	+	·	−	−
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	+	+	+	+	+	+	−	·
DTLZ6	CI-NSGA-II	·	+	+	+	+	+	−	−
	CI-SMS-EMOA	−	·	−	−	+	−	−	−
	CI-SPEA2	−	+	·	−	+	−	−	−
	CI-NSGA-II-P	−	+	+	·	+	+	−	−
	CI-SMS-EMOA-P	−	−	−	−	·	−	−	−
	CI-SPEA2-P	−	+	+	−	+	·	−	−
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	+	+	+	+	+	+	−	·
DTLZ7	CI-NSGA-II	·	+	+	−	−	∼	+	−
	CI-SMS-EMOA	−	·	+	−	−	−	+	−
	CI-SPEA2	−	−	·	−	−	−	+	−
	CI-NSGA-II-P	+	+	+	·	−	+	+	−
	CI-SMS-EMOA-P	+	+	+	+	·	+	+	+
	CI-SPEA2-P	∼	+	+	−	−	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	+	+	+	+	−	+	+	·

Δ_P

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
DTLZ1	CI-NSGA-II	·	−	−	−	−	+	+	+
	CI-SMS-EMOA	+	·	−	−	+	+	+	+
	CI-SPEA2	+	+	·	∼	+	+	+	+
	CI-NSGA-II-P	+	+	∼	·	+	+	+	+
	CI-SMS-EMOA-P	+	−	−	−	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·
DTLZ2	CI-NSGA-II	·	−	−	−	−	−	−	−
	CI-SMS-EMOA	+	·	−	+	−	−	−	−
	CI-SPEA2	+	+	·	+	+	−	+	−
	CI-NSGA-II-P	+	−	−	·	−	−	−	−
	CI-SMS-EMOA-P	+	+	−	+	·	−	+	−
	CI-SPEA2-P	+	+	+	+	+	·	+	−
	R-NSGA-II	+	+	−	+	−	−	·	−
	WHYPE	+	+	+	+	+	+	+	·
DTLZ3	CI-NSGA-II	·	−	−	+	−	−	−	−
	CI-SMS-EMOA	+	·	−	+	−	−	−	−
	CI-SPEA2	+	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	−	−	·	−	−	−	−
	CI-SMS-EMOA-P	+	+	−	+	·	−	−	−
	CI-SPEA2-P	+	+	−	+	+	·	+	−
	R-NSGA-II	+	+	−	+	+	−	·	−
	WHYPE	+	+	−	+	+	+	+	·
DTLZ4	CI-NSGA-II	·	+	−	−	+	−	−	−
	CI-SMS-EMOA	−	·	−	−	+	−	−	−
	CI-SPEA2	+	+	·	+	+	−	+	+
	CI-NSGA-II-P	+	+	−	·	+	−	+	−
	CI-SMS-EMOA-P	−	−	−	−	·	−	−	−
	CI-SPEA2-P	+	+	+	+	+	·	+	+
	R-NSGA-II	+	+	−	−	+	−	·	−
	WHYPE	+	+	−	+	+	−	+	·
DTLZ5	CI-NSGA-II	·	−	−	+	−	−	+	−
	CI-SMS-EMOA	+	·	−	+	+	−	+	+
	CI-SPEA2	+	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	−	−	·	−	−	−	−
	CI-SMS-EMOA-P	+	−	−	+	·	−	+	−
	CI-SPEA2-P	+	+	−	+	+	·	+	+
	R-NSGA-II	−	−	−	+	−	−	·	−
	WHYPE	+	−	−	+	+	−	+	·
DTLZ6	CI-NSGA-II	·	+	∼	+	∼	+	+	+
	CI-SMS-EMOA	−	·	−	+	−	+	+	+
	CI-SPEA2	∼	+	·	+	∼	+	+	+
	CI-NSGA-II-P	−	−	−	·	−	+	+	+
	CI-SMS-EMOA-P	∼	+	∼	+	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	+	+	·
DTLZ7	CI-NSGA-II	·	+	−	−	+	−	−	−
	CI-SMS-EMOA	−	·	−	−	+	−	−	−
	CI-SPEA2	+	+	·	+	+	−	−	−
	CI-NSGA-II-P	+	+	−	·	+	−	−	−
	CI-SMS-EMOA-P	−	−	−	−	·	−	−	−
	CI-SPEA2-P	+	+	+	+	+	·	−	−
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	+	+	+	+	+	+	−	·

Ψ

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
DTLZ1	CI-NSGA-II	·	∼	−	−	+	+	+	∼
	CI-SMS-EMOA	∼	·	−	−	+	+	+	∼
	CI-SPEA2	+	+	·	−	+	+	+	+
	CI-NSGA-II-P	+	+	+	·	+	+	+	+
	CI-SMS-EMOA-P	−	−	−	−	·	−	−	−
	CI-SPEA2-P	−	−	−	−	+	·	−	−
	R-NSGA-II	−	−	−	−	+	+	·	−
	WHYPE	∼	∼	−	−	+	+	+	·
DTLZ2	CI-NSGA-II	·	∼	+	∼	∼	+	∼	+
	CI-SMS-EMOA	∼	·	+	∼	∼	+	∼	+
	CI-SPEA2	−	−	·	−	−	−	−	+
	CI-NSGA-II-P	∼	∼	+	·	∼	+	∼	+
	CI-SMS-EMOA-P	∼	∼	+	∼	·	+	∼	+
	CI-SPEA2-P	−	−	+	−	−	·	−	+
	R-NSGA-II	∼	∼	+	∼	∼	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
DTLZ3	CI-NSGA-II	·	+	+	+	−	+	+	+
	CI-SMS-EMOA	−	·	+	−	−	+	−	+
	CI-SPEA2	−	−	·	−	−	+	−	+
	CI-NSGA-II-P	−	+	+	·	−	+	−	+
	CI-SMS-EMOA-P	+	+	+	+	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	−	+
	R-NSGA-II	−	+	+	+	−	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
DTLZ4	CI-NSGA-II	·	+	∼	∼	∼	∼	∼	+
	CI-SMS-EMOA	−	·	−	−	−	−	−	+
	CI-SPEA2	∼	+	·	∼	∼	∼	∼	+
	CI-NSGA-II-P	∼	+	∼	·	∼	∼	∼	+
	CI-SMS-EMOA-P	∼	+	∼	∼	·	∼	∼	+
	CI-SPEA2-P	∼	+	∼	∼	∼	·	∼	+
	R-NSGA-II	∼	+	∼	∼	∼	∼	·	+
	WHYPE	−	−	−	−	−	−	−	·
DTLZ5	CI-NSGA-II	·	∼	∼	∼	∼	∼	∼	+
	CI-SMS-EMOA	∼	·	∼	∼	∼	∼	∼	+
	CI-SPEA2	∼	∼	·	∼	∼	∼	∼	+
	CI-NSGA-II-P	∼	∼	∼	·	∼	∼	∼	+
	CI-SMS-EMOA-P	∼	∼	∼	∼	·	∼	∼	+
	CI-SPEA2-P	∼	∼	∼	∼	∼	·	∼	+
	R-NSGA-II	∼	∼	∼	∼	∼	∼	·	+
	WHYPE	−	−	−	−	−	−	−	·
DTLZ6	CI-NSGA-II	·	∼	∼	∼	∼	∼	∼	+
	CI-SMS-EMOA	∼	·	∼	∼	∼	∼	∼	+
	CI-SPEA2	∼	∼	·	∼	∼	∼	∼	+
	CI-NSGA-II-P	∼	∼	∼	·	∼	∼	∼	+
	CI-SMS-EMOA-P	∼	∼	∼	∼	·	∼	∼	+
	CI-SPEA2-P	∼	∼	∼	∼	∼	·	∼	+
	R-NSGA-II	∼	∼	∼	∼	∼	∼	·	+
	WHYPE	−	−	−	−	−	−	−	·
DTLZ7	CI-NSGA-II	·	∼	−	−	+	+	+	+
	CI-SMS-EMOA	∼	·	−	−	+	+	+	+
	CI-SPEA2	+	+	·	∼	+	+	+	+
	CI-NSGA-II-P	+	+	∼	·	+	+	+	+
	CI-SMS-EMOA-P	−	−	−	−	·	−	−	−
	CI-SPEA2-P	−	−	−	−	+	·	−	−
	R-NSGA-II	−	−	−	−	+	+	·	−
	WHYPE	−	−	−	−	+	+	+	·

κ

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
DTLZ1	CI-NSGA-II	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	−	·	+	+	−	+	+	+
	CI-SPEA2	−	−	·	−	−	+	+	+
	CI-NSGA-II-P	−	−	+	·	−	+	+	+
	CI-SMS-EMOA-P	−	+	+	+	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	−	+
	R-NSGA-II	−	−	−	−	−	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
DTLZ2	CI-NSGA-II	·	+	+	+	+	+	−	+
	CI-SMS-EMOA	−	·	+	−	+	+	−	+
	CI-SPEA2	−	−	·	−	−	+	−	+
	CI-NSGA-II-P	−	+	+	·	+	+	−	+
	CI-SMS-EMOA-P	−	−	+	−	·	+	−	+
	CI-SPEA2-P	−	−	−	−	−	·	−	−
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	−	−	−	−	+	−	·
DTLZ3	CI-NSGA-II	·	+	+	+	+	+	−	+
	CI-SMS-EMOA	−	·	+	−	−	+	−	+
	CI-SPEA2	−	−	·	−	−	−	−	+
	CI-NSGA-II-P	−	+	+	+	+	+	−	+
	CI-SMS-EMOA-P	−	+	+	−	·	+	−	+
	CI-SPEA2-P	−	−	+	−	−	·	−	+
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
DTLZ4	CI-NSGA-II	·	+	+	−	+	+	+	+
	CI-SMS-EMOA	−	·	+	−	−	+	−	−
	CI-SPEA2	−	−	·	−	−	−	−	−
	CI-NSGA-II-P	+	+	+	·	+	+	+	+
	CI-SMS-EMOA-P	−	+	+	−	·	+	−	+
	CI-SPEA2-P	−	−	+	−	−	·	−	−
	R-NSGA-II	−	+	+	−	+	+	·	+
	WHYPE	−	+	+	−	−	+	−	·
DTLZ5	CI-NSGA-II	·	+	+	−	+	+	−	−
	CI-SMS-EMOA	−	·	+	−	−	−	−	−
	CI-SPEA2	−	−	·	−	−	−	−	−
	CI-NSGA-II-P	+	+	+	·	+	+	−	−
	CI-SMS-EMOA-P	−	+	+	−	·	+	−	−
	CI-SPEA2-P	−	+	+	−	−	·	−	−
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	+	+	+	+	+	+	−	·
DTLZ6	CI-NSGA-II	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	−	+	−	−	−
	CI-SPEA2	−	+	·	+	+	+	−	+
	CI-NSGA-II-P	−	+	−	·	+	+	−	−
	CI-SMS-EMOA-P	−	−	−	−	·	−	−	−
	CI-SPEA2-P	−	+	−	−	+	·	−	−
	R-NSGA-II	−	+	+	+	+	+	·	+
	WHYPE	−	+	−	+	+	+	−	·
DTLZ7	CI-NSGA-II	·	+	+	−	−	−	+	+
	CI-SMS-EMOA	−	·	+	−	−	−	+	+
	CI-SPEA2	−	−	·	−	−	−	+	+
	CI-NSGA-II-P	+	+	+	·	+	+	+	+
	CI-SMS-EMOA-P	+	+	+	+	·	−	+	+
	CI-SPEA2-P	+	+	+	−	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·

R-metric

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
DTLZ1	CI-NSGA-II	·	−	+	+	−	+	+	−
	CI-SMS-EMOA	+	·	+	+	−	+	+	−
	CI-SPEA2	−	−	·	−	−	−	+	−
	CI-NSGA-II-P	−	−	+	·	−	+	+	−
	CI-SMS-EMOA-P	+	+	+	+	·	+	+	−
	CI-SPEA2-P	−	−	+	−	−	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	+	+	+	+	+	+	+	·
DTLZ2	CI-NSGA-II	·	−	−	+	−	−	−	+
	CI-SMS-EMOA	+	·	+	+	+	+	+	+
	CI-SPEA2	+	−	·	+	+	−	−	+
	CI-NSGA-II-P	−	−	−	·	−	−	−	+
	CI-SMS-EMOA-P	+	−	−	+	·	−	−	+
	CI-SPEA2-P	+	−	+	+	+	·	+	+
	R-NSGA-II	+	−	−	+	+	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
DTLZ3	CI-NSGA-II	·	+	+	−	+	+	+	+
	CI-SMS-EMOA	−	·	−	−	−	+	+	+
	CI-SPEA2	−	+	·	−	∼	+	+	+
	CI-NSGA-II-P	+	+	+	·	+	+	+	+
	CI-SMS-EMOA-P	−	+	∼	−	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
DTLZ4	CI-NSGA-II	·	+	+	+	+	+	−	+
	CI-SMS-EMOA	−	·	−	+	−	−	−	−
	CI-SPEA2	−	+	·	+	+	+	−	+
	CI-NSGA-II-P	−	−	−	·	−	−	−	−
	CI-SMS-EMOA-P	−	+	−	+	·	−	−	−
	CI-SPEA2-P	−	+	−	+	+	·	−	+
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	+	−	+	+	−	−	·
DTLZ5	CI-NSGA-II	·	−	+	−	∼	∼	+	+
	CI-SMS-EMOA	+	·	+	+	+	+	+	+
	CI-SPEA2	−	−	·	−	−	−	−	−
	CI-NSGA-II-P	+	−	+	·	+	+	+	+
	CI-SMS-EMOA-P	∼	−	+	−	·	∼	+	+
	CI-SPEA2-P	∼	−	+	−	∼	·	+	+
	R-NSGA-II	−	−	+	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
DTLZ6	CI-NSGA-II	·	+	+	+	+	+	−	+
	CI-SMS-EMOA	−	·	−	−	+	−	−	−
	CI-SPEA2	−	+	·	−	+	−	−	−
	CI-NSGA-II-P	−	+	+	·	+	+	−	−
	CI-SMS-EMOA-P	−	−	−	−	·	−	−	−
	CI-SPEA2-P	−	+	+	−	+	·	−	−
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	+	+	+	+	+	−	·
DTLZ7	CI-NSGA-II	·	−	−	+	−	+	−	−
	CI-SMS-EMOA	+	·	−	+	+	+	−	−
	CI-SPEA2	+	+	·	+	+	+	−	−
	CI-NSGA-II-P	−	−	−	·	−	−	−	−
	CI-SMS-EMOA-P	+	−	−	+	·	+	−	−
	CI-SPEA2-P	−	−	−	+	−	·	−	−
	R-NSGA-II	+	+	+	+	+	+	·	−
	WHYPE	+	+	+	+	+	+	+	·

$C2_R$

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
DTLZ1	CI-NSGA-II	·	−	+	+	−	+	+	∼
	CI-SMS-EMOA	−	·	+	+	−	+	+	+
	CI-SPEA2	−	−	·	−	−	+	+	+
	CI-NSGA-II-P	−	−	+	·	−	+	+	+
	CI-SMS-EMOA-P	+	+	+	+	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
DTLZ2	CI-NSGA-II	·	+	+	+	+	+	−	+
	CI-SMS-EMOA	−	·	+	−	+	+	−	+
	CI-SPEA2	−	−	·	−	−	+	−	−
	CI-NSGA-II-P	−	+	+	·	+	+	−	+
	CI-SMS-EMOA-P	−	−	+	−	·	+	−	+
	CI-SPEA2-P	−	−	−	−	−	·	−	−
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	−	+	−	−	+	−	·
DTLZ3	CI-NSGA-II	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	−	·	+	−	+	+	+	+
	CI-SPEA2	−	−	·	−	−	−	−	+
	CI-NSGA-II-P	−	+	+	·	+	+	+	+
	CI-SMS-EMOA-P	−	−	+	−	·	+	+	+
	CI-SPEA2-P	−	−	+	−	−	·	−	+
	R-NSGA-II	−	−	+	−	−	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
DTLZ4	CI-NSGA-II	·	+	+	−	+	+	+	+
	CI-SMS-EMOA	−	·	+	−	−	+	−	−
	CI-SPEA2	−	−	·	−	−	−	−	−
	CI-NSGA-II-P	+	+	+	·	+	+	+	+
	CI-SMS-EMOA-P	−	+	+	−	·	+	−	−
	CI-SPEA2-P	−	−	+	−	−	·	−	−
	R-NSGA-II	−	+	+	−	+	+	·	−
	WHYPE	−	+	+	−	+	+	+	·
DTLZ5	CI-NSGA-II	·	+	+	−	+	+	+	+
	CI-SMS-EMOA	−	·	+	−	−	−	+	+
	CI-SPEA2	−	−	·	−	−	−	−	+
	CI-NSGA-II-P	+	+	+	·	+	+	+	+
	CI-SMS-EMOA-P	−	+	+	−	·	+	+	+
	CI-SPEA2-P	−	+	+	−	−	·	+	+
	R-NSGA-II	−	−	+	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
DTLZ6	CI-NSGA-II	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	−	−	−	−	−
	CI-SPEA2	−	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	+	−	·	+	+	−	+
	CI-SMS-EMOA-P	−	+	−	−	·	−	−	−
	CI-SPEA2-P	−	+	−	−	+	·	−	−
	R-NSGA-II	−	+	−	+	+	+	·	+
	WHYPE	−	+	−	−	+	+	−	·
DTLZ7	CI-NSGA-II	·	+	+	−	−	−	+	+
	CI-SMS-EMOA	−	·	+	−	−	−	+	+
	CI-SPEA2	−	−	·	−	−	−	+	+
	CI-NSGA-II-P	+	+	+	·	+	+	+	+
	CI-SMS-EMOA-P	+	+	+	−	·	−	+	+
	CI-SPEA2-P	+	+	+	−	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·

Table 6.4: WFG results of the Conover-Inman statistical hypothesis [53] tests based on the mean values. The preference-based algorithms are compared by the performance indicators used in the experiments. Green cells (+) denote cases where the algorithm in the row statistically was better than the one in the column. Cells marked in red (−) are cases where the method in the column yielded statistically better results when compared to the method in the row. Cases where no significant difference was established are identified with a "∼".

$D_{S\to P_r}$

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
WFG1	CI-NSGA-II	·	−	+	+	+	+	+	+
	CI-SMS-EMOA	+	·	+	+	+	+	+	+
	CI-SPEA2	−	−	·	−	−	−	+	+
	CI-NSGA-II-P	−	−	+	·	−	+	+	+
	CI-SMS-EMOA-P	−	−	+	+	·	+	+	+
	CI-SPEA2-P	−	−	+	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·
WFG2	CI-NSGA-II	·	+	+	+	+	−	+	+
	CI-SMS-EMOA	−	·	−	+	−	−	−	+
	CI-SPEA2	−	+	·	+	+	−	+	+
	CI-NSGA-II-P	−	−	−	·	−	−	−	+
	CI-SMS-EMOA-P	−	+	−	+	·	−	−	+
	CI-SPEA2-P	+	+	+	+	+	·	+	+
	R-NSGA-II	−	+	−	+	+	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG3	CI-NSGA-II	·	+	−	−	+	−	−	+
	CI-SMS-EMOA	−	·	−	−	+	−	−	+
	CI-SPEA2	+	+	·	+	+	+	−	+
	CI-NSGA-II-P	+	+	−	·	+	−	−	+
	CI-SMS-EMOA-P	−	−	−	−	·	−	−	+
	CI-SPEA2-P	+	+	−	+	+	·	−	+
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG4	CI-NSGA-II	·	−	+	+	+	−	+	+
	CI-SMS-EMOA	+	·	+	+	+	+	+	+
	CI-SPEA2	−	−	·	−	+	−	+	+
	CI-NSGA-II-P	−	−	+	·	+	−	+	+
	CI-SMS-EMOA-P	−	−	−	−	·	−	+	+
	CI-SPEA2-P	+	−	+	+	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG5	CI-NSGA-II	·	+	−	+	∼	+	+	+
	CI-SMS-EMOA	−	·	−	+	−	+	+	+
	CI-SPEA2	+	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	−	−	·	−	+	+	+
	CI-SMS-EMOA-P	∼	+	−	+	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG6	CI-NSGA-II	·	+	+	−	−	+	+	+
	CI-SMS-EMOA	−	·	+	−	−	+	+	+
	CI-SPEA2	−	−	·	−	−	−	+	+
	CI-NSGA-II-P	+	+	+	·	−	+	+	+
	CI-SMS-EMOA-P	+	+	+	+	·	+	+	+
	CI-SPEA2-P	−	−	+	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG7	CI-NSGA-II	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	−	−	+	+	−
	CI-SPEA2	−	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	+	−	·	−	+	+	+
	CI-SMS-EMOA-P	−	+	−	+	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	+	−	−	−	+	+	·
WFG8	CI-NSGA-II	·	−	−	+	+	+	−	+
	CI-SMS-EMOA	+	·	+	+	+	+	−	+
	CI-SPEA2	+	−	·	+	+	+	−	+
	CI-NSGA-II-P	−	−	−	·	−	+	−	+
	CI-SMS-EMOA-P	−	−	−	+	·	+	−	+
	CI-SPEA2-P	−	−	−	−	−	·	−	+
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG9	CI-NSGA-II	·	+	+	−	+	−	−	+
	CI-SMS-EMOA	−	·	−	−	−	−	−	−
	CI-SPEA2	−	+	·	−	+	−	−	−
	CI-NSGA-II-P	+	+	+	·	+	−	+	+
	CI-SMS-EMOA-P	−	+	−	−	·	−	−	−
	CI-SPEA2-P	+	+	+	+	+	·	+	+
	R-NSGA-II	+	+	+	−	+	−	·	+
	WHYPE	−	+	+	−	+	−	−	·

Δ_P

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
WFG1	CI-NSGA-II	·	+	−	+	+	+	+	−
	CI-SMS-EMOA	−	·	−	+	−	−	−	−
	CI-SPEA2	+	+	·	+	+	+	+	−
	CI-NSGA-II-P	−	−	−	·	−	−	−	−
	CI-SMS-EMOA-P	−	+	−	+	·	−	−	−
	CI-SPEA2-P	−	+	−	+	+	·	+	−
	R-NSGA-II	−	+	−	+	+	−	·	−
	WHYPE	+	+	+	+	+	+	+	·
WFG2	CI-NSGA-II	·	−	−	−	+	−	−	−
	CI-SMS-EMOA	+	·	−	+	+	+	+	−
	CI-SPEA2	+	+	·	+	+	+	+	+
	CI-NSGA-II-P	+	−	−	·	+	+	+	−
	CI-SMS-EMOA-P	−	−	−	−	·	−	−	−
	CI-SPEA2-P	+	−	−	−	+	·	−	−
	R-NSGA-II	+	−	−	−	+	+	·	−
	WHYPE	+	+	−	+	+	+	+	·
WFG3	CI-NSGA-II	·	−	+	−	−	+	+	−
	CI-SMS-EMOA	+	·	+	+	+	+	+	+
	CI-SPEA2	−	−	·	−	−	+	+	−
	CI-NSGA-II-P	+	−	+	·	+	+	+	+
	CI-SMS-EMOA-P	+	−	+	−	·	−	−	−
	CI-SPEA2-P	−	−	−	−	−	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	+	−	+	−	−	+	+	·
WFG4	CI-NSGA-II	·	−	−	+	−	+	+	−
	CI-SMS-EMOA	+	·	−	+	+	+	+	−
	CI-SPEA2	+	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	−	−	·	−	+	+	−
	CI-SMS-EMOA-P	+	−	−	+	·	+	+	−
	CI-SPEA2-P	−	−	−	−	−	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	+	+	−	+	+	+	+	·
WFG5	CI-NSGA-II	·	−	−	+	−	−	−	−
	CI-SMS-EMOA	+	·	−	+	−	+	−	−
	CI-SPEA2	+	+	·	+	−	+	−	−
	CI-NSGA-II-P	−	−	−	·	−	−	−	−
	CI-SMS-EMOA-P	+	+	+	+	·	+	−	−
	CI-SPEA2-P	+	−	−	+	−	·	−	−
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	+	+	+	+	+	+	−	·
WFG6	CI-NSGA-II	·	+	−	+	+	−	−	−
	CI-SMS-EMOA	−	·	−	+	+	−	−	−
	CI-SPEA2	+	+	·	+	+	−	−	−
	CI-NSGA-II-P	−	−	−	·	+	−	−	−
	CI-SMS-EMOA-P	−	−	−	−	·	−	−	−
	CI-SPEA2-P	+	+	+	+	+	·	+	−
	R-NSGA-II	+	+	+	+	+	−	·	−
	WHYPE	+	+	+	+	+	+	+	·
WFG7	CI-NSGA-II	·	−	−	+	+	−	+	−
	CI-SMS-EMOA	+	·	−	+	+	+	+	−
	CI-SPEA2	+	+	·	+	+	+	+	−
	CI-NSGA-II-P	−	−	−	·	−	−	+	−
	CI-SMS-EMOA-P	−	−	−	+	·	−	+	−
	CI-SPEA2-P	+	−	−	+	+	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	+	+	+	+	+	+	+	·
WFG8	CI-NSGA-II	·	+	−	+	+	−	−	−
	CI-SMS-EMOA	−	·	−	+	+	−	−	−
	CI-SPEA2	+	+	·	+	+	−	−	−
	CI-NSGA-II-P	−	−	−	·	−	−	−	−
	CI-SMS-EMOA-P	−	−	−	+	·	−	−	−
	CI-SPEA2-P	+	+	+	+	+	·	−	−
	R-NSGA-II	+	+	+	+	+	+	·	−
	WHYPE	+	+	+	+	+	+	+	·
WFG9	CI-NSGA-II	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	−	−	+	+	−
	CI-SPEA2	−	+	·	−	−	+	+	−
	CI-NSGA-II-P	−	+	+	·	−	+	+	−
	CI-SMS-EMOA-P	−	+	+	+	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	+	+	+	−	+	+	·

Ψ

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
WFG1	CI-NSGA-II	·	+	−	+	−	−	+	+
	CI-SMS-EMOA	−	·	−	−	−	−	+	+
	CI-SPEA2	+	+	·	+	+	−	+	+
	CI-NSGA-II-P	−	+	−	·	−	−	+	+
	CI-SMS-EMOA-P	+	+	−	+	·	−	+	+
	CI-SPEA2-P	+	+	+	+	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG2	CI-NSGA-II	·	−	−	+	−	−	+	+
	CI-SMS-EMOA	+	·	−	+	+	−	+	+
	CI-SPEA2	+	+	·	+	+	−	+	+
	CI-NSGA-II-P	−	−	−	·	−	−	+	+
	CI-SMS-EMOA-P	+	−	−	+	·	−	+	+
	CI-SPEA2-P	+	+	+	+	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG3	CI-NSGA-II	·	∼	∼	∼	∼	∼	∼	+
	CI-SMS-EMOA	∼	·	∼	∼	∼	∼	∼	+
	CI-SPEA2	∼	∼	·	∼	∼	∼	∼	+
	CI-NSGA-II-P	∼	∼	∼	·	∼	∼	∼	+
	CI-SMS-EMOA-P	∼	∼	∼	∼	·	∼	∼	+
	CI-SPEA2-P	∼	∼	∼	∼	∼	·	∼	+
	R-NSGA-II	∼	∼	∼	∼	∼	∼	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG4	CI-NSGA-II	·	+	−	+	+	+	−	+
	CI-SMS-EMOA	−	·	−	+	−	−	−	+
	CI-SPEA2	+	+	·	+	+	+	−	+
	CI-NSGA-II-P	−	−	−	·	−	−	−	+
	CI-SMS-EMOA-P	−	+	−	+	·	−	−	+
	CI-SPEA2-P	−	+	−	+	+	·	−	+
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG5	CI-NSGA-II	·	+	+	+	+	−	−	+
	CI-SMS-EMOA	−	·	−	+	−	−	−	+
	CI-SPEA2	−	+	·	+	+	−	−	+
	CI-NSGA-II-P	−	−	−	·	−	−	−	+
	CI-SMS-EMOA-P	−	+	−	+	·	−	−	+
	CI-SPEA2-P	+	+	+	+	+	·	−	+
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG6	CI-NSGA-II	·	+	+	+	+	−	+	+
	CI-SMS-EMOA	−	·	−	−	−	−	+	+
	CI-SPEA2	−	+	·	+	+	−	+	+
	CI-NSGA-II-P	−	+	−	·	+	−	+	+
	CI-SMS-EMOA-P	−	+	−	−	·	−	+	+
	CI-SPEA2-P	+	+	+	+	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG7	CI-NSGA-II	·	+	∼	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	+	−	−	+	+
	CI-SPEA2	∼	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	−	−	·	−	−	+	+
	CI-SMS-EMOA-P	−	−	−	−	·	−	+	+
	CI-SPEA2-P	−	+	−	+	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG8	CI-NSGA-II	·	−	−	−	−	−	−	+
	CI-SMS-EMOA	+	·	−	−	−	−	−	+
	CI-SPEA2	+	+	·	−	−	−	−	+
	CI-NSGA-II-P	+	+	+	·	−	−	−	+
	CI-SMS-EMOA-P	+	+	+	+	·	+	+	+
	CI-SPEA2-P	+	+	+	+	−	·	−	+
	R-NSGA-II	+	+	+	+	−	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG9	CI-NSGA-II	·	−	−	+	+	+	+	+
	CI-SMS-EMOA	+	·	−	+	+	+	+	+
	CI-SPEA2	+	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	−	−	·	−	−	−	+
	CI-SMS-EMOA-P	−	−	−	+	·	+	−	+
	CI-SPEA2-P	−	−	−	+	−	·	−	+
	R-NSGA-II	−	−	−	+	+	+	·	+
	WHYPE	−	−	−	−	−	−	−	·

κ

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
WFG1	CI-NSGA-II	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	+	+	+	+	−
	CI-SPEA2	−	+	·	+	+	+	+	−
	CI-NSGA-II-P	−	−	−	·	+	−	+	−
	CI-SMS-EMOA-P	−	−	−	−	·	−	+	−
	CI-SPEA2-P	−	−	−	+	+	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	+	+	+	+	+	+	·
WFG2	CI-NSGA-II	·	+	+	+	−	+	+	+
	CI-SMS-EMOA	−	·	+	−	−	−	−	+
	CI-SPEA2	−	−	·	−	−	−	−	+
	CI-NSGA-II-P	−	+	+	·	−	−	+	+
	CI-SMS-EMOA-P	+	+	+	+	+	·	+	+
	CI-SPEA2-P	−	+	+	+	−	·	+	+
	R-NSGA-II	−	+	+	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG3	CI-NSGA-II	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	−	·	+	+	+	+	+	+
	CI-SPEA2	−	−	·	+	+	+	+	+
	CI-NSGA-II-P	−	−	−	·	−	+	−	+
	CI-SMS-EMOA-P	−	−	−	+	·	+	−	+
	CI-SPEA2-P	−	−	−	−	−	·	−	−
	R-NSGA-II	−	−	−	+	+	+	·	+
	WHYPE	−	−	−	−	−	+	−	·
WFG4	CI-NSGA-II	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	−	+	−	+	+
	CI-SPEA2	−	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	+	−	·	+	+	+	+
	CI-SMS-EMOA-P	−	−	−	−	·	−	+	+
	CI-SPEA2-P	−	+	−	−	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·
WFG5	CI-NSGA-II	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	+	+	+	+	+
	CI-SPEA2	−	+	+	+	+	+	+	+
	CI-NSGA-II-P	−	−	−	·	−	+	+	+
	CI-SMS-EMOA-P	−	−	−	+	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·
WFG6	CI-NSGA-II	·	+	−	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	+	+	+	+	+
	CI-SPEA2	+	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	−	−	·	−	+	+	+
	CI-SMS-EMOA-P	−	−	−	+	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG7	CI-NSGA-II	·	−	−	−	−	−	−	−
	CI-SMS-EMOA	−	·	−	−	+	−	+	+
	CI-SPEA2	−	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	+	−	·	+	+	+	+
	CI-SMS-EMOA-P	−	−	−	−	·	−	+	+
	CI-SPEA2-P	−	+	−	−	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG8	CI-NSGA-II	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	−	−	+	+	+
	CI-SPEA2	−	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	+	−	·	−	+	+	+
	CI-SMS-EMOA-P	−	+	−	+	·	+	+	+
	CI-SPEA2-P	−	−	−	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG9	CI-NSGA-II	·	+	+	+	+	+	∼	+
	CI-SMS-EMOA	−	·	−	+	+	+	∼	+
	CI-SPEA2	−	+	·	+	+	+	∼	+
	CI-NSGA-II-P	−	−	−	·	+	+	∼	+
	CI-SMS-EMOA-P	−	−	−	−	·	+	∼	+
	CI-SPEA2-P	−	−	−	−	−	·	∼	+
	R-NSGA-II	∼	∼	∼	∼	∼	∼	·	∼
	WHYPE	−	−	−	−	−	−	∼	·

R-metric

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
WFG1	CI-NSGA-II	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	−	·	+	+	+	+	∼	+
	CI-SPEA2	−	−	·	−	∼	+	−	+
	CI-NSGA-II-P	−	−	+	·	+	+	+	+
	CI-SMS-EMOA-P	−	−	∼	−	·	+	−	+
	CI-SPEA2-P	−	−	−	−	−	·	−	−
	R-NSGA-II	−	∼	+	−	+	+	·	+
	WHYPE	−	−	−	−	−	−	+	·
WFG2	CI-NSGA-II	·	−	−	+	+	−	+	+
	CI-SMS-EMOA	+	·	∼	+	+	−	+	+
	CI-SPEA2	+	∼	·	+	+	−	+	+
	CI-NSGA-II-P	−	−	−	·	−	−	+	+
	CI-SMS-EMOA-P	−	−	−	+	·	−	+	+
	CI-SPEA2-P	+	+	+	+	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·
WFG3	CI-NSGA-II	·	+	−	+	+	+	+	+
	CI-SMS-EMOA	−	·	−	−	+	+	−	+
	CI-SPEA2	+	+	·	+	+	+	+	+
	CI-NSGA-II-P	−	+	−	·	+	+	−	+
	CI-SMS-EMOA-P	−	−	−	−	·	+	−	+
	CI-SPEA2-P	−	−	−	−	−	·	−	+
	R-NSGA-II	−	+	−	+	+	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG4	CI-NSGA-II	·	−	+	−	−	−	−	−
	CI-SMS-EMOA	+	·	+	−	−	−	−	+
	CI-SPEA2	−	−	·	−	−	−	−	−
	CI-NSGA-II-P	+	+	+	·	+	+	+	+
	CI-SMS-EMOA-P	+	+	+	−	·	−	+	+
	CI-SPEA2-P	+	+	+	−	+	·	+	+
	R-NSGA-II	+	+	+	−	−	−	·	+
	WHYPE	+	−	+	−	−	−	−	·
WFG5	CI-NSGA-II	·	+	∼	−	−	−	−	−
	CI-SMS-EMOA	−	·	−	−	−	−	−	−
	CI-SPEA2	∼	+	·	−	−	−	−	−
	CI-NSGA-II-P	+	+	+	·	+	−	+	+
	CI-SMS-EMOA-P	+	+	+	−	·	−	+	+
	CI-SPEA2-P	+	+	+	+	+	·	+	+
	R-NSGA-II	+	+	+	−	−	−	·	−
	WHYPE	+	+	+	−	−	−	−	·
WFG6	CI-NSGA-II	·	−	+	−	−	−	−	+
	CI-SMS-EMOA	+	·	+	−	−	−	−	+
	CI-SPEA2	−	−	·	−	−	−	−	+
	CI-NSGA-II-P	+	+	+	·	+	∼	+	+
	CI-SMS-EMOA-P	+	+	+	−	·	−	+	+
	CI-SPEA2-P	+	+	+	∼	+	·	+	+
	R-NSGA-II	+	+	+	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG7	CI-NSGA-II	·	+	+	−	−	−	+	+
	CI-SMS-EMOA	−	·	−	−	−	−	+	−
	CI-SPEA2	−	+	·	−	−	−	+	+
	CI-NSGA-II-P	+	+	+	·	+	−	+	+
	CI-SMS-EMOA-P	+	+	+	−	·	−	+	+
	CI-SPEA2-P	+	+	+	+	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	+	−	−	−	−	+	·
WFG8	CI-NSGA-II	·	−	+	−	−	−	−	+
	CI-SMS-EMOA	+	·	+	−	−	−	−	+
	CI-SPEA2	−	−	·	−	−	−	−	+
	CI-NSGA-II-P	+	+	+	·	+	+	−	+
	CI-SMS-EMOA-P	+	+	+	−	·	−	−	+
	CI-SPEA2-P	+	+	+	−	+	·	−	+
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG9	CI-NSGA-II	·	+	+	−	−	−	−	−
	CI-SMS-EMOA	−	·	−	−	−	−	−	−
	CI-SPEA2	−	+	·	−	−	−	−	−
	CI-NSGA-II-P	+	+	+	·	+	−	−	+
	CI-SMS-EMOA-P	+	+	+	−	·	−	−	+
	CI-SPEA2-P	+	+	+	+	+	·	+	+
	R-NSGA-II	+	+	+	+	+	−	·	+
	WHYPE	+	+	+	−	−	−	−	·

$C2_R$

		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
WFG1	CI-NSGA-II	·	+	+	+	+	+	−	+
	CI-SMS-EMOA	−	·	−	+	+	−	−	+
	CI-SPEA2	−	+	·	+	+	+	−	+
	CI-NSGA-II-P	−	−	−	·	+	−	−	+
	CI-SMS-EMOA-P	−	−	−	−	·	−	−	+
	CI-SPEA2-P	−	+	−	+	+	·	−	+
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG2	CI-NSGA-II	·	+	+	−	+	+	+	+
	CI-SMS-EMOA	−	·	+	−	−	−	+	+
	CI-SPEA2	−	−	·	−	−	−	+	+
	CI-NSGA-II-P	+	+	+	·	+	+	+	+
	CI-SMS-EMOA-P	−	+	+	−	·	+	+	+
	CI-SPEA2-P	−	+	+	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·
WFG3	CI-NSGA-II	·	+	−	−	−	+	+	+
	CI-SMS-EMOA	−	·	−	−	−	−	+	+
	CI-SPEA2	+	+	·	−	−	+	+	+
	CI-NSGA-II-P	+	+	+	·	−	+	+	+
	CI-SMS-EMOA-P	+	+	+	+	·	+	+	+
	CI-SPEA2-P	−	+	−	−	−	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·
WFG4	CI-NSGA-II	·	+	+	−	−	−	−	+
	CI-SMS-EMOA	−	·	−	−	−	−	−	+
	CI-SPEA2	−	+	·	−	−	−	−	+
	CI-NSGA-II-P	+	+	+	·	+	+	−	+
	CI-SMS-EMOA-P	+	+	+	−	·	−	−	+
	CI-SPEA2-P	+	+	+	−	+	·	−	+
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG5	CI-NSGA-II	·	+	−	−	−	−	+	+
	CI-SMS-EMOA	−	·	−	−	−	−	+	+
	CI-SPEA2	+	+	·	−	−	−	+	+
	CI-NSGA-II-P	+	+	+	·	+	−	+	+
	CI-SMS-EMOA-P	+	+	+	−	·	−	+	+
	CI-SPEA2-P	+	+	+	+	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·
WFG6	CI-NSGA-II	·	+	−	−	−	−	+	+
	CI-SMS-EMOA	−	·	−	−	−	−	+	+
	CI-SPEA2	+	+	·	−	−	−	+	+
	CI-NSGA-II-P	+	+	+	·	+	+	+	+
	CI-SMS-EMOA-P	+	+	+	−	·	−	+	+
	CI-SPEA2-P	+	+	+	−	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−
	WHYPE	−	−	−	−	−	−	+	·
WFG7	CI-NSGA-II	·	+	+	−	−	−	+	+
	CI-SMS-EMOA	−	·	−	−	−	−	−	+
	CI-SPEA2	−	+	·	−	−	−	+	+
	CI-NSGA-II-P	+	+	+	·	+	−	+	+
	CI-SMS-EMOA-P	+	+	+	−	·	−	+	+
	CI-SPEA2-P	+	+	+	+	+	·	+	+
	R-NSGA-II	−	+	−	−	−	−	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG8	CI-NSGA-II	·	+	+	−	−	−	−	+
	CI-SMS-EMOA	−	·	−	−	−	−	−	+
	CI-SPEA2	−	+	·	−	−	−	−	+
	CI-NSGA-II-P	+	+	+	·	+	+	−	+
	CI-SMS-EMOA-P	+	+	+	−	·	−	−	+
	CI-SPEA2-P	+	+	+	−	+	·	−	+
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	−	−	−	−	−	−	·
WFG9	CI-NSGA-II	·	+	+	−	−	−	−	+
	CI-SMS-EMOA	−	·	−	−	−	−	−	+
	CI-SPEA2	−	+	·	−	−	−	−	+
	CI-NSGA-II-P	+	+	+	·	+	−	−	+
	CI-SMS-EMOA-P	+	+	+	−	·	−	−	+
	CI-SPEA2-P	+	+	+	+	+	·	−	+
	R-NSGA-II	+	+	+	+	+	+	·	+
	WHYPE	−	−	−	−	−	−	−	·

Table 6.5: WFG with 5 objectives results of the Conover-Inman statistical hypothesis [53] tests based on the mean values. The preference-based algorithms are compared by the performance indicators used in the experiments. Green cells (+) denote cases where the algorithm in the row statistically was better than the one in the column. Cells marked in red (−) are cases where the method in the column yielded statistically better results when compared to the method in the row. Cases where no significant difference was established are identified with a "∼".

		Δ_P								Ψ								κ								R-metric								$C2_R$							
		CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE	CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE	CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE	CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE	CI-NSGA-II	CI-SMS-EMOA	CI-SPEA2	CI-NSGA-II-P	CI-SMS-EMOA-P	CI-SPEA2-P	R-NSGA-II	WHYPE
WFG1 5D	CI-NSGA-II	·	−	−	−	−	−	+	−	·	+	−	−	+	+	+	−	·	+	+	+	+	+	+	−	·	+	+	+	+	+	+	+	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	+	·	−	−	−	−	+	−	−	·	−	−	+	+	+	−	−	·	−	+	+	+	+	−	−	·	−	+	+	+	+	+	−	·	−	+	+	+	+	+
	CI-SPEA2	+	+	·	+	+	−	+	+	+	+	·	+	+	+	+	+	−	+	·	+	+	+	+	−	−	+	·	+	+	+	+	+	−	+	·	+	+	+	+	+
	CI-NSGA-II-P	+	+	−	·	−	−	+	−	+	+	−	·	+	+	+	+	−	−	−	·	+	+	+	−	−	−	−	·	+	+	+	−	−	−	−	·	+	+	+	−
	CI-SMS-EMOA-P	+	+	−	+	·	−	+	+	−	−	−	−	·	+	+	−	−	−	−	−	·	+	+	−	−	−	−	−	·	+	+	−	−	−	−	−	·	+	+	−
	CI-SPEA2	+	+	+	+	+	·	+	+	−	−	−	−	−	·	+	−	−	−	−	−	−	·	+	−	−	−	−	−	−	·	+	−	−	−	−	−	−	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−	−	−	−	−	−	−	·	−	−	−	−	−	−	−	·	−	−	−	−	−	−	−	·	−	−	−	−	−	−	−	·	−
	WHYPE	+	+	−	+	−	−	+	·	+	+	−	−	+	+	+	·	+	+	+	+	+	+	+	·	−	−	−	+	+	+	+	·	−	−	−	+	+	+	+	·
WFG4 5D	CI-NSGA-II	·	−	−	−	−	−	−	−	·	+	−	−	+	−	+	+	·	+	+	+	+	+	+	+	·	+	+	−	+	+	+	+	·	+	+	−	+	+	+	+
	CI-SMS-EMOA	+	·	−	+	−	−	+	−	−	·	−	−	+	−	+	−	−	·	−	−	+	+	+	−	−	·	−	−	+	+	+	−	−	·	−	−	+	+	+	−
	CI-SPEA2	+	+	·	+	+	+	+	+	+	+	·	+	+	+	+	+	−	+	·	−	+	+	+	+	−	+	·	−	+	+	+	−	−	+	·	−	+	+	+	+
	CI-NSGA-II-P	+	−	−	·	−	−	−	−	+	+	−	·	+	−	+	+	−	+	+	·	+	+	+	+	+	+	+	·	+	+	+	+	+	+	+	·	+	+	+	+
	CI-SMS-EMOA-P	+	+	−	+	·	−	+	+	−	−	−	−	·	−	+	−	−	−	−	−	·	+	+	−	−	−	−	−	·	+	+	−	−	−	−	−	·	+	+	−
	CI-SPEA2	+	+	−	+	+	·	+	+	+	+	−	+	+	·	+	+	−	−	−	−	−	·	−	−	−	−	−	−	−	·	+	−	−	−	−	−	−	·	+	−
	R-NSGA-II	+	−	−	+	−	−	·	−	−	−	−	−	−	−	·	−	−	−	−	−	−	+	·	−	−	−	−	−	−	−	·	−	−	−	−	−	−	−	·	−
	WHYPE	+	+	−	+	−	−	+	·	−	+	−	−	+	−	+	·	−	+	−	−	+	+	+	·	−	+	+	−	+	+	+	·	−	+	−	−	+	+	+	·
WFG6 5D	CI-NSGA-II	·	−	−	−	−	−	+	−	·	+	−	+	−	−	+	+	·	+	−	+	+	+	+	+	·	+	−	+	+	+	+	+	·	+	−	+	+	+	+	+
	CI-SMS-EMOA	+	·	−	−	−	−	+	−	−	·	−	+	−	−	+	+	−	·	−	+	+	+	+	+	−	·	−	+	+	+	+	+	−	·	−	+	+	+	+	+
	CI-SPEA2	+	+	·	+	+	+	+	+	+	+	·	+	−	+	+	+	+	+	·	+	+	+	+	+	+	+	·	+	+	+	+	+	+	+	·	+	+	+	+	+
	CI-NSGA-II-P	+	+	−	·	−	−	+	−	−	−	−	·	−	−	+	−	−	−	−	·	+	−	−	−	−	−	−	·	+	−	−	−	−	−	−	·	+	−	+	−
	CI-SMS-EMOA-P	+	+	−	+	·	−	+	−	+	+	+	+	·	+	+	+	−	−	−	−	·	−	−	−	−	−	−	−	·	−	−	−	−	−	−	−	·	−	−	−
	CI-SPEA2	+	+	−	+	+	·	+	+	+	+	−	+	−	·	+	+	−	−	−	+	+	·	−	−	−	−	−	+	+	·	−	−	−	−	−	+	+	·	+	−
	R-NSGA-II	−	−	−	−	−	−	·	−	−	−	−	−	−	−	·	−	−	−	−	+	+	+	·	−	−	−	−	+	+	+	·	−	−	−	−	−	+	−	·	−
	WHYPE	+	+	−	+	+	−	+	·	−	−	−	+	−	−	+	·	−	−	−	+	+	+	+	·	−	−	−	+	+	+	+	·	−	−	−	+	+	+	+	·
WFG7 5D	CI-NSGA-II	·	+	+	+	∼	∼	+	+	·	+	∼	+	∼	∼	∼	+	·	+	+	+	+	−	+	+	·	+	+	+	+	+	+	+	·	+	+	+	+	+	∼	+
	CI-SMS-EMOA	−	·	−	∼	−	−	+	+	−	·	−	+	−	−	−	+	−	·	−	−	+	−	+	−	−	·	−	+	+	−	+	−	−	·	−	−	+	−	−	−
	CI-SPEA2	−	+	·	+	−	−	+	+	∼	+	·	+	∼	∼	∼	+	−	+	·	+	+	−	+	+	−	+	·	+	+	−	+	−	−	+	·	+	+	−	−	+
	CI-NSGA-II-P	−	∼	−	·	−	−	+	+	−	−	−	·	−	−	−	+	−	+	−	·	+	−	+	−	−	−	−	·	+	−	∼	−	−	+	−	·	+	−	−	−
	CI-SMS-EMOA-P	∼	+	+	∼	·	∼	+	+	∼	+	∼	+	·	∼	∼	+	−	−	−	−	·	−	−	−	−	−	−	−	·	−	−	−	−	−	−	−	·	−	−	−
	CI-SPEA2	∼	+	+	+	∼	·	+	+	∼	+	∼	+	∼	·	∼	+	+	+	+	+	+	·	+	+	−	+	+	+	+	·	+	+	−	+	+	+	+	·	+	+
	R-NSGA-II	−	−	−	−	−	−	·	−	∼	+	∼	+	∼	∼	·	+	−	−	−	−	+	−	·	−	−	−	−	∼	+	−	·	−	∼	+	+	+	+	+	·	+
	WHYPE	−	−	−	−	−	−	+	·	−	−	−	−	−	−	−	·	−	+	−	+	+	−	+	·	−	+	+	+	+	−	+	·	−	+	−	+	+	−	−	·
WFG8 5D	CI-NSGA-II	·	−	−	−	−	−	−	−	·	+	+	+	+	−	+	+	·	+	+	+	+	+	+	+	·	−	−	−	+	−	−	−	·	+	+	+	+	+	+	+
	CI-SMS-EMOA	+	·	−	−	−	−	−	−	−	·	−	+	+	−	+	+	−	·	−	+	+	+	−	+	+	·	−	+	+	+	−	+	−	·	−	+	+	+	−	+
	CI-SPEA2	+	+	·	+	−	+	+	−	−	+	·	+	+	−	+	+	−	+	·	+	+	+	−	+	+	+	·	+	+	+	−	+	−	+	·	+	+	+	−	+
	CI-NSGA-II-P	+	+	−	·	−	−	−	−	−	−	−	·	−	−	+	−	−	−	−	·	+	−	−	−	+	−	−	·	+	−	−	−	−	−	−	·	+	−	−	−
	CI-SMS-EMOA-P	+	+	+	+	·	+	+	+	−	−	−	+	·	−	+	−	−	−	−	−	·	−	−	−	−	−	−	−	·	−	−	−	−	−	−	−	·	−	−	−
	CI-SPEA2	+	+	−	+	−	·	−	−	+	+	+	+	+	·	+	+	−	−	−	+	+	·	−	−	+	−	−	+	+	·	−	−	−	−	−	+	+	·	−	−
	R-NSGA-II	+	+	−	+	−	+	·	−	−	−	−	−	−	−	·	−	−	+	+	+	+	+	·	+	+	+	+	+	+	+	·	+	+	+	+	+	+	+	·	+
	WHYPE	+	+	+	+	−	+	+	·	−	−	−	+	+	−	+	·	−	−	−	+	+	+	−	·	+	−	−	+	+	+	−	·	−	−	−	+	+	+	−	·
WFG9 5D	CI-NSGA-II	·	−	−	−	−	+	+	+	·	+	∼	∼	+	+	+	+	·	+	+	+	+	+	+	+	·	+	+	+	+	+	−	+	·	+	+	+	+	+	−	+
	CI-SMS-EMOA	+	·	−	−	−	+	+	+	−	·	−	−	+	+	+	+	−	·	−	+	+	+	+	+	−	·	−	+	+	+	−	+	−	·	−	+	+	+	−	+
	CI-SPEA2	+	+	·	∼	+	+	+	+	∼	+	·	∼	+	+	+	+	−	+	·	+	+	+	+	+	−	+	·	+	+	+	−	+	−	+	·	+	+	+	−	+
	CI-NSGA-II-P	+	+	∼	·	+	+	+	+	∼	+	∼	·	+	+	+	+	−	−	−	·	+	+	+	−	−	−	−	·	+	+	−	+	−	−	−	·	+	+	−	+
	CI-SMS-EMOA-P	+	+	−	−	·	+	+	+	−	−	−	−	·	+	+	+	−	−	−	−	·	+	−	−	−	−	−	−	·	+	−	−	−	−	−	−	·	+	−	−
	CI-SPEA2	−	−	−	−	−	·	+	−	−	−	−	−	−	·	+	+	−	−	−	−	−	·	−	−	−	−	−	−	−	·	−	−	−	−	−	−	−	·	−	−
	R-NSGA-II	−	−	−	−	−	−	·	−	−	−	−	−	−	−	·	−	−	−	−	−	+	+	·	−	+	+	+	+	+	+	·	+	+	+	+	+	+	+	·	+
	WHYPE	−	−	−	−	+	+	+	·	−	−	−	−	−	−	+	·	−	−	−	+	+	+	+	·	−	−	−	−	+	+	−	·	−	−	−	−	+	+	−	·

6.1.5 Discussion of the Results

It must be emphasized that the experiments compare the performance of two *a priori* preference-based algorithms with the COIN-based interactive algorithms. The final result from these tests are analysed by the convergence and extension of the preferred region in the trade-off set. However, the R-NSGA-II and W-HYPE has a predefined reference point, whereas the COIN MOEAs have to discover the collective reference point at each interaction. The COIN-based algorithms collect the DMs' intention during the optimization process. This feature enhances the application of the *a priori* algorithms to problems where no earlier information is known or available.

In the ZDT tests, the CI-SPEA2-P managed to find a better convergence ($D_{S \to P_F}$) in 3 of the 5 problems, followed by WHYPE and R-NSGA with 2 victories each. Regarding the Filatovas indicator (Δ_P), the CI-NSGA-II won 3 first places against only 1 first place to R-NSGA and WHYPE. The CI-NSGA-II-P consistently outperformed the others algorithms in the Convex Hull indicator (Ψ) with 4 first places. It demonstrates how the CI-NSGA-II-P solutions are well concentrated on the reference point area. In the Referential Cluster Variance indicator (κ), the CI-SPEA2 and CI-SMS-EMOA showed equivalent score: 4 wins; followed by R-NSGA on the ZDT4 problem. The CI-SMS-EMOA-P had the best performance regarding the R-metric indicator, followed by 1 first place to each of the algorithms: CI-NSGA-II, R-NSGA-II and CI-SPEA2. Finally, the winners on the Reference Set Distance indicator ($C2_R$) were the new COIN-based algorithms for all of the 5 ZDT problems.

By grouping all the indicators, the COIN-based MOEAs had the best results on 70% of the ZDT problems. The WHYPE and R-NSGA-II got first places in only 30% of them. In the general case, when the collective reference point is used, the algorithms get more victories than the classic ones.

In the DTLZ tests, the CI-SMS-EMOA-P and the R-NSGA-II had the best approximation to the P_F with 2 first places each. The WHYPE won 3 second places on this indicator. Regarding the Filatovas indicator, the CI-SPEA2 outperformed the others algorithms in 4 problems. The first places were distributed between the R-NSGA-II and the CI-NSGA-II in the Referential Cluster Variance indicator, both with 3 victories. The R-metric indicator winners were: WHYPE, R-NSGA-II and CI-SMS-EMOA. The CI-NSGA-II won 3 second places on this indicator. Regarding the Reference Set Distance indicator, the CI-NSGA-II and CI-NSGA-II-P had the best performance with 3 first places and 2 first places, respectively.

Similarly to the ZDT case, after grouping all the indicators, the COIN-based MOEAs had the best results on 76% of the DTLZ problems. The WHYPE and R-NSGA-II got first places in only 24% of them.

In the WFG tests, the CI-SMS-EMOA and CI-SPEA2-P managed to find a better convergence with 2 first places each, followed by CI-NSGA-II. Regarding the Filatovas indicator, the WHYPE won 4 first places and 3 second places. The CI-NSGA-II consistently outperformed the others algorithms in the Referential Cluster Variance indicator with 7 victories. In the R-metric indicator, the algorithms with the collective reference point projection had a better efficiency. The CI-NSGA-II-P and CI-SPEA2 showed equivalent score: 2 wins and 3 second places each. Finally, the winner on the Reference Set Distance indicator was the R-NSGA-II with 4 first places, followed by the CI-NSGA-II-P with 2 first places and 5 second places.

By grouping all the indicators, the COIN-based MOEAs had the best results on 72% of the WFG problems. The WHYPE and R-NSGA-II got first places in only 28% of them.

MOPs having more than three objectives are referred to as many-objective optimization problems. Pareto dominance-based algorithms such as NSGA-II and SPEA2 usually have a good performance on multi-objective problems with two or three objectives. But an increase in the number of objectives degrades their search ability. This is because almost all points in the current population become non-dominated in early generations and this undermines the selection pressure toward the P_F [99, 103].

However, the search for Pareto optimal solutions in many-objective problems is not always difficult for Pareto dominance-based algorithms [96,97]. In the WFG tests with five objectives, the CI-SPEA2 won 3 first places and 2 second places on the Filatovas indicator, followed by CI-SPEA2-P with 2 first places. The CI-SPEA2 had the best performance on the Convex Hull indicator too, 4 victories. The CI-NSGA-II outperformed the others algorithms in the Referential Cluster Variance indicator and R-metric indicator with 3 first places and 2 second places, respectively. The winner on the Reference Set Distance indicator was the R-NSGA-II with 3 first places, followed by CI-NSGA-II with 2 victories.

By grouping all the indicators, the COIN-based MOEAs had the best results on 80% of the WFG problems with 5 objectives. The WHYPE and R-NSGA-II got first places in only 20% of them.

The process of discovering the best algorithm is rather difficult as it implies cross-examining and comparing the results of their performance indicators. Figures 6.6-6.9

present a more integrative representation by grouping their indicators.

A higher value of average performance ranking implies that the algorithm consistently achieved lower values of the indicators being assessed: $D_{S \to P_F}$, Δ_P, $C2_R$, Ψ, κ; and higher values for the R-metric indicator. In this case, lower values mean better convergence to P_F, higher concentration around the collective reference points and small distance to the collective reference point.

For a given set of algorithms $A_1, \ldots, A_K$, a set of P test problem instances $\Phi_1, \ldots, \Phi_P$, the function δ is defined as:

$$\delta\left(A_i, A_j, \Phi_p\right) = \begin{cases} 1 & \text{if } A_i \gg A_j \text{ solving } \Phi_p \\ 0 & \text{otherwise} \end{cases} \tag{6.1}$$

where the relation $A_i \gg A_j$ defines if A_i is better than A_j when solving the problem instance Φ_p in terms of the performance indicators: $D_{S \to P_F}$, κ and Ψ. Relying on δ, the performance index $P_p(A_i)$ of a given algorithm A_i when solving Φ_p is then computed as: $P_p(A_i) = \sum_{j=1, j \neq i}^{K} \delta\left(A_i, A_j, \Phi_p\right)$.

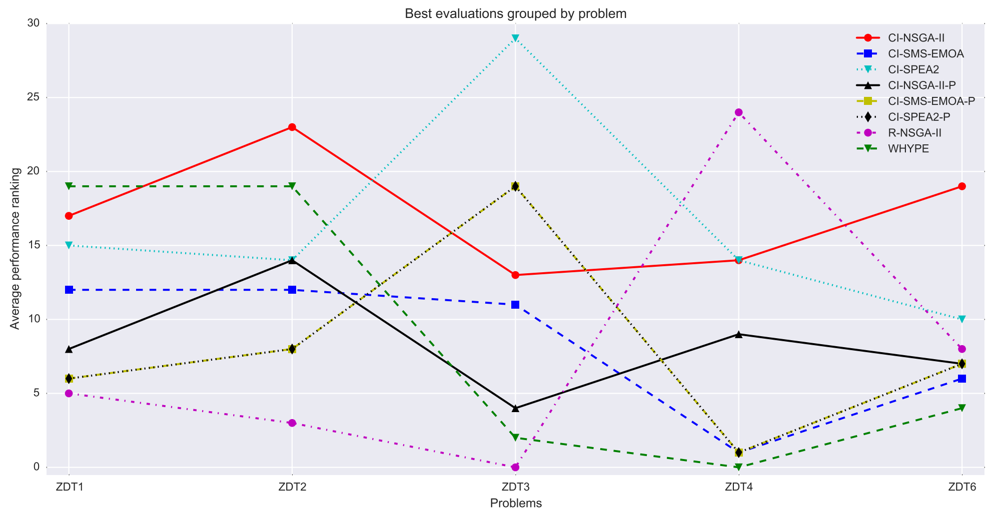


Figure 6.6: Average performance ranking across ZDT test problems.

Considering the Figures 6.6, 6.7, 6.8 and 6.9, the CI-NSGA-II obtained the largest number of first places in this integrative representation of the indicators. In other words, the CI-NSGA-II performed better in most of the problems and indicators among all the 8 preference-based algorithms tested. It appears between the first and third places in 64% of the analysis. Along with the the CI-NSGA-II-P, they cover 77% of the 3 first positions

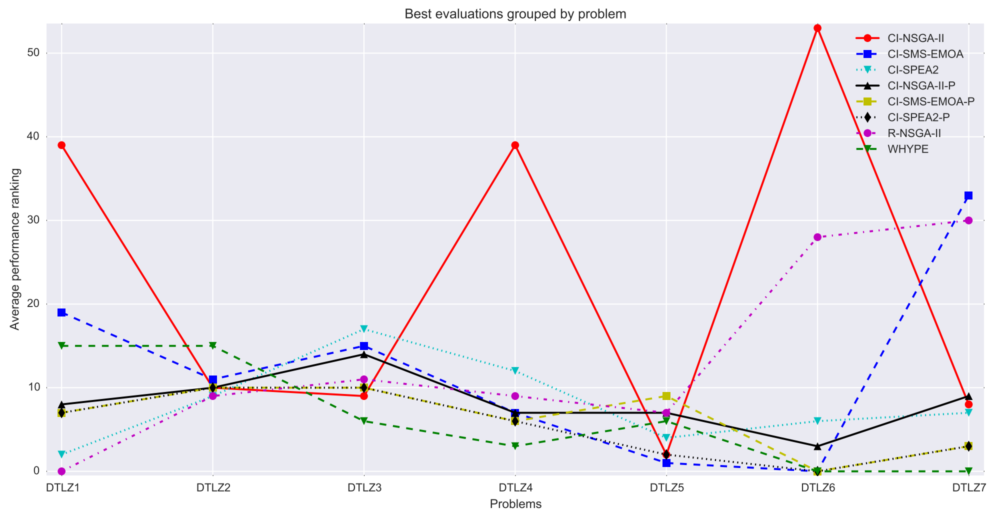


Figure 6.7: Average performance ranking across DTLZ test problems.

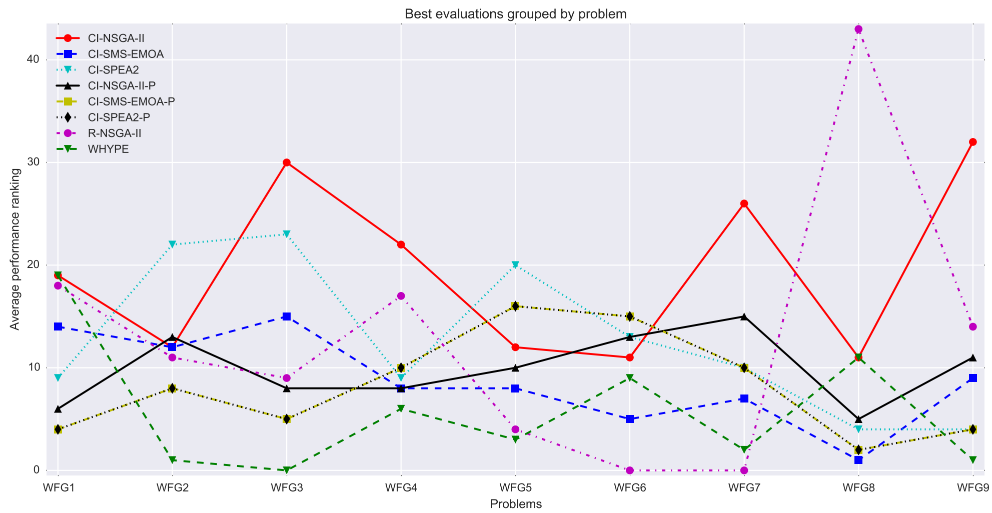


Figure 6.8: Average performance ranking across WFG test problems.

in the benchmark problems.

It is worth mentioning that the algorithms with the reference point projection performed close to those without this approach. This behaviour grants the use of the reference point projection in cases where the human interactions are constrained for some reason.

In summary, the interactive MOEAs and their reference points proved to be well matched for the range of scalable test problems. According the Tables 6.6-6.9, the R-NSGA-II outperformed the WHYPE with more first places. The algorithms built upon NSGA-II performed better and, particularly, the CI-NSGA-II had the best results. Besides, based on the Reference Set Distance indicator that expresses the mean distance

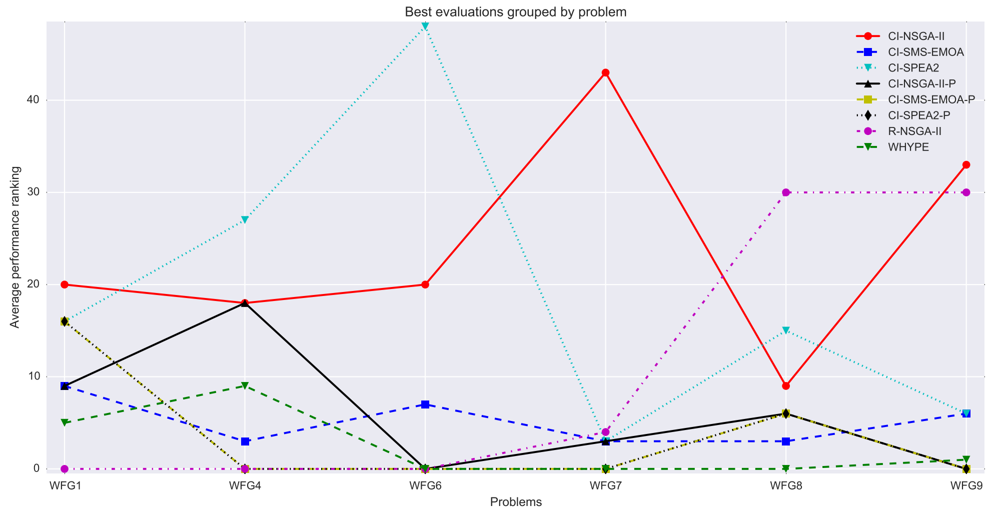


Figure 6.9: Average performance ranking across five-objective WFG test problems.

between the non-dominated points and the reference set, the proposed COIN MOEAs dominated the first places of all test problems.

Tables 6.6-6.9 present the mean values (μ) and standard deviation (σ) of the performance indicators for all the benchmark problems.

Table 6.6: Mean values of the performance indicators for the ZDT problems and standard the deviation in parenthesis.

ZDT1	$D_{S \to P_F}$	Δ_P	Ψ	κ	R-metric	$C2_R$
CI-NSGA-II	0.014 (0.006)	0.877 (0.171)	0.004 (0.004)	0.011 (0.011)	**0.935** (0.001)	0.052 (0.053)
CI-SMS-EMOA	0.014 (0.003)	0.787 (0.199)	0.003 (0.002)	0.025 (0.017)	**0.935** (0.001)	0.110 (0.074)
CI-SPEA2	0.018 (0.005)	1.161 (0.390)	0.007 (0.011)	**0.003** (0.005)	0.928 (0.002)	**0.026** (0.018)
CI-NSGA-II-P	0.008 (0.002)	0.973 (0.241)	**0.002** (0.003)	0.036 (0.012)	0.934 (0.001)	0.154 (0.045)
CI-SMS-EMOA-P	0.009 (0.002)	0.977 (0.291)	0.004 (0.004)	0.069 (0.034)	0.932 (0.003)	0.254 (0.086)
CI-SPEA2-P	0.009 (0.002)	1.267 (0.429)	0.023 (0.023)	0.049 (0.020)	0.929 (0.002)	0.193 (0.066)
R-NSGA-II	0.242 (0.045)	**0.662** (0.055)	0.187 (0.054)	0.155 (0.046)	0.933 (0.002)	0.492 (0.083)
WHYPE	**0.005** (0.002)	1.002 (0.079)	0.120 (0.036)	0.063 (0.044)	0.925 (0.005)	0.199 (0.141)
ZDT2						
CI-NSGA-II	**0.005** (0.001)	1.023 (0.283)	**0.001** (0.001)	0.033 (0.063)	0.889 (0.004)	0.109 (0.208)
CI-SMS-EMOA	0.007 (0.003)	0.973 (0.176)	**0.001** (0.002)	0.111 (0.066)	0.895 (0.005)	0.385 (0.208)
CI-SPEA2	0.021 (0.015)	1.073 (0.490)	0.012 (0.020)	**0.018** (0.032)	0.883 (0.002)	**0.061** (0.092)
CI-NSGA-II-P	0.008 (0.003)	**0.972** (0.224)	**0.001** (0.006)	0.057 (0.074)	0.892 (0.004)	0.213 (0.229)
CI-SMS-EMOA-P	0.010 (0.006)	1.033 (0.187)	0.004 (0.004)	0.112 (0.081)	**0.896** (0.006)	0.379 (0.227)
CI-SPEA2-P	**0.005** (0.004)	1.373 (0.413)	0.014 (0.027)	0.078 (0.041)	0.886 (0.004)	0.341 (0.101)
R-NSGA-II	0.009 (0.004)	1.497 (0.307)	0.022 (0.027)	0.180 (0.092)	0.892 (0.009)	0.523 (0.196)
WHYPE	**0.005** (0.001)	1.108 (0.233)	0.260 (0.201)	0.074 (0.003)	0.882 (0.010)	0.237 (0.006)
ZDT3						
CI-NSGA-II	0.005 (0.006)	**1.119** (0.565)	0.002 (0.007)	0.010 (0.008)	0.938 (0.002)	0.157 (0.052)
CI-SMS-EMOA	0.009 (0.013)	1.497 (0.435)	0.004 (0.008)	**0.007** (0.004)	0.940 (0.003)	0.127 (0.066)
CI-SPEA2	0.002 (0.002)	1.297 (0.524)	0.034 (0.046)	0.010 (0.012)	0.939 (0.004)	**0.029** (0.033)
CI-NSGA-II-P	0.005 (0.006)	1.603 (0.526)	**0.001** (0.003)	0.013 (0.008)	0.940 (0.002)	0.165 (0.052)
CI-SMS-EMOA-P	0.002 (0.003)	1.318 (0.561)	**0.001** (0.001)	0.009 (0.007)	**0.941** (0.002)	0.092 (0.090)
CI-SPEA2-P	**0.001** (0.002)	1.495 (0.466)	0.055 (0.067)	0.061 (0.028)	0.927 (0.005)	0.259 (0.103)
R-NSGA-II	**0.001** (0.001)	1.789 (0.081)	0.003 (0.006)	0.074 (0.007)	0.921 (0.003)	0.275 (0.024)
WHYPE	0.054 (0.029)	1.132 (0.441)	0.118 (2.7e-17)	0.039 (0.016)	0.910 (0.005)	1.411 (0.031)
ZDT4						
CI-NSGA-II	1.047 (0.691)	1.421 (0.468)	0.003 (0.011)	0.547 (0.768)	0.793 (0.073)	1.206 (0.920)
CI-SMS-EMOA	1.449 (1.166)	1.447 (0.469)	0.041 (0.139)	2.391 (3.853)	0.771 (0.082)	1.909 (1.133)
CI-SPEA2	1.253 (1.895)	1.051 (0.457)	0.007 (0.015)	6.227 (17.90)	0.847 (0.033)	1.277 (1.970)
CI-NSGA-II-P	0.777 (0.318)	**1.031** (0.681)	**0.001** (0.001)	0.257 (0.246)	0.813 (0.039)	0.974 (0.475)
CI-SMS-EMOA-P	1.493 (0.766)	1.299 (0.682)	0.027 (0.086)	1.510 (1.265)	0.759 (0.061)	1.972 (0.736)
CI-SPEA2-P	0.619 (0.689)	1.141 (0.525)	0.847 (2.054)	2.364 (5.302)	**0.870** (0.027)	**0.971** (0.728)
R-NSGA-II	**0.084** (0.121)	1.503 (0.608)	**0.001** (0.001)	**0.137** (0.189)	0.767 (0.045)	1.876 (0.589)
WHYPE	13.2 (1.039)	–	–	3.340 (0.0)	0.475 (0.050)	2.477 (0.051)
ZDT6						
CI-NSGA-II	0.066 (0.019)	1.134 (0.389)	0.021 (0.075)	0.014 (0.025)	0.880 (0.002)	**0.074** (0.031)
CI-SMS-EMOA	0.077 (0.014)	**1.046** (0.360)	**0.008** (0.016)	**0.013** (0.012)	0.881 (0.002)	0.085 (0.016)
CI-SPEA2	0.050 (0.044)	1.174 (0.368)	0.058 (0.211)	0.068 (0.071)	0.885 (0.005)	0.166 (0.102)
CI-NSGA-II-P	0.070 (0.022)	1.251 (0.381)	0.053 (0.113)	0.027 (0.030)	0.878 (0.004)	0.143 (0.069)
CI-SMS-EMOA-P	0.077 (0.016)	1.139 (0.361)	0.025 (0.041)	0.036 (0.022)	0.878 (0.002)	0.214 (0.070)
CI-SPEA2-P	**0.024** (0.008)	1.356 (0.356)	0.010 (0.016)	0.033 (0.024)	0.881 (0.004)	0.122 (0.071)
R-NSGA-II	0.045 (0.007)	1.737 (0.158)	0.022 (0.016)	0.160 (0.016)	**0.890** (0.008)	0.459 (0.023)
WHYPE	0.107 (0.021)	1.324 (0.242)	0.074 (0.014)	0.089 (0.007)	0.886 (0.009)	0.266 (0.028)

Table 6.7: Mean values of the performance indicators for the DTLZ problems and standard the deviation in parenthesis.

DTLZ1	$D_{S \to P_F}$	Δ_P	Ψ	κ	R-metric	$C2_R$
CI-NSGA-II	0.229 (0.231)	0.318 (0.474)	**0.0** (0.0)	**0.011** (0.019)	0.916 (0.027)	0.272 (0.241)
CI-SMS-EMOA	0.147 (0.171)	0.114 (0.426)	**0.0** (0.0)	0.016 (0.02)	0.926 (0.019)	0.227 (0.224)
CI-SPEA2	0.578 (0.326)	**0.0** (0.0)	**0.0** (0.0)	0.093 (0.157)	0.878 (0.038)	0.612 (0.355)
CI-NSGA-II-P	0.261 (0.287)	**0.0** (0.0)	**0.0** (0.0)	0.019 (0.033)	0.912 (0.034)	0.328 (0.321)
CI-SMS-EMOA-P	**0.052** (0.119)	0.283 (0.57)	**0.0** (0.0)	0.014 (0.012)	0.937 (0.013)	**0.155** (0.155)
CI-SPEA2-P	1.496 (2.061)	0.615 (0.757)	1.697 (6.35)	38.666 (92.221)	0.894 (0.036)	1.599 (2.089)
R-NSGA-II	65.956 (79.425)	0.88 (0.846)	2706.534 (8319.215)	13.56 (10.844)	0.372 (0.064)	7.047 (1.094)
WHYPE	18.418 (2.473)	0.778 (0.071)	3464.915 (0.0)	61.917 (25.403)	**3.432** (0.772)	7.884 (0.844)
DTLZ2						
CI-NSGA-II	0.189 (0.14)	1.498 (0.27)	**0.0** (0.0)	0.015 (0.028)	0.846 (0.024)	0.092 (0.147)
CI-SMS-EMOA	0.363 (0.276)	1.469 (0.299)	**0.0** (0.0)	0.062 (0.064)	**0.874** (0.034)	0.302 (0.268)
CI-SPEA2	0.384 (0.306)	1.202 (0.341)	0.043 (0.086)	0.092 (0.072)	0.865 (0.034)	0.399 (0.268)
CI-NSGA-II-P	**0.145** (0.168)	1.495 (0.33)	**0.0** (0.0)	0.044 (0.037)	0.842 (0.027)	0.265 (0.169)
CI-SMS-EMOA-P	0.217 (0.321)	1.304 (0.188)	**0.0** (0.001)	0.082 (0.07)	0.848 (0.034)	0.381 (0.273)
CI-SPEA2-P	0.291 (0.301)	1.049 (0.354)	0.01 (0.038)	0.121 (0.065)	0.869 (0.032)	0.54 (0.204)
R-NSGA-II	0.259 (0.035)	1.459 (0.294)	**0.0** (0.0)	**0.006** (0.004)	0.857 (0.007)	**0.07** (0.054)
WHYPE	0.207 (0.009)	**0.538** (0.181)	0.349 (0.0)	0.113 (0.037)	0.84 (0.004)	0.382 (0.006)
DTLZ3						
CI-NSGA-II	**2.584** (1.354)	1.437 (0.27)	0.008 (0.029)	2.881 (2.21)	0.608 (0.178)	**3.195** (1.351)
CI-SMS-EMOA	4.23 (3.368)	1.364 (0.367)	5.891 (11.603)	36.487 (50.53)	0.547 (0.361)	4.251 (3.074)
CI-SPEA2	160.536 (86.576)	**0.678** (0.669)	2279.865 (8523.861)	20900.219 (37686.239)	0.562 (0.392)	160.771 (86.48)
CI-NSGA-II-P	3.313 (2.682)	1.649 (0.217)	0.018 (0.035)	5.79 (7.613)	**0.627** (0.255)	3.45 (2.697)
CI-SMS-EMOA-P	4.181 (2.467)	1.353 (0.235)	**0.005** (0.012)	6.165 (4.615)	0.562 (0.32)	4.41 (2.462)
CI-SPEA2-P	128.404 (48.913)	0.832 (0.703)	169689.67 (412604.401)	8021.122 (11506.598)	0.478 (0.448)	128.74 (48.878)
R-NSGA-II	5.199 (3.556)	0.989 (0.196)	0.015 (0.017)	**2.323** (2.397)	0.316 (0.151)	4.997 (3.817)
WHYPE	200.185 (44.507)	0.805 (0.0)	17218.998 (21595.623)	36765.282 (0.0)	0.0 (0.0)	234.892 (0.478)
DTLZ4						
CI-NSGA-II	0.016 (0.015)	0.818 (0.897)	**0.0** (0.0)	0.101 (0.095)	0.854 (0.022)	0.45 (0.368)
CI-SMS-EMOA	0.005 (0.005)	0.94 (0.885)	0.005 (0.008)	0.165 (0.053)	0.832 (0.026)	0.727 (0.154)
CI-SPEA2	**0.002** (0.003)	0.406 (0.673)	**0.0** (0.0)	0.202 (0.034)	0.848 (0.017)	0.824 (0.09)
CI-NSGA-II-P	0.025 (0.02)	0.616 (0.777)	**0.0** (0.0)	**0.09** (0.076)	0.831 (0.02)	**0.42** (0.303)
CI-SMS-EMOA-P	0.008 (0.01)	1.18 (0.753)	**0.0** (0.0)	0.148 (0.072)	0.838 (0.015)	0.65 (0.233)
CI-SPEA2-P	0.004 (0.005)	**0.317** (0.635)	**0.0** (0.0)	0.195 (0.033)	0.845 (0.017)	0.806 (0.087)
R-NSGA-II	0.016 (0.007)	0.714 (0.037)	**0.0** (0.0)	0.128 (0.006)	**0.864** (0.043)	0.615 (0.025)
WHYPE	0.069 (0.026)	0.417 (0.0)	0.384 (0.0)	0.154 (0.052)	0.844 (0.012)	0.506 (0.008)
DTLZ5						
CI-NSGA-II	0.663 (0.002)	1.292 (0.606)	**0.0** (0.0)	0.104 (0.001)	0.892 (0.001)	0.391 (0.029)
CI-SMS-EMOA	0.634 (0.052)	0.342 (0.567)	**0.0** (0.0)	0.157 (0.121)	**0.897** (0.009)	0.411 (0.388)
CI-SPEA2	0.877 (0.141)	**0.0** (0.0)	**0.0** (0.0)	0.251 (0.083)	0.832 (0.029)	0.609 (0.239)
CI-NSGA-II-P	0.663 (0.003)	1.44 (0.325)	**0.0** (0.0)	0.103 (0.004)	0.894 (0.002)	**0.385** (0.027)
CI-SMS-EMOA-P	0.666 (0.006)	1.151 (0.643)	**0.0** (0.0)	0.107 (0.004)	0.892 (0.001)	0.393 (0.025)
CI-SPEA2-P	0.658 (0.012)	0.182 (0.464)	**0.0** (0.001)	0.108 (0.009)	0.892 (0.002)	0.396 (0.026)
R-NSGA-II	**0.001** (0.0)	1.38 (0.294)	**0.0** (0.0)	**0.016** (0.0)	0.87 (0.005)	0.571 (0.014)
WHYPE	0.002 (0.0)	0.743 (0.091)	0.002 (0.0)	0.039 (0.0)	0.864 (0.0)	0.619 (0.009)
DTLZ6						
CI-NSGA-II	0.146 (0.016)	**0.0** (0.0)	**0.0** (0.0)	**0.011** (0.008)	0.863 (0.011)	**0.166** (0.187)
CI-SMS-EMOA	0.843 (0.271)	0.387 (0.642)	**0.0** (0.0)	0.212 (0.155)	0.793 (0.042)	0.989 (0.363)
CI-SPEA2	0.294 (0.168)	**0.0** (0.0)	**0.0** (0.0)	0.053 (0.048)	0.816 (0.037)	0.27 (0.285)
CI-NSGA-II-P	0.188 (0.025)	0.446 (0.748)	**0.0** (0.0)	0.072 (0.029)	0.836 (0.016)	0.356 (0.229)
CI-SMS-EMOA-P	0.857 (0.152)	**0.0** (0.0)	**0.0** (0.0)	0.249 (0.098)	0.755 (0.028)	0.939 (0.143)
CI-SPEA2-P	0.207 (0.0)	0.523 (0.867)	**0.0** (0.0)	0.092 (0.008)	0.834 (0.004)	0.453 (0.159)
R-NSGA-II	**0.003** (0.001)	1.767 (0.131)	**0.0** (0.0)	0.017 (0.002)	**0.871** (0.003)	0.34 (0.139)
WHYPE	0.098 (0.008)	0.522 (0.04)	0.025 (0.0)	0.055 (0.018)	0.849 (0.008)	0.403 (0.043)
DTLZ7						
CI-NSGA-II	0.026 (0.013)	1.421 (0.33)	**0.0** (0.001)	2.176 (0.583)	0.809 (0.023)	1.18 (0.326)
CI-SMS-EMOA	0.033 (0.013)	1.429 (0.412)	**0.0** (0.0)	3.108 (0.961)	0.84 (0.036)	1.682 (0.509)
CI-SPEA2	0.166 (0.23)	0.858 (0.644)	0.128 (0.201)	5.438 (1.206)	0.881 (0.022)	2.652 (0.337)
CI-NSGA-II-P	0.014 (0.001)	1.416 (0.367)	**0.0** (0.0)	**0.837** (0.471)	0.775 (0.02)	**0.096** (0.338)
CI-SMS-EMOA-P	**0.011** (0.002)	1.555 (0.316)	**0.0** (0.0)	1.966 (0.886)	0.824 (0.037)	0.947 (0.667)
CI-SPEA2-P	0.026 (0.01)	0.797 (0.269)	**0.0** (0.0)	1.396 (0.738)	0.795 (0.029)	0.567 (0.481)
R-NSGA-II	4.215 (0.866)	**0.734** (0.202)	0.677 (0.318)	22.649 (4.893)	0.906 (0.017)	7.564 (0.884)
WHYPE	0.012 (0.001)	0.776 (0.031)	0.025 (0.0)	7.525 (0.0)	**0.935** (0.009)	3.408 (0.027)

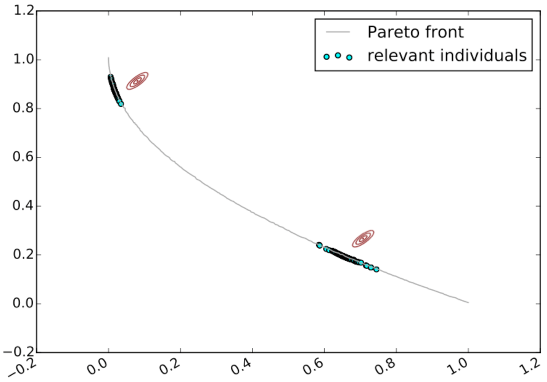
Table 6.8: Mean values of the performance indicators for the WFG problems and standard the deviation in parenthesis.

WFG1	$D_{S\to P_F}$	Δ_P	Ψ	κ	R-metric	$C2_R$
CI-NSGA-II	1.13 (0.026)	1.026 (0.232)	6e-05 (0.00012)	**0.237** (0.029)	**0.614** (0.005)	0.514 (0.059)
CI-SMS-EMOA	**1.123** (0.03)	1.269 (0.431)	0.00036 (0.00065)	0.287 (0.037)	0.609 (0.006)	0.589 (0.067)
CI-SPEA2	1.182 (0.022)	0.979 (0.507)	3e-05 (6e-05)	0.26 (0.028)	0.602 (0.004)	0.558 (0.07)
CI-NSGA-II-P	1.137 (0.029)	1.277 (0.307)	7e-05 (0.00016)	0.323 (0.059)	0.603 (0.005)	0.613 (0.086)
CI-SMS-EMOA-P	1.136 (0.027)	1.246 (0.208)	5e-05 (6e-05)	0.382 (0.024)	0.602 (0.006)	0.704 (0.031)
CI-SPEA2-P	1.18 (0.02)	1.127 (0.512)	**2e-05** (3e-05)	0.313 (0.042)	0.597 (0.004)	0.561 (0.073)
R-NSGA-II	1.502 (0.055)	1.218 (0.529)	0.001 (0.0)	6.48 (0.55)	0.609 (0.003)	**0.041** (0.058)
WHYPE	1.337 (0.014)	**0.446** (0.031)	0.03716 (0.02864)	0.224 (0.0)	0.595 (0.005)	0.747 (0.014)
WFG2						
CI-NSGA-II	0.116 (0.038)	1.279 (0.312)	0.00095 (0.00147)	0.205 (0.257)	0.825 (0.023)	0.677 (0.437)
CI-SMS-EMOA	0.136 (0.045)	1.034 (0.526)	0.0004 (0.00119)	0.293 (0.34)	0.839 (0.032)	0.979 (0.67)
CI-SPEA2	0.121 (0.025)	**0.605** (0.534)	0.00011 (0.00037)	0.373 (0.347)	0.839 (0.028)	1.228 (0.571)
CI-NSGA-II-P	0.139 (0.042)	1.09 (0.456)	0.00317 (0.00477)	0.234 (0.184)	0.813 (0.033)	**0.545** (0.445)
CI-SMS-EMOA-P	0.131 (0.047)	1.294 (0.264)	0.00041 (0.00127)	**0.129** (0.047)	0.824 (0.023)	0.704 (0.433)
CI-SPEA2-P	**0.107** (0.032)	1.244 (0.329)	**0.0001** (0.00019)	0.231 (0.287)	**0.842** (0.021)	0.947 (0.625)
R-NSGA-II	0.13 (0.03)	1.208 (0.402)	0.00085 (0.00225)	0.226 (0.444)	0.801 (0.028)	5.916 (0.407)
WHYPE	0.141 (0.012)	0.942 (0.046)	1.15454 (0.0)	1.572 (0.183)	0.811 (0.004)	2.299 (0.403)
WFG3						
CI-NSGA-II	0.426 (0.092)	0.723 (0.693)	**0.0** (2e-05)	**0.034** (0.025)	0.656 (0.013)	0.702 (0.18)
CI-SMS-EMOA	0.827 (0.08)	**0.138** (0.353)	**0.0** (0.0)	0.087 (0.111)	0.622 (0.018)	0.991 (0.391)
CI-SPEA2	**0.126** (0.069)	0.805 (0.765)	**0.0** (0.0)	0.107 (0.024)	**0.679** (0.01)	0.566 (0.249)
CI-NSGA-II-P	0.286 (0.064)	0.171 (0.45)	**0.0** (0.0)	0.282 (0.171)	0.625 (0.019)	0.484 (0.241)
CI-SMS-EMOA-P	0.45 (0.109)	0.214 (0.546)	**0.0** (0.0)	0.211 (0.114)	0.616 (0.012)	**0.481** (0.15)
CI-SPEA2-P	0.187 (0.053)	1.085 (0.625)	**0.0** (0.0)	0.557 (0.023)	0.609 (0.003)	0.819 (0.021)
R-NSGA-II	0.124 (0.039)	1.507 (0.2)	**0.0** (0.0)	0.136 (0.008)	0.654 (0.004)	4.645 (0.027)
WHYPE	1.337 (0.043)	0.47 (0.0)	0.03972 (0.0)	0.304 (0.01)	0.588 (0.01)	1.462 (0.033)
WFG4						
CI-NSGA-II	0.078 (0.048)	1.216 (0.297)	0.0041 (0.01009)	**0.078** (0.07)	0.555 (0.017)	2.225 (0.239)
CI-SMS-EMOA	**0.057** (0.033)	1.18 (0.241)	0.01278 (0.0229)	0.962 (0.615)	0.57 (0.047)	3.672 (0.402)
CI-SPEA2	0.117 (0.055)	**0.89** (0.506)	0.00042 (0.00089)	0.253 (0.11)	0.53 (0.021)	2.648 (0.357)
CI-NSGA-II-P	0.094 (0.058)	1.295 (0.277)	0.01484 (0.01795)	0.73 (0.107)	**0.682** (0.048)	0.627 (0.275)
CI-SMS-EMOA-P	0.131 (0.079)	1.186 (0.273)	0.00587 (0.01137)	1.093 (0.777)	0.62 (0.098)	1.407 (1.216)
CI-SPEA2-P	0.059 (0.032)	1.344 (0.25)	0.00488 (0.01331)	0.734 (0.117)	0.633 (0.071)	0.902 (0.558)
R-NSGA-II	0.207 (0.013)	1.515 (0.31)	**0.00023** (0.00035)	3.377 (0.235)	0.592 (0.038)	**0.071** (0.13)
WHYPE	0.25 (0.007)	0.904 (0.072)	7.14473 (0.0)	2.37 (0.145)	0.562 (0.011)	4.347 (0.012)
WFG5						
CI-NSGA-II	0.079 (0.006)	1.422 (0.401)	0.00052 (0.00102)	**0.1** (0.063)	0.55 (0.029)	2.368 (0.295)
CI-SMS-EMOA	0.08 (0.007)	1.375 (0.354)	0.00249 (0.00615)	0.244 (0.149)	0.531 (0.016)	2.818 (0.466)
CI-SPEA2	**0.075** (0.007)	1.332 (0.276)	0.00083 (0.00219)	0.101 (0.058)	0.55 (0.028)	2.158 (0.391)
CI-NSGA-II-P	0.086 (0.009)	1.556 (0.215)	0.0151 (0.03599)	0.739 (0.118)	0.661 (0.059)	0.63 (0.364)
CI-SMS-EMOA-P	0.079 (0.011)	1.308 (0.282)	0.00287 (0.00673)	0.712 (0.134)	0.633 (0.07)	0.848 (0.516)
CI-SPEA2-P	0.091 (0.005)	1.412 (0.35)	0.00012 (0.00042)	0.799 (0.105)	**0.679** (0.055)	**0.582** (0.305)
R-NSGA-II	0.103 (0.007)	**1.0** (0.148)	**2e-05** (3e-05)	2.635 (0.07)	0.606 (0.006)	4.953 (0.036)
WHYPE	0.114 (0.006)	1.021 (0.25)	5.01155 (1.83043)	2.511 (0.0)	0.585 (0.01)	4.447 (0.017)
WFG6						
CI-NSGA-II	0.076 (0.01)	1.249 (0.435)	0.00024 (0.00063)	0.089 (0.05)	0.545 (0.022)	2.359 (0.316)
CI-SMS-EMOA	0.08 (0.011)	1.304 (0.289)	0.00287 (0.00487)	0.202 (0.198)	0.549 (0.029)	2.656 (0.34)
CI-SPEA2	0.114 (0.028)	1.122 (0.493)	0.00069 (0.00232)	**0.082** (0.059)	0.542 (0.018)	2.329 (0.338)
CI-NSGA-II-P	0.068 (0.008)	1.324 (0.334)	0.00087 (0.00136)	0.641 (0.208)	**0.657** (0.054)	**0.602** (0.308)
CI-SMS-EMOA-P	**0.066** (0.009)	1.326 (0.212)	0.00088 (0.0015)	0.608 (0.166)	0.631 (0.066)	0.772 (0.627)
CI-SPEA2-P	0.087 (0.016)	0.924 (0.636)	**2e-05** (8e-05)	0.762 (0.128)	**0.657** (0.065)	0.617 (0.334)
R-NSGA-II	0.153 (0.005)	1.04 (0.204)	0.0001 (0.0)	2.648 (0.056)	0.593 (0.003)	4.967 (0.086)
WHYPE	0.437 (0.019)	**0.448** (0.0)	9.22655 (0.0)	2.853 (0.078)	0.505 (0.023)	4.452 (0.072)
WFG7						
CI-NSGA-II	**0.071** (0.088)	0.922 (0.606)	**0.0** (0.0)	**0.06** (0.053)	0.541 (0.016)	2.356 (0.325)
CI-SMS-EMOA	0.341 (0.168)	0.827 (0.705)	9e-05 (0.00023)	1.15 (0.572)	0.512 (0.037)	4.096 (0.459)
CI-SPEA2	0.088 (0.033)	0.692 (0.685)	**0.0** (0.0)	0.138 (0.072)	0.53 (0.016)	2.708 (0.216)
CI-NSGA-II-P	0.252 (0.245)	1.06 (0.499)	0.00015 (0.00044)	0.474 (0.296)	0.62 (0.07)	1.005 (0.65)
CI-SMS-EMOA-P	0.144 (0.191)	1.029 (0.552)	7e-05 (0.00016)	1.165 (1.91)	0.588 (0.11)	2.277 (1.528)
CI-SPEA2-P	0.39 (0.315)	0.887 (0.664)	2e-05 (5e-05)	0.754 (0.1)	**0.65** (0.076)	**0.803** (0.457)
R-NSGA-II	1.364 (0.069)	1.264 (0.32)	0.00019 (0.00021)	1.644 (1.664)	0.419 (0.1)	3.809 (1.188)
WHYPE	0.275 (0.007)	**0.408** (0.0)	9.49827 (0.0)	2.094 (0.0)	0.521 (0.012)	4.178 (0.142)
WFG8						
CI-NSGA-II	0.17 (0.022)	1.249 (0.226)	0.00283 (0.00452)	**0.083** (0.062)	0.526 (0.015)	2.517 (0.208)
CI-SMS-EMOA	0.114 (0.036)	1.288 (0.329)	0.00176 (0.00375)	0.587 (0.325)	0.529 (0.024)	3.535 (0.526)
CI-SPEA2	0.167 (0.012)	1.161 (0.335)	0.00046 (0.00099)	0.135 (0.075)	0.516 (0.013)	2.694 (0.246)
CI-NSGA-II-P	0.219 (0.03)	1.366 (0.334)	0.00037 (0.00083)	0.579 (0.261)	0.585 (0.06)	1.052 (0.738)
CI-SMS-EMOA-P	0.187 (0.038)	1.325 (0.28)	**9e-05** (0.00018)	0.518 (0.338)	0.537 (0.059)	1.996 (0.922)
CI-SPEA2-P	0.235 (0.035)	1.074 (0.44)	0.00019 (0.00038)	0.814 (0.158)	0.539 (0.06)	1.552 (0.809)
R-NSGA-II	**0.025** (0.015)	0.731 (0.445)	0.0001 (0.0)	0.833 (0.047)	**0.666** (0.007)	**0.034** (0.063)
WHYPE	0.408 (0.047)	**0.438** (0.036)	6.24101 (3.13704)	2.035 (0.0)	0.511 (0.014)	4.347 (0.049)
WFG9						
CI-NSGA-II	0.195 (0.064)	**0.288** (0.49)	0.00209 (0.00712)	**0.108** (0.115)	0.53 (0.021)	2.657 (0.329)
CI-SMS-EMOA	0.364 (0.24)	0.815 (0.668)	0.00063 (0.00237)	0.503 (0.373)	0.505 (0.057)	3.318 (0.744)
CI-SPEA2	0.233 (0.102)	0.552 (0.715)	**1e-05** (4e-05)	0.117 (0.054)	0.522 (0.023)	2.759 (0.271)
CI-NSGA-II-P	0.162 (0.051)	0.551 (0.757)	0.47146 (1.25232)	0.644 (0.28)	0.633 (0.064)	0.824 (0.828)
CI-SMS-EMOA-P	0.275 (0.177)	0.471 (0.532)	0.01537 (0.03918)	0.837 (0.791)	0.58 (0.092)	1.498 (1.403)
CI-SPEA2-P	**0.152** (0.037)	1.096 (0.602)	0.17525 (0.44329)	1.013 (0.604)	**0.688** (0.012)	0.473 (0.804)
R-NSGA-II	0.186 (0.075)	1.169 (0.641)	0.01504 (0.03656)	– (–)	0.679 (0.005)	**0.114** (0.217)
WHYPE	0.2 (0.013)	0.505 (0.024)	8.2346 (0.0)	2.109 (0.0)	0.544 (0.02)	4.249 (0.043)

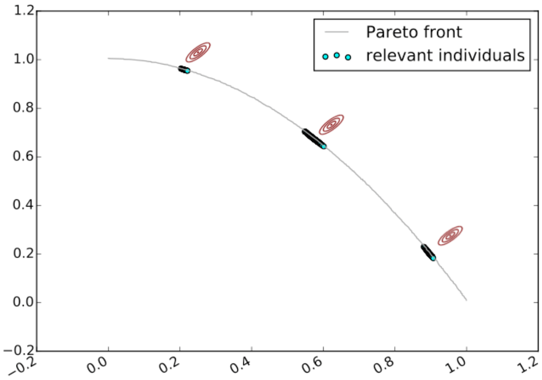
Table 6.9: Mean values of the performance indicators for the WFG problems with 5 objectives and standard the deviation in parenthesis.

WFG1 5D	Δ_P	Ψ	κ	R-metric	$C2_R$
CI-NSGA-II	0.968 (0.243)	0.12618 (0.02495)	0.126 (0.024)	**0.522** (0.007)	**0.644** (0.072)
CI-SMS-EMOA	0.955 (0.575)	0.13376 (0.11273)	0.205 (0.038)	0.5 (0.012)	0.813 (0.124)
CI-SPEA2	0.548 (0.504)	**0.09378** (0.09202)	0.141 (0.055)	0.503 (0.013)	0.699 (0.145)
CI-NSGA-II-P	0.684 (0.563)	0.10172 (0.08813)	0.242 (0.146)	0.461 (0.025)	1.307 (0.395)
CI-SMS-EMOA-P	0.564 (0.473)	0.14771 (0.24171)	0.648 (0.841)	0.458 (0.077)	1.567 (1.234)
CI-SPEA2-P	**0.546** (0.477)	0.31145 (0.2656)	1.038 (0.728)	0.387 (0.036)	2.673 (0.774)
R-NSGA-II	1.269 (0.327)	3.87818 (6.43124)	8.676 (4.317)	0.212 (0.22)	7.185 (1.884)
WHYPE	0.678 (0.03)	0.12259 (0.0)	**0.121** (0.0)	0.474 (0.018)	0.707 (0.016)
WFG4 5D					
CI-NSGA-II	1.308 (0.092)	0.56507 (0.1493)	**0.516** (0.121)	0.342 (0.037)	4.061 (0.475)
CI-SMS-EMOA	0.932 (0.24)	4.4675 (3.96587)	5.562 (3.296)	0.227 (0.089)	7.54 (1.109)
CI-SPEA2	**0.001** (0.0)	**0.002** (0.0)	1.233 (1.308)	0.291 (0.059)	4.778 (0.956)
CI-NSGA-II-P	1.154 (0.274)	0.50181 (0.41384)	0.612 (0.331)	**0.345** (0.031)	**3.981** (0.47)
CI-SMS-EMOA-P	0.559 (0.421)	4.68741 (5.97819)	12.629 (2.833)	0.068 (0.061)	9.347 (0.78)
CI-SPEA2-P	0.145 (0.289)	0.01 (0.0)	13.493 (2.437)	0.052 (0.058)	9.47 (0.798)
R-NSGA-II	1.149 (0.544)	8.64512 (7.09465)	13.469 (1.35)	0.005 (0.006)	10.477 (0.242)
WHYPE	0.612 (0.018)	4.0609 (0.87954)	3.948 (1.039)	0.309 (0.131)	5.88 (2.098)
WFG6 5D					
CI-NSGA-II	1.0 (0.673)	1.05365 (1.21659)	2.498 (1.754)	0.302 (0.1)	5.343 (1.554)
CI-SMS-EMOA	0.95 (0.489)	2.64464 (2.20507)	3.791 (1.559)	0.273 (0.153)	6.531 (1.098)
CI-SPEA2	**0.001** (0.0)	0.002 (0.0)	**0.516** (0.297)	**0.355** (0.063)	**3.772** (0.519)
CI-NSGA-II-P	0.897 (0.639)	6.08015 (5.98706)	11.462 (2.499)	0.095 (0.064)	8.951 (0.743)
CI-SMS-EMOA-P	0.723 (0.646)	**0.001** (0.0)	13.042 (2.205)	0.054 (0.044)	9.56 (0.347)
CI-SPEA2-P	0.567 (0.701)	0.66672 (2.49463)	11.037 (2.963)	0.099 (0.066)	8.911 (0.835)
R-NSGA-II	1.17 (0.249)	8.8697 (0.61994)	8.87 (0.62)	0.102 (0.012)	9.183 (0.171)
WHYPE	0.621 (0.0)	5.2361 (0.0)	5.323 (0.0)	0.158 (0.012)	7.52 (0.038)
WFG7 5D					
CI-NSGA-II	**0.0** (0.0)	**0.0** (0.0)	0.433 (0.278)	**0.286** (0.042)	**4.414** (0.694)
CI-SMS-EMOA	0.226 (0.451)	1.31364 (2.62729)	7.142 (1.673)	0.211 (0.15)	7.983 (0.804)
CI-SPEA2	0.161 (0.322)	**0.0** (0.0)	0.671 (0.307)	0.237 (0.015)	5.419 (0.421)
CI-NSGA-II-P	0.226 (0.452)	1.67824 (3.35648)	4.552 (2.994)	0.199 (0.062)	6.73 (1.798)
CI-SMS-EMOA-P	**0.0** (0.0)	**0.0** (0.0)	8.532 (3.462)	0.105 (0.084)	8.949 (1.009)
CI-SPEA2-P	**0.0** (0.0)	**0.0** (0.0)	**0.309** (0.187)	0.276 (0.023)	4.616 (0.367)
R-NSGA-II	1.018 (0.088)	**0.0** (0.0)	7.22 (2.123)	0.199 (0.129)	8.25 (0.581)
WHYPE	0.485 (0.0)	4.31094 (0.0)	4.272 (0.0)	0.261 (0.01)	6.725 (0.016)
WFG8 5D					
CI-NSGA-II	1.45 (0.214)	0.37453 (0.11174)	**0.372** (0.112)	0.33 (0.058)	4.068 (0.778)
CI-SMS-EMOA	1.146 (0.197)	2.35881 (2.28882)	2.322 (2.283)	0.237 (0.081)	5.701 (1.73)
CI-SPEA2	0.741 (0.664)	0.62431 (0.60756)	0.968 (0.521)	0.366 (0.029)	4.534 (0.731)
CI-NSGA-II-P	1.088 (0.577)	5.69718 (4.65372)	10.13 (1.047)	0.119 (0.036)	8.841 (0.518)
CI-SMS-EMOA-P	**0.649** (0.371)	5.52709 (5.44231)	10.36 (3.461)	0.089 (0.056)	8.88 (1.015)
CI-SPEA2-P	0.986 (0.646)	**0.0** (0.0)	9.94 (4.728)	0.129 (0.126)	8.162 (2.361)
R-NSGA-II	0.974 (0.036)	10.42521 (0.03873)	0.636 (0.015)	**0.601** (0.033)	**0.23** (0.264)
WHYPE	0.678 (0.02)	5.22633 (0.0)	5.241 (0.0)	0.194 (0.018)	7.498 (0.126)
WFG9 5D					
CI-NSGA-II	0.48 (0.588)	**0.0** (0.0)	**0.415** (0.219)	0.307 (0.038)	4.532 (0.685)
CI-SMS-EMOA	0.276 (0.553)	1.64257 (3.28515)	5.003 (2.734)	0.143 (0.076)	7.938 (1.254)
CI-SPEA2	**0.0** (0.0)	**0.0** (0.0)	0.829 (0.613)	0.251 (0.013)	5.305 (0.874)
CI-NSGA-II-P	**0.0** (0.0)	**0.0** (0.0)	9.063 (5.438)	0.107 (0.111)	8.279 (2.007)
CI-SMS-EMOA-P	0.153 (0.305)	2.35033 (4.70066)	11.783 (2.704)	0.055 (0.044)	9.375 (0.57)
CI-SPEA2-P	0.556 (0.685)	3.08663 (6.17327)	14.656 (1.524)	0.016 (0.028)	9.863 (0.279)
R-NSGA-II	0.633 (0.716)	9.4586 (0.10709)	9.448 (0.103)	**0.595** (0.05)	**0.099** (0.347)
WHYPE	0.52 (0.0)	6.40395 (0.0)	6.416 (0.0)	0.102 (0.008)	8.311 (0.188)

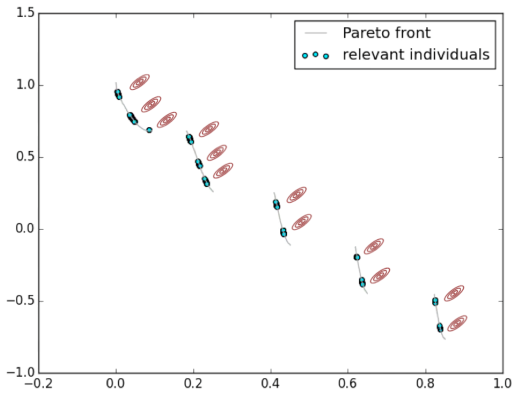
As an example, figure 6.10 shows the relevant regions in Pareto front found by the best run of CI-NSGA-II in terms of $D_{S \to P_F}$, κ and Ψ.



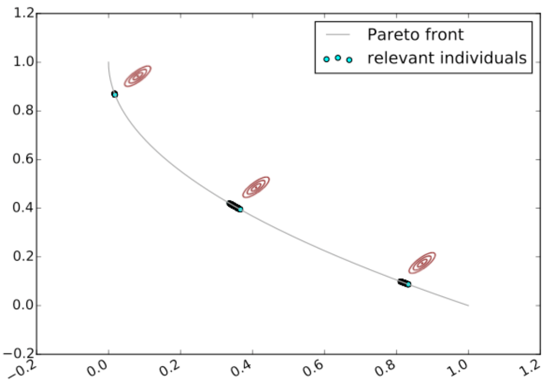
(a) ZDT1



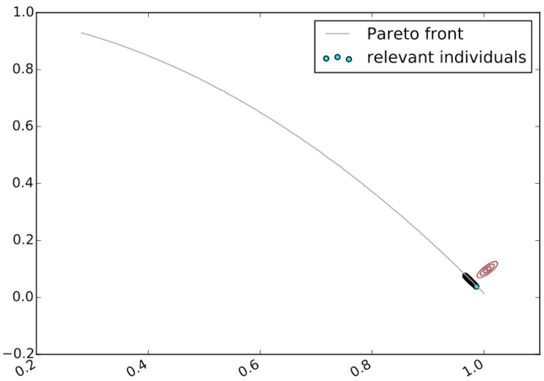
(b) ZDT2



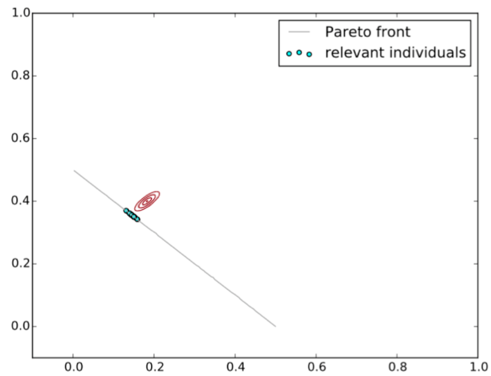
(c) ZDT3



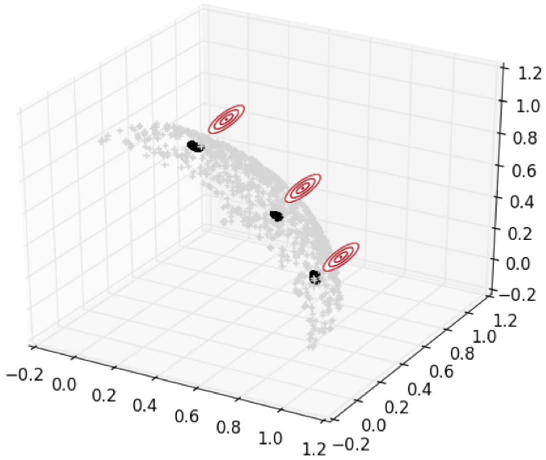
(d) ZDT4



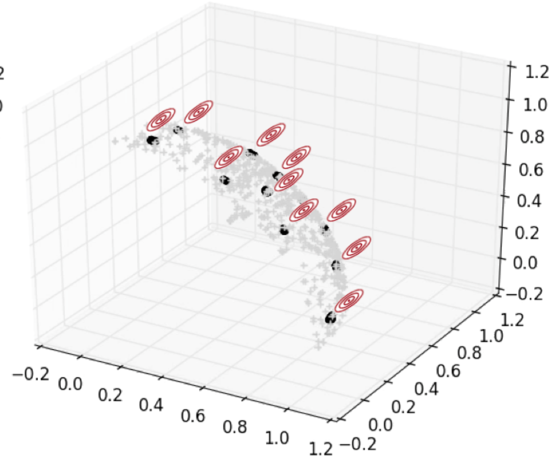
(e) ZDT6



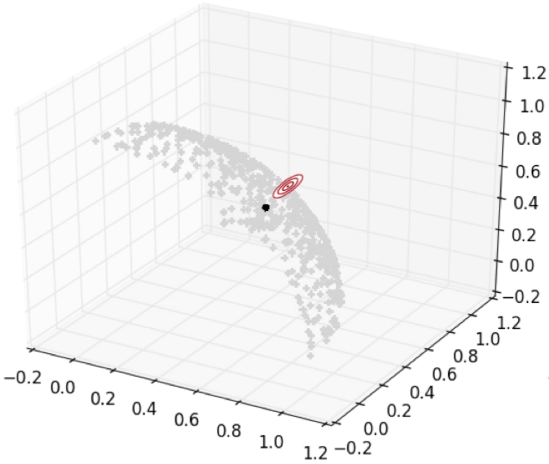
(f) DTLZ1



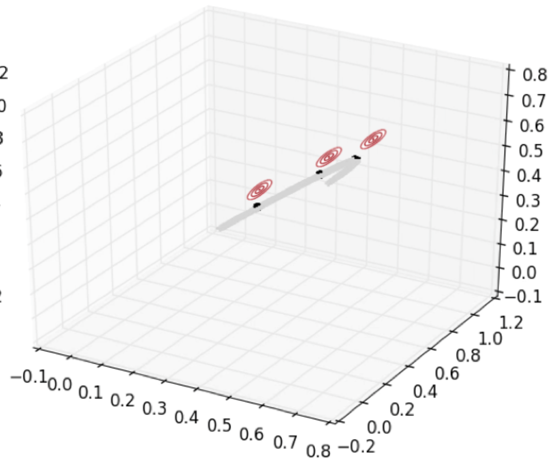
(g) DTLZ2



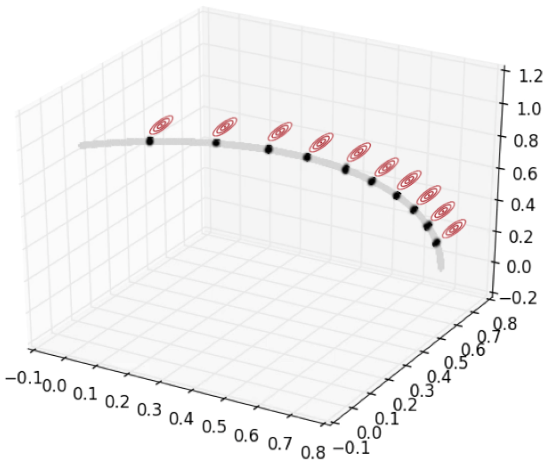
(h) DTLZ3



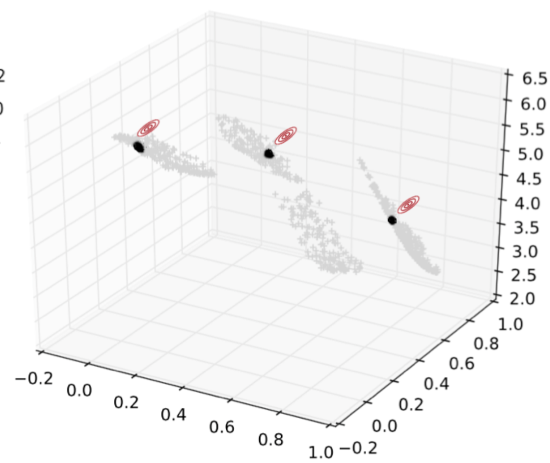
(i) DTLZ4



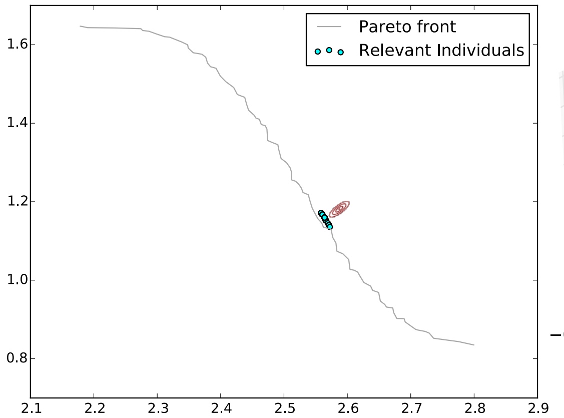
(j) DTLZ5



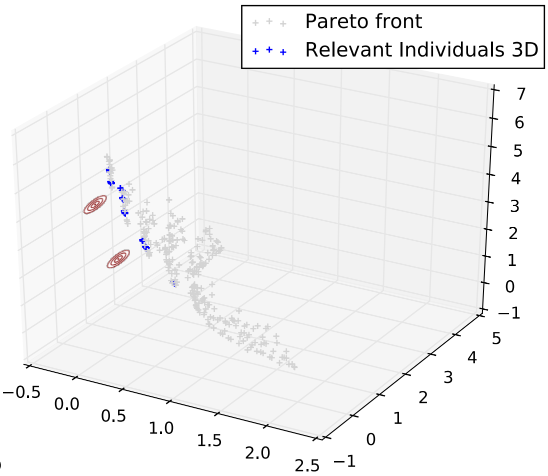
(k) DTLZ6



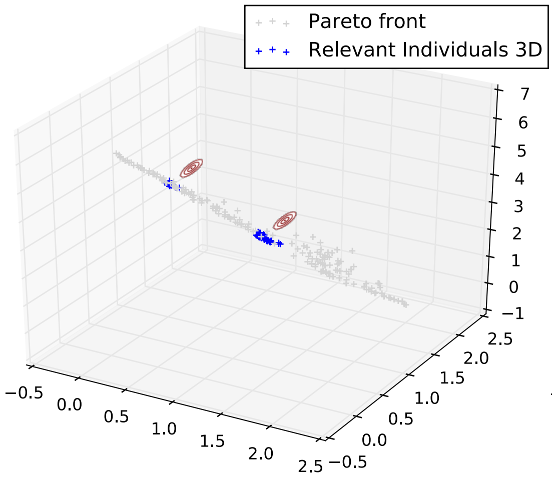
(l) DTLZ7



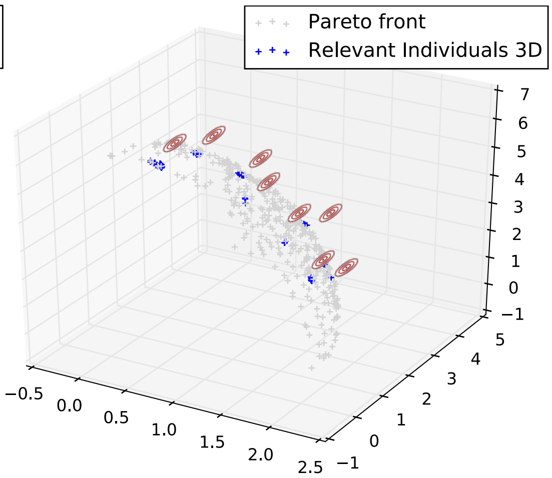
(m) WFG1



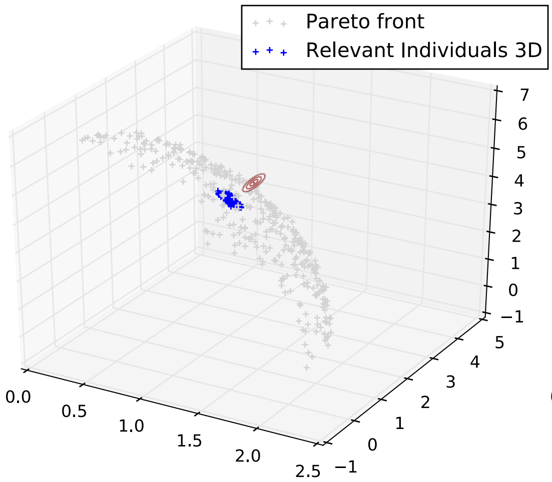
(n) WFG2



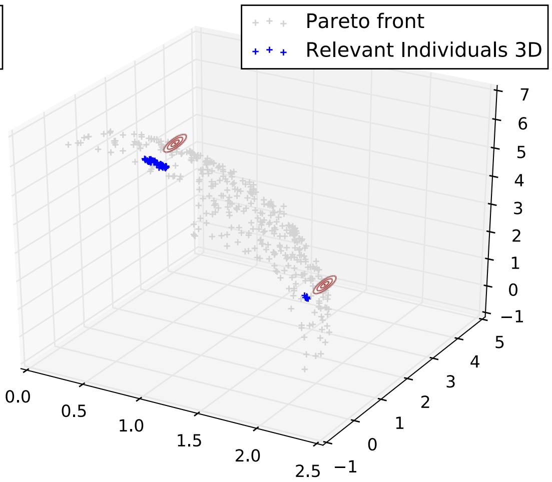
(o) WFG3



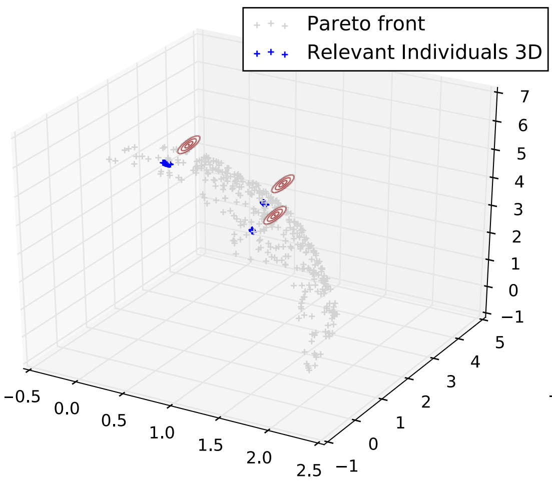
(p) WFG4



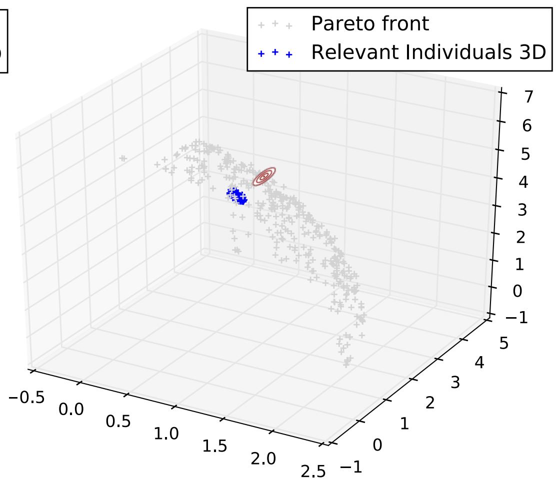
(q) WFG5



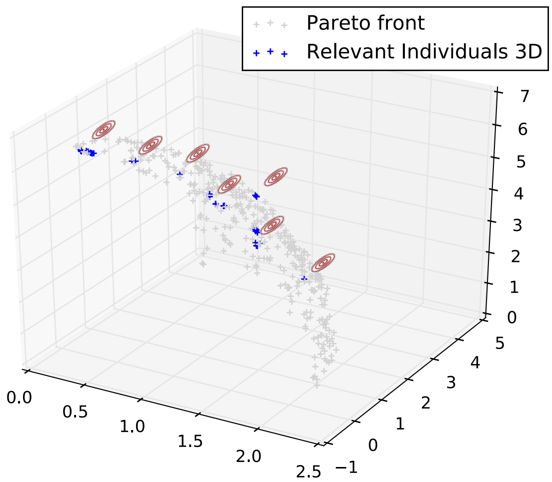
(r) WFG6



(s) WFG7



(t) WFG8



(u) WFG9

Figure 6.10: Relevant regions in Pareto front found by CI-NSGA-II for all the test problems.

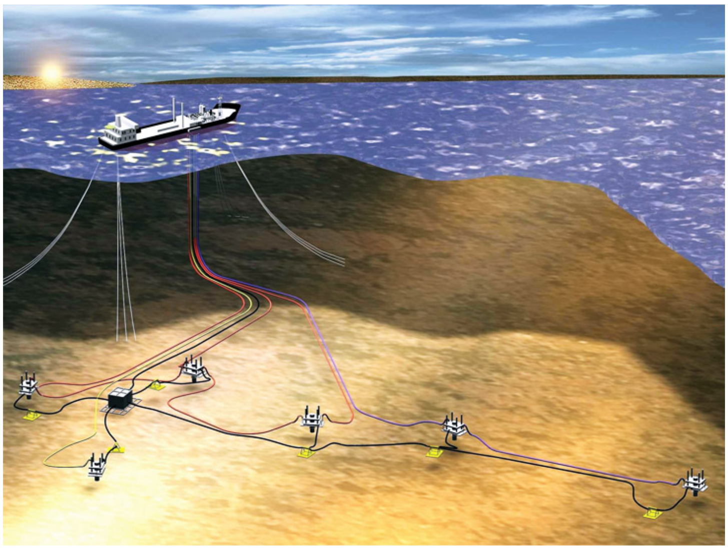


Figure 6.11: Conceptual studies for Oil & Gas field developments covering subsea and offshore systems. Figure taken from [4].

6.2 Collective Intelligence In a Facility Location Case

The location of operational facilities is a strategic goal for many companies. Usually, they have to efficiently manage demands in a network environment. The petroleum and mining industry is one of the domain contexts where these circumstances prevail. Those companies must extract oil or other geological materials from resource areas and allocate facilities (offshore platforms or warehouses) in such a way that optimizes its operational costs and production capacity. More generally, they transform the management of resources into a multi-objective decision making problem with many stakeholders looking for efficient approaches. Figure 6.11 illustrates a conceptual study for Oil & Gas subsea field with a specialized vessels and pipeline installation [3].

6.2.1 Problem Formalization

The problem —to put it in simple terms— has to find a good solution for positioning the processing units according the resource area. It is formally represented as:

$$\min \sum_{i=1}^{N} \sum_{j=1}^{M} \sigma_{ij} d_{ij} + \sum_{j=1}^{M} c_j \mu \tag{6.2}$$

$$\max \sum_{i=1}^{N} \sum_{j=1}^{M} \sigma_{ij} v_j \tag{6.3}$$

Let μ be the cost of one processing unit, v the productive capacity of one processing unit linked to one resource area, M a set of available positions to processing units, N a set of available positions to resource area and D a distance matrix $(d_{ef})_{n_x m}$, where $n \in N$ and $m \in M$. The decision variables are the processing unit c_j $(j \in M)$ that assumes 1 if it is placed at position j or 0 otherwise and σ_{ij} that assumes 1 if there is a link between the resource area g at position $i \in N$ and the processing unit at position $j \in M$.

$$c_j = \begin{cases} 1 & \text{if the processing unit is placed} \\ 0 & \text{otherwise} \end{cases} \tag{6.4}$$

$$\sigma_{ij} = \begin{cases} 1 & \text{if there is a link between } g_i \text{ and } c_j \\ 0 & \text{otherwise} \end{cases} \tag{6.5}$$

Regarding the constraints of the problem, let P the available positions of the resource area inside the extraction area, S the actual positions of all resource areas, R the actual positions of all processing units and α a distance function. The constraint 6.6 defines that the quantity of processing units cannot be greater than the number of resource areas; 6.7 limits the link of one resource area to only one processing unit; 6.8 ensures that all the processing units are allocated at least one time and no more than z times; 6.9 restricts a resource area g to a unique position inside one specific extraction area in the map.

In 6.10 the minimum distance (k_1) allowed between the processing unit and the resource area is defined, 6.11 sets the minimum distance (k_2) between two different processing units.

$$1 \leq \sum_{j=1}^{M} c_j \leq \sum_{i=1}^{N} g_i \tag{6.6}$$

$$\sum_{j=1}^{M} \sigma_{ij} = 1, \qquad \forall i \in S \tag{6.7}$$

$$1 \leq \sum_{i=1}^{N} \sigma_{ij} \leq z, \qquad \forall j \in R, z \geq 1 \tag{6.8}$$

$$\sum_{i=1}^{P} g_i = 1 \tag{6.9}$$

$$\alpha\left(g_i, c_j\right) \geq k_1, \qquad \forall i \in S, \forall j \in R \tag{6.10}$$

$$\alpha\left(c_i, c_j\right) \geq k_2, \qquad \forall i, j \in R, i \neq j \tag{6.11}$$

$$\tag{6.12}$$

6.2.2 Chromosome Encoding

The processing unit is computationally represented as a tuple $c_i =< x, y, t >$; where $c_i \in C = \{c_1, ..., c_k\}$, t is the type of the unit, x and y are the Cartesian coordinates of the position. The resource area is represented by the tuple $a_j =< x, y, l >$; where $a_j \in A = \{a_1, ..., a_q\}$, $l \in \{1, ..., k\}$ is the index of the processing unit that links a resource area a_j to the unit c_j. For example, if the second resource area a_2 is linked to the first processing unit c_1, then $l = 1$.

Each individual candidate route is also encoded as a set of genes. The intermediate routes $r^{a_j c_i}$ between the resource area a_j and the processing units c_i are appended to the end of the chromosome. The routes store only the vertices $< x, y >$ of the line segments that connect the objects. Thus, the chromosome encoding (Figure 6.12) is the aggregation of these tuples regulated by q resource areas and k processing units.

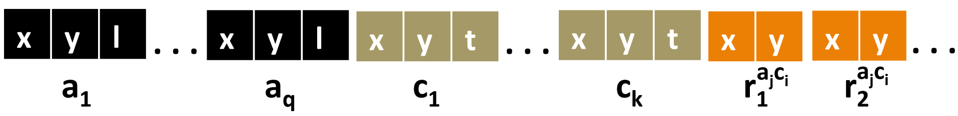


Figure 6.12: Chromosome encoding.

6.2.3 Gamification

Different constraints from real life and several new interdependencies among the variables will increase the search complexity of this facility location problem. Progressive articulation of preferences and collective intelligence can implement a dynamism not managed by *a priori* methods and enhance its efficiency. Therefore, the situation described is a candidate for this experiment due to some reasons: a) as a real-world case example, the objectives and decision variables are meaningful to the group; b) the problem interacts with crowd's cognition and requires a 3D spatial reasoning to avoid natural obstacles in the scenario; c) incentive engines and gamification can be used to retain the users' interest on the interaction during the optimization; d) the users' feedback can be parallelized in synchrony with the evolution of individuals in an evolutionary algorithm.

Gamification [69] is the integration of game design elements and game engines in non-game contexts. This is usually intended to increase engagement of players, create gameful and playful user experiences, motivate them and set clear objectives to guide a cooperative or competitive behaviour. Games have been used to support science by leveraging human problem solving ability to work on computationally difficult scientific problems.

Humans have a more range of exploration methods than computers [20, 21]. Some players prefer to focus on winning and competition, they try to achieve goals quickly. Others users focus on a more collaborative behaviour, they work together by combining strengths to accomplish a set of goals.

In this context, the facility location problem was designed as a game where every player competes among themselves to obtain points and recognition of success. This is usually intended to increase engagement of players, create gameful and playful user experiences, motivate them and set clear objectives to guide a cooperative or competitive behaviour. The game elements were transformed to preserve the sensitive details of a major petroleum extraction and processing conglomerate of Brazil. Trucks represent the resource areas and the warehouses or barracks symbolize the processing units with two different types. The graphical interface of the game allows the user to manually define and position any given object.

Software architecture is an abstract and modular description of a system. It refers to the software elements, structures, properties and interfaces necessary to design a system [152]. The gamified facility location problem is divided into 5 different layers: database,

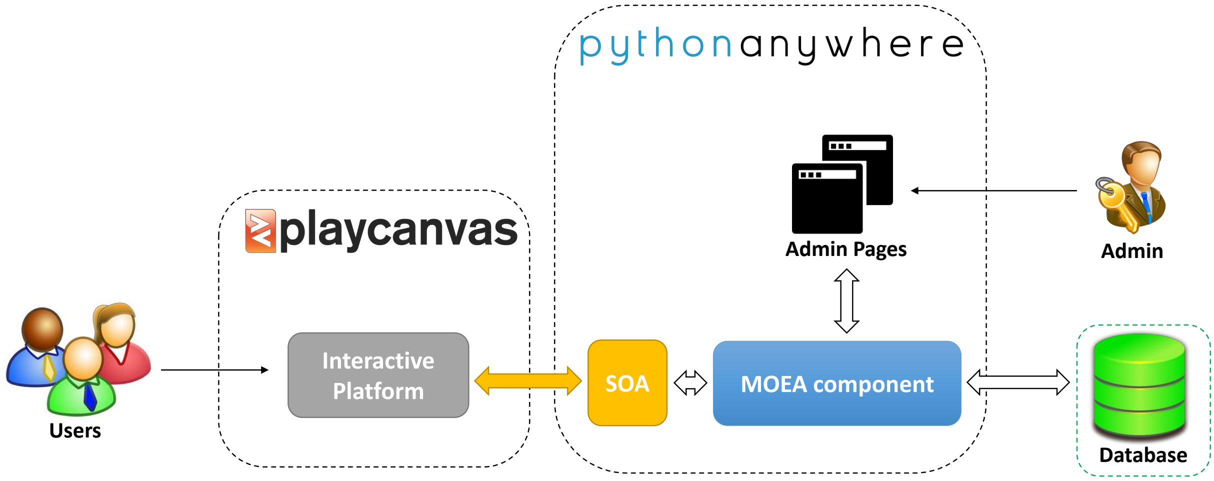


Figure 6.13: High level description of the game architecture.

core components of MOEA, communication components, user interactive platform and the web pages to control the game progress. A high level description of the software architecture is demonstrated in Figure 6.13. The dashed green box denotes the database layer, the MOEA's component is shown in blue, the communication layer is drawn in yellow, the interactive platform is represented in gray and the administration web pages layer in black.

The database module uses the SQLite relational database [86]. SQLite is an open source engine and work with PostgreSQL as a reference platform. It is widely applied in the data management of several applications, such as: mobile devices, websites, CAD/CAM products, industrial control, etc [9]. The MOEA's components were developed on a python evolutionary framework (DEAP [79]) that controls the genetic operators and the interactive behavior of the new collective intelligence algorithms.

Following the Service-Oriented Architecture (SOA) model [8,108], the communication layer provides a collection of web services responsible for transferring messages from the user platform to the core components of MOEA. This set of functionalities is available on the internet through RESTful web services [71]. The user interactive platform was implemented in an open source 3D WebGL game engine called Playcanvas. This platform supports JavaScript programming language and real-time editing of the code. All participants of the collective intelligence will access this graphical interface[1] to interact with the game. Finally, the administrative web pages were developed using Django framework (Figure 6.14), which follows a model-view-template (MVT) architectural pattern. These web pages manage the interruption of the evolution process and control when to start or

[1]http://playcanv.as/p/1ARj738G

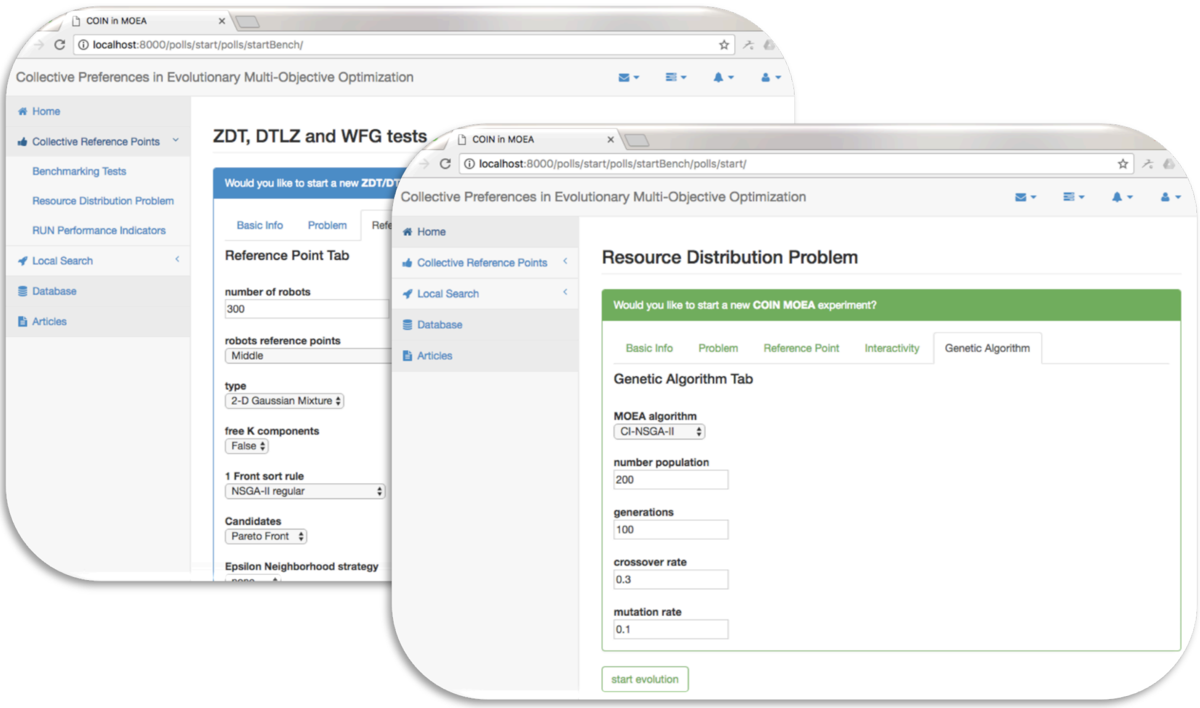


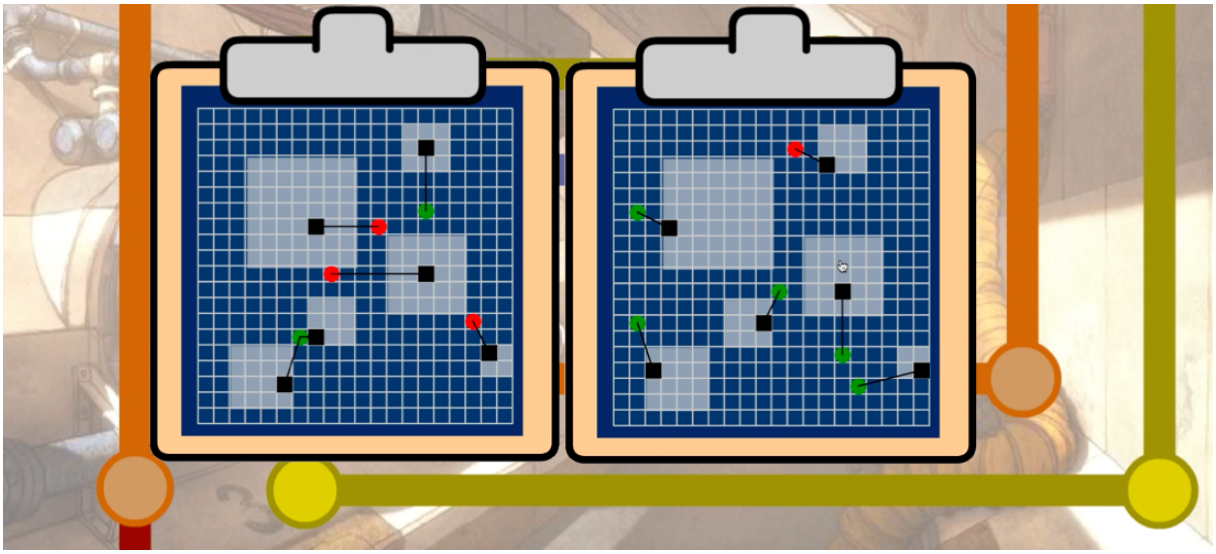
Figure 6.14: Administrative web pages to manage the interruption of the evolution process and control when to start or finish a problem optimization.

finish a problem optimization.

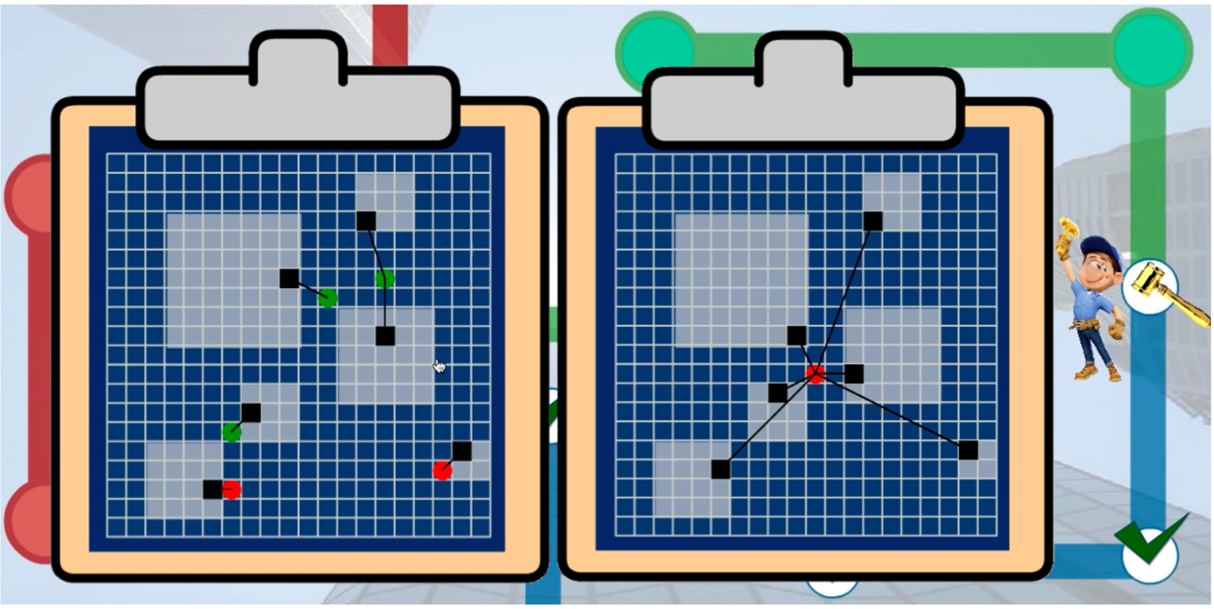
The code was written primarily in Python language, except for the interactive platform on Playcanvas that has been developed in JavaScript. The complete source code is available at GitHub repository: https://github.com/quatrosem/Bangkok. Both the SQLite database and the admin pages are hosted on PythonAnywhere (www.pythonanywhere.com), an internet web server. Appendix E brings more details of the tools used as part of the system architecture.

6.2.3.1 Game Modes

There are two options in the game: a) pairwise comparison, which implements the selection operator; b) free design mode, which implements the variation operator. In the pairwise comparison mode, the players must vote on the best candidate (individual from population) between two or more facility location scenarios. As votes on the scenarios happen, the Gaussian Mixture model calculates the collective reference point to restrict the search to relevant areas in Pareto front. The players who have chosen the individuals near the collective reference point receive a higher score. They compete at every evolution interval for choices around the collective mean. Figure 6.15 exhibits different phases of the game and the screen for pairwise comparisons of individuals. Note that the early generations present less efficient individuals in terms of cost, because the distances between the processing units and the resource areas are better optimized in the final generations.



(a) Two individuals from the early generations.



(b) Two individuals from the final generations

Figure 6.15: Gamification features and pairwise comparisons.

In the free design mode, some individuals from the population are distributed to the players who have to fix and change their position arrangement. The dynamic game scenario allows the creation of objects like trucks or warehouses, changing their arrangements and rebuilding their connections with straight lines or zigzag lines. This game mode uses the collaboration of people to apply rational improvements in the quality of EA population. Figure 6.16 shows the dynamic board scene and its internal representation inside the algorithm.

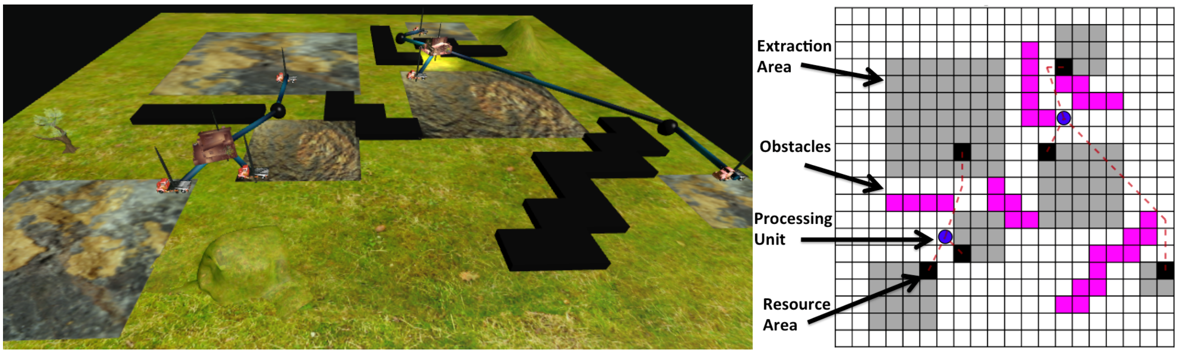


Figure 6.16: Game and Computational representation of the facility location problem.

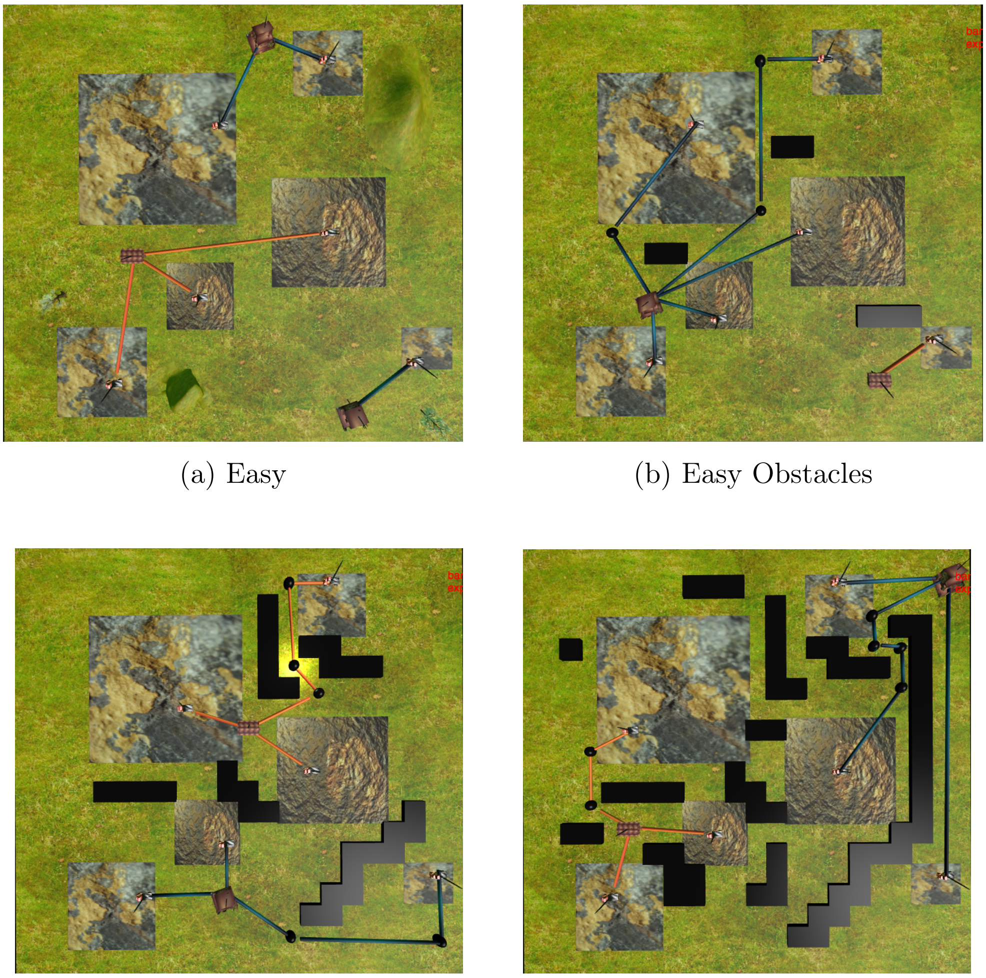
Trucks represent the resource areas and the warehouses or barracks symbolize the processing units with two different types. All the participants can control their interactions with the evolutionary algorithm through the buttons: *join game*; *get scene*; *send scene*;*evolve*; *redraw*; *last winner*.

- *join game*: Connect the user's game instance to the running MOEA. This button introduces the user in the game.
- *get scene*: Get one individual scenario from the current Pareto front and display on the user's screen.
- *send scene*: Send the modified individual to the current MOEA population.
- *evolve*: Run another block of generations on the server.
- *redraw*: Redraw the original individual in the screen (discard any local modification).
- *last winner*: Get the best individual from the last ended game.

6.2.3.2 Scenarios of Free Design Mode

There are four different scenarios available in the game: *easy*, *easy obstacles*, *medium obstacles* and *hard obstacles*. The first one allows only straight lines to connect objects and ignores any sort of obstacles. The lines are defined by segments connecting points, representing for example pre-existent pipelines or flowlines. From second to the fourth scenario, the zigzag lines have to be used to avoid the obstacles. The level of difficulty increases according to the number of obstacles in the game scene. The *medium* and *hard* scenarios are complex and simulate aspects of the real world. The placement of facilities has to consider factors like competition for shared resources and obstructed paths.

The barrier regions simulate real-world constraints where travelling or locating new facilities are not allowed: mountains, lakes, forests, hazardous and residential areas. In 2D space, this infeasible region is defined as: $\mathcal{B} = \cup_{h=1}^{H} B_h \subset \mathbb{R}^2$, where H is the set of barriers. On the other hand, the feasible region is: $\mathcal{F} = \mathbb{R}^2 \setminus int(\mathcal{B})$. So, all processing units $c_1, \ldots, c_k$, resource areas $a_1, \ldots, a_q$ and routes are placed in $\mathcal{F}$. Figure 6.17 shows the four different scenarios available in the game.



(a) Easy

(b) Easy Obstacles

(c) Medium Obstacles

(d) Hard Obstacles

Figure 6.17: The four different scenarios available in the game.

In addition, the four scenarios were developed to work in discrete or continuous space. The objective functions presented in 6.2.1 can also take discrete or continuous values. Discrete data has a finite set of possible values. Continuous data are not restricted to separate values. It can occupy any value over a continuous range, which makes the proposed optimization problem even more difficult.

6.2.4 Results with Collective Intelligence

In [87], the authors proposed a 5-position schema to describe all location models. The classification has the following format: *Pos1 / Pos2 / Pos3 / Pos4 / Pos5*. The *Pos1* contains information about the number and type of the new facilities. *Pos2* gives information regarding the decision space: discrete, network or continuous problems. *Pos3* expresses particularities of the specific location problem. *Pos4* describes the relation of the facilities. Finally, *Pos5* contains information regarding the objective functions.

Based on the 5-position schema, the facility location problem used in this experiment (Subsection 6.2.1) is then classified as: $p,\ A\ /\ \mathbb{R}^2\ /\ \mathcal{F}, \mathcal{B}, w_m = 1, mc/\ l_2\ /\ Q{-}(\sum, min)_{par}$. The *Pos1* and *Pos2* express that the problem must locate many paths of several line segments (p) and multiple areas (A) in the 2-dimensional space ($\mathbb{R}^2$). *Pos3* indicates the existence of a feasible region ($\mathcal{F}$) with barriers ($\mathcal{B}$), where neither placement of facilities nor paths are allowed. The information in this position also means that the problem is unweighted ($w_m = 1$) and there is mutual communication between the new facilities (mc). *Pos4* determines the distance function used: Euclidean distance (l_2). The last position, *Pos5*, stands for a minimization of a multi-criteria problem ($Q - (\sum, min)_{par}$).

The coming subsections present the results found for the discrete and continuous facility location problem.

6.2.4.1 Experimental Settings and Definitions

The experiments were applied in two different computer labs: a Brazilian professional education center with more than 25 students' attendance; a private company training room with engineers and IT analysts. This way, there are two different participants' profiles. The group of engineer specialists and a second one of IT professionals and students. Besides, one offshore engineer was selected as the main DM to evaluate the final scenarios obtained by the COIN algorithms.

There were performed 12 experiments in each lab. The results in the following subsections show the mean values for each distinct game scenario (easy – easy obstacles – medium obstacles – hard obstacles). The population size has been set to 200 for all the problems (discrete or continuous). The crossover and mutation probability are 0.9 and 0.1, respectively.

According to the test results in Section 6.1.5, CI-NSGA-II outperformed the algorithms CI-SMS-EMOA and CI-SPEA2. Along with the the CI-NSGA-II-P, they cover

77% of the 3 first positions in the benchmark problems. As the R-NSGA overcame the WHYPE, the experiment with a true collectivity compares the new CI-NSGA-II with the original NSGA-II and R-NSGA-II. The R-NSGA-II uses the ideal point z_i^* as a fixed reference point. The main goal is to analyse the performance of CI-NSGA-II facing distinct environments and to identify when the collective intelligence has a positive influence on the results.

The CI-NSGA-II has three variations regarding the sort of input from the collective intelligence. The *CI-NSGA-II Vote* variation interrupts the evolution process and asks the players to choose the best candidate between two scenarios. This variation uses the COIN-based Selection operator to vote on the most fitting individuals and discover the collective preferences. In the *CI-NSGA-II Fix* variation, the participants can interactively update and redesign all the elements in the game scene. They use their cognition and reasoning skills to improve the configuration of intermediate MOEAs solutions. The third variation, *CI-NSGA-II Ignition*, accepts the collective contributions only in the beginning of the evolution (first generation), then it runs to the end without interference.

All the three variations aggregate a rational input to the current population. The time interval for human collaboration is 30 seconds for pairwise comparisons (*CI-NSGA-II Vote*) and 60 seconds for game scene update (*CI-NSGA-II Fix* and *CI-NSGA-II Ignition*).

In [118], the authors investigate the impact of the number of interactions on the convergence of EMOAs. Based on their findings, the number of human–computer interactions adopted in these experiments are 6 for the CI-NSGA-II. Regardless, the original NSGA-II and the R-NSGA runs independently with no interaction.

The users' contributions maintain a pattern with a constant number of generations between each interaction. This pattern calls for interactions after 14%, 28%, 42%, 56%, 70% and 84% of the generations (fixed number) have elapsed. Different patterns could be used to collect the preferences, such as the front-loaded where more interactions occur during the early stages of the algorithm or the center-loaded pattern under which more interactions occur during the middle stages of the algorithm. However, the distinct interaction patterns did not produce statistically significant differences in the results [118].

This work introduces the COIN operator with the purpose of iteratively refine the search parameters with rational collaborations and improve the overall quality of evolutionary population. It is expected that the suggested approach decreases the number of function evaluations, accelerates the convergence and achieves relevant regions of Pareto front at a lower computational cost.

Thus, the methods used to evaluate the performance of the CI-NSGA-II in the discrete problem are: the number of function evaluations, fixed distance and fixed time. The evaluation function is the most time consuming element and it is normally taken for comparison. In the fixed distance, a minimum distance between the current approximation set S and the Pareto-optimal front is measured by the front coverage indicator, $D_{S \to P_F}$. A proximity of $D_{S \to P_F} = 20$ is the criteria to stop the evolution and compare the algorithms. In the fixed time, the algorithms run in a time interval previously defined.

Among the participants, there is an offshore engineer elected as the main DM. This role evaluates the final scenarios obtained by the COIN algorithms. For a different analysis of the results, the offshore engineer provided an *a priori* reference point to be used as a comparison. This way, the Reference Set Distance ($C2_R$) performance indicator can be applied to indicate the distance between the *a priori* reference point and the outcomes of COIN algorithms.

To summarise, in the discrete problem, the experiment with a true collectivity compares the algorithms:

- NSGA-II
- R-NSGA-II
- CI-NSGA-II Vote
- CI-NSGA-II Fix
- CI-NSGA-II Ignition

The performance indicators are:

- number of function evaluations
- fixed distance ($D_{S \to P_F}$)
- fixed time

In the continuous problem, the experiment compares the algorithms:

- NSGA-II
- CI-NSGA-II Fix

- R-NSGA-II

The performance indicators are:

- number of function evaluations
- fixed distance ($D_{S \to P_F}$)
- fixed time
- $C2_R$ (Reference Set)
- R-metric
- Filatovas Spread (Δ_P)
- Referential Cluster (κ)
- Convex Hull (Ψ)

6.2.4.2 Discrete Problem

The bi-objective problem is solved using the COIN MOEAs. Tables 6.10 and 6.11 present the results for fixed time and distance evaluation, respectively. Based on the results, the original R-NSGA-II won in the *Easy* and *Easy Obstacles* scenarios. The problem without obstacles is so simple that the algorithm took only two seconds to reach a convergence of $D_{S \to P_F} = 20$. In the case of fixed time: 5", there was not sufficient time to involve a collective participation of users, so the CI-NSGA-II variations were not applicable (NA).

The *CI-NSGA-II Fix* and *Igni* had a better performance in the *Medium Obstacles* scenario. Although the R-NSGA-II required less number of function evaluations to reach the convergence $D_{S \to P_F} = 20$, the *CI-NSGA-II Igni* results were close and obtained the lowest Referential Cluster Variance indicator κ, which means the points are clustered closely around the collective reference point. *CI-NSGA-II Fix* dominated the values of the fixed time evaluation.

The great difference in the result appears in the *Hard Obstacles* scenario. The *CI-NSGA-II Fix* succeeded in all three indicators with the support of the collective intelligence. Considering the fixed distance evaluation, the algorithm required 5 times less function evaluation and performed 3.2 times faster than NSGA-II. In terms of the fixed time evaluation, it managed to find a better convergence 3.5 times better than R-NSGA-II.

Table 6.10: Results of fixed time evaluation in discrete problem.

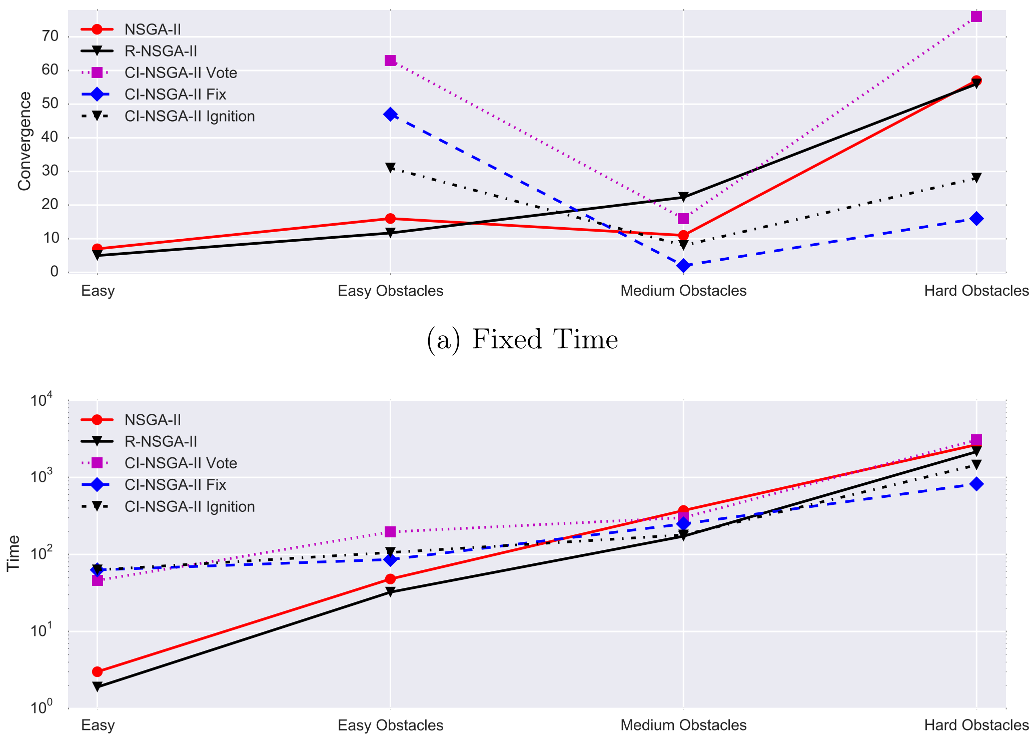
Algorithms	**Easy**				**Easy Obstacles**			
	Time:5"				Time:70"			
	$D_{S\to P_F}$	$Num.Eval.$	κ	Ψ	$D_{S\to P_F}$	$Num.Eval.$	κ	Ψ
NSGA-II	7,1	15.440	16,5 e05	**2590,0**	16,0	91.680	16,2 e05	2974,8
R-NSGA-II	**5,54**	**8.000**	**14,5 e05**	2617,0	**11,7**	31.600	13,4 e05	1115,6
CI-NSGA-II Vote	NA	NA	NA	NA	63,6	19.240	22,4 e05	6632,5
CI-NSGA-II Fix	NA	NA	NA	NA	47,0	**8.820**	5,9 e05	1881,6
CI-NSGA-II Ignition	NA	NA	NA	NA	31,6	39.520	**0,7**	**113,0**
	Medium Obstacles				**Hard Obstacles**			
	Time:300"				Time:900"			
NSGA-II	11,2	68.000	18,2 e05	3268,4	57	8.000	17,4 e05	4720,0
R-NSGA-II	22,3	54.000	10,1 e05	4565,6	56	10.400	16,2 e05	9654,0
CI-NSGA-II Vote	16,5	55.000	22,9 e05	3456,5	76,3	**7.640**	16,6 e05	4089,8
CI-NSGA-II Fix	**1,9**	**29.580**	**2,1 e05**	**1196,5**	**16,2**	8.640	**10,1 e05**	**1109,3**
CI-NSGA-II Ignition	7,7	55.480	7,2 e05	1656,7	28,9	23.300	10,8 e05	1298,2

Table 6.11: Results of fixed distance evaluation in discrete problem.

Algorithms	**Easy**				**Easy Obstacles**			
	$D_{S\to P_F}$: 20				$D_{S\to P_F}$: 20			
	Time	$Num.Eval.$	κ	Ψ	Time	$Num.Eval.$	κ	Ψ
NSGA-II	3,0	3.440	e05	2896,2	48,6	62.160	18,2 e05	2889,6
R-NSGA-II	**1,9**	**1.200**	17,2 e05	3449,8	**32,4**	**11.600**	14,7 e05	7130,4
CI-NSGA-II Vote	46,0	3.200	16,5 e05	**2272,3**	196,3	78.520	28,4	451,6
CI-NSGA-II Fix	63,3	1.380	**14,0 e05**	4101,7	86,1	20.660	4,3 e05	896,9
CI-NSGA-II Ignition	63,0	1.280	14,8 e05	5581,8	106,9	29.680	**2,2**	**10,25**
	Medium Obstacles				**Hard Obstacles**			
NSGA-II	372,4	68.080	17,5 e05	3743,4	2661,9	22.000	18,3 e05	6270,6
R-NSGA-II	**172,4**	**18.400**	16,0 e05	1081,0	2155,7	27.200	15,1 e05	4598,0
CI-NSGA-II Vote	300,0	57.640	25,1 e05	**283,5**	3058,7	34.820	17,5 e05	3098,4
CI-NSGA-II Fix	249,5	25.180	2,5 e05	1800,0	**823,1**	**4.600**	**7,7 e05**	**1267,3**
CI-NSGA-II Ignition	179,6	25.280	**1,9 e05**	1164,3	1448,8	44.100	15,02 e05	2890,9

Altogether, *CI-NSGA-II Fix* iteratively refines the search parameters and adopts players collaborations to achieve more appropriated points in the final trade-off set. It encourages the creativity and cognition to produce new solutions. Figure 6.18 demonstrates how the collective intelligence contributions in *CI-NSGA-II Fix* outperform the regular NSGA-II and R-NSGA-II from the *Medium Obstacles* scenario onwards (low values are desired). This concludes that collective intelligence and reference points enhance the MOEA results when faced with more complex scenarios. In facility location problems, the interactive MOEA could benefit from human characteristics, such as 3D spatial reasoning and strategic thinking.

Figures 6.19 and 6.20 show the number of function evaluations for each scenario in the fixed time and fixed distance evaluation, respectively. The *CI-NSGA-II Fix* consistently



(a) Fixed Time

(b) Fixed Distance on logarithmic scale

Figure 6.18: Results of distance and time measurements for each game scenario in the discrete problem.

presented the lowest values of function evaluations in the fixed time evaluation. The R-NSGA-II had the lowest values on the fixed distance evaluation, except for the *Hard Obstacles* scenario where the *CI-NSGA-II Fix* was the winner. As the scenario is getting more complex, the greater is the advantage of a collective intelligence MOEA in terms of the computation of function evaluations.

From a practical point of view, the arrangement of facilities involves large sums of capital resources. This method shows its potential use in finding a handful of preferred solutions and giving the company a competitive advantage.

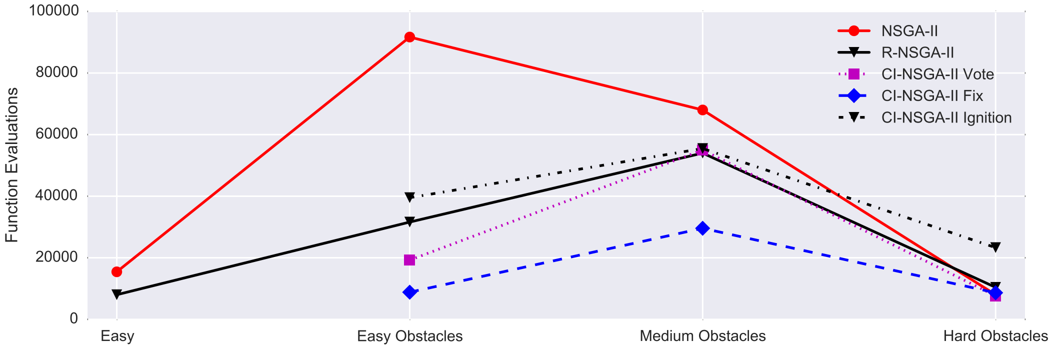


Figure 6.19: Number of function evaluations for each scenario in the fixed time evaluation. (discrete problem)

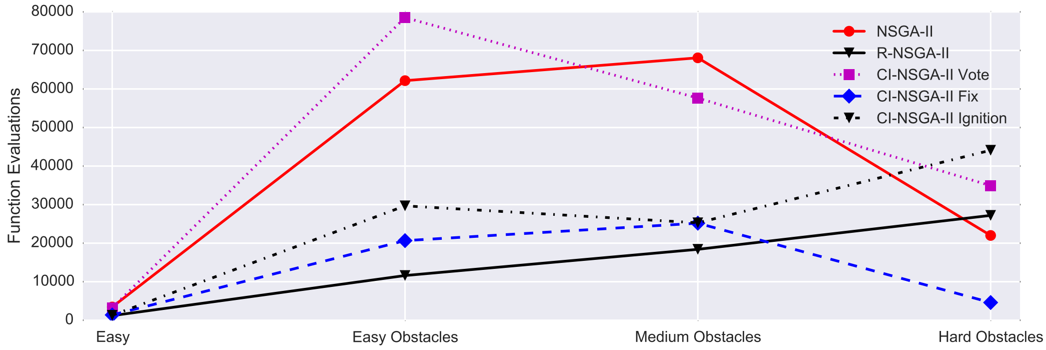


Figure 6.20: Number of function evaluations for each scenario in the fixed distance evaluation (discrete problem).

6.2.4.3 Continuous Problem

According the test results in the discrete problem (Section 6.2.4.2), the *CI-NSGA-II Fix* outperformed the others variations: *CI-NSGA-II Vote* and *CI-NSGA-II Ignition*. For this reason, the continuous experiment selects only the *CI-NSGA-II Fix* and compares it with the original NSGA-II and R-NSGA-II. The methods used to evaluate the performance of the algorithms are the same adopted in the discrete problem: number of function evaluations, fixed distance of convergence ($D_{S \to P_F}$), fixed time. But also, the preference-based indicators: R-metric, Reference Set ($C2_R$), Convex Hull (Ψ), Referential Cluster (κ) and Filatovas Spread (Δ_P); are computed in order to improve the analysis.

The NSGA-II is not a preference-based MOEA. But it is of interest to compare the convergence of NSGA-II ($D_{S \to P_F}$) with the others MOEAs. As the preference-based indicators have to measure a limited area of the P_F and the NSGA-II searches for an entire Pareto frontier, the preference-based indicators were not applied to NSGA-II.

In the fixed distance, a minimum distance between the current approximation set

Table 6.12: Results of fixed time evaluation in continuous problem.

Algorithms							
	Easy						
	Time:5"						
	$D_{S\to P_F}$	$Num.Eval.$	κ	Ψ	R-metric	$C2_R$	Δ_P
NSGA-II	**21,4**	**4.000**	–	–	–	–	–
R-NSGA-II	23,38	48.000	17,3 e05	3066,7	0,296	1278,2	0,2188
CI-NSGA-II Fix	NA	NA	NA	NA	NA	NA	NA
	Easy Obstacles						
	Time:70"						
	$D_{S\to P_F}$	$Num.Eval.$	κ	Ψ	R-metric	$C2_R$	Δ_P
NSGA-II	59,13	1.600	–	–	–	–	–
R-NSGA-II	**57,4**	**5.200**	**17,1 e05**	**3066**	**0,2962**	**1186,1**	**0.2994**
CI-NSGA-II Fix	74,2	1.200	21,0 e05	3316	0,1261	2111,9	0.5214
	Medium Obstacles						
	Time:900"						
	$D_{S\to P_F}$	$Num.Eval.$	κ	Ψ	R-metric	$C2_R$	Δ_P
NSGA-II	52,9	**1.600**	–	–	–	–	–
R-NSGA-II	44	3.200	16,1 e05	3.293	0,237	1169,8	0,4757
CI-NSGA-II Fix	**34,1**	3.200	**10,7 e05**	**2.180**	**0,332**	**1091,0**	**0,4085**
	Hard Obstacles						
	Time:12.000"						
	$D_{S\to P_F}$	$Num.Eval.$	κ	Ψ	R-metric	$C2_R$	Δ_P
NSGA-II	29,6	121.600	–	–	–	–	–
R-NSGA-II	27,8	119.887	26,34 e05	1.198	0,532	1121,8	0,1727
CI-NSGA-II Fix	**17,1**	**76.822**	**3,9 e05**	**993**	**0,899**	**989,1**	**0,0947**

S and the Pareto-optimal front is measured by the front coverage indicator, $D_{S\to P_F}$. A proximity of $D_{S\to P_F} = 20$ is the criteria to stop the evolution and compare the algorithms. In the fixed time, the algorithms run in a time interval previously defined. $C2_R$ indicator uses a reference point provided by one external user (an offshore engineer elected as the main DM) to indicate the distance between the *a priori* preference and the outcomes of COIN MOEAs.

Tables 6.12 and 6.13 present the results for fixed time and distance evaluation, respectively. The continuous problem is more complex than the discrete one. In this case, the time interval for the fixed time evaluation experiment was redefined: *Medium Obstacles* scenario have 900 seconds; *Hard Obstacles* have 12.000 seconds. In the case of fixed time: 5", there was not sufficient time to involve a collective participation of users, so the CI-NSGA-II variations were not applicable (NA).

Based on the results in Table 6.12, the original R-NSGA-II won in the simplest scenario: *Easy Obstacles*. But the *CI-NSGA-II Fix* algorithm had a better performance in the more complex ones: *Medium Obstacles* and *Hard Obstacles*.

As the complexity grows, the support of the collective intelligence and reference points enhance the MOEA results. In the *Hard Obstacles* scenario (Table 6.13), the *CI-NSGA-*

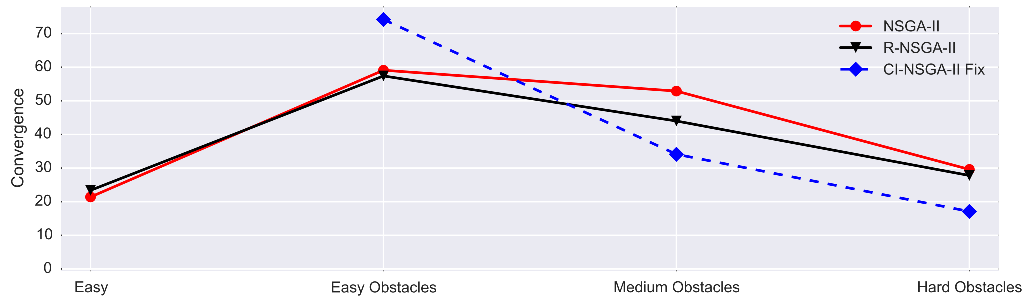
Table 6.13: Results of fixed distance evaluation in continuous problem.

Algorithms	Time	*Num.Eval.*	κ	Ψ	R-metric	$C2_R$	Δ_P
	Easy						
	$D_{S\to P_F}$: 20						
NSGA-II	7,6	**1.600**	–	–	–	–	–
R-NSGA-II	**5,8**	**1.600**	**15,1 e05**	**3304,9**	**0,932**	**2216,1**	**0,8917**
CI-NSGA-II Fix	NA	NA	NA	NA	NA	NA	NA
	Easy Obstacles						
	Time	*Num.Eval.*	κ	Ψ	R-metric	$C2_R$	Δ_P
NSGA-II	274,6	13.600	–	–	–	–	–
R-NSGA-II	**271,5**	12.800	**16,2 e05**	6999,5	0,333	981,0	**0,877**
CI-NSGA-II Fix	366,7	**6.800**	58,9 e05	**1991,1**	**0,422**	**977,8**	0,910
	Medium Obstacles						
	Time	*Num.Eval.*	κ	Ψ	R-metric	$C2_R$	Δ_P
NSGA-II	4702,7	22.000	–	–	–	–	–
R-NSGA-II	3620,7	11.200	22,9 e05	1991,1	0,231	1020,8	0,4799
CI-NSGA-II Fix	**1204,9**	**9.635**	**12,8 e05**	**1873,0**	**0,636**	**921,1**	**0,3712**
	Hard Obstacles						
	Time	*Num.Eval.*	κ	Ψ	R-metric	$C2_R$	Δ_P
NSGA-II	490.732	120.873	–	–	–	–	–
R-NSGA-II	345.600	89.200	16,2 e05	1911,1	0,112	1532,5	0,3147
CI-NSGA-II Fix	**92.965**	**36.912**	**16,9 e05**	**1239**	**0,903**	**925,1**	**0,2714**

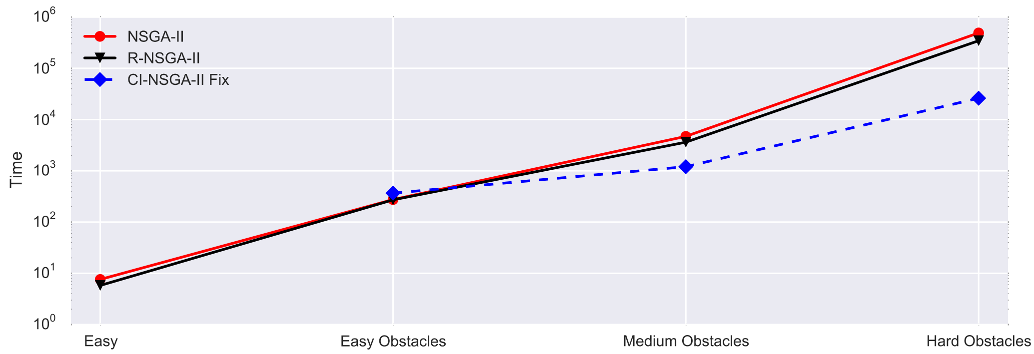
II fix required almost 3 times less function evaluation and performed 3.71 times faster than R-NSGA-II. When compared with the NSGA-II, the *CI-NSGA-II fix* performed 5.27 times faster. The number of evaluation function computed by CI-NSGA was smaller since the *Easy Obstacles* scenario. The $C2_R$ values indicate that *CI-NSGA-II fix* results are closer to the *a priori* reference point provided by the offshore engineer elected as the main DM. R-metric values were also superior for this algorithm.

As in the discrete problem, Figure 6.21 demonstrates how the collective intelligence contributions in *CI-NSGA-II Fix* outperform the regular NSGA-II and R-NSGA-II from the *Medium Obstacles* scenario onwards (low values are desired).

Figures 6.22 and 6.23 show the number of function evaluations for each scenario in the fixed time and fixed distance evaluation, respectively. In the continuous problem, the *CI-NSGA-II Fix* consistently presented the lowest values of function evaluations in the fixed time and distance evaluation. The COIN operator and users' rational collaborations decrease the number of function evaluations, accelerate the convergence and achieve relevant regions of Pareto front at a lower computational cost.



(a) Fixed Time



(b) Fixed Distance on logarithmic scale

Figure 6.21: Results of distance and time measurements for each game scenario in the continuous problem.

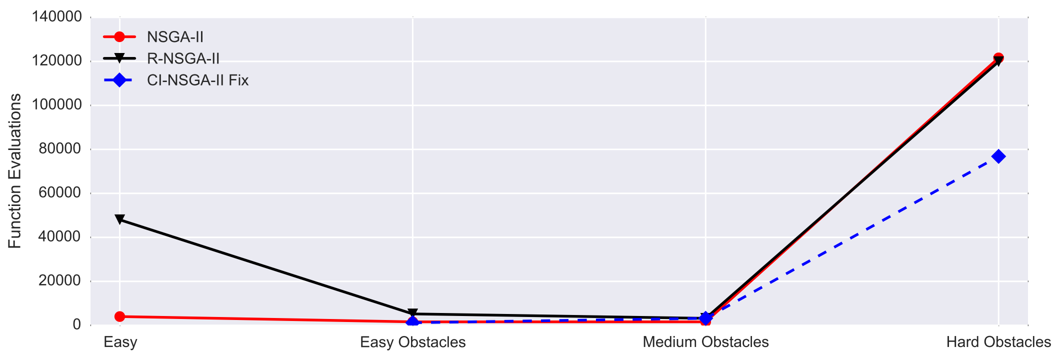


Figure 6.22: Number of function evaluations for each scenario in the fixed time evaluation (continuous problem).

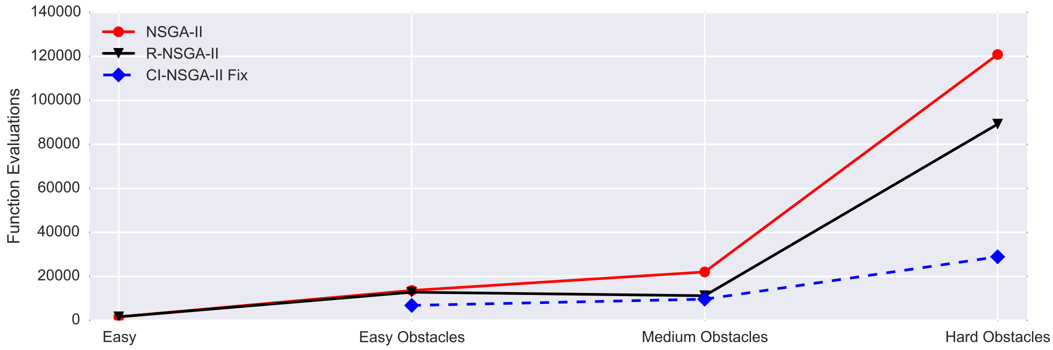


Figure 6.23: Number of function evaluations for each scenario in the fixed distance evaluation (continuous problem).

Chapter 7

Conclusions and Future Work

7.1 Final Remarks

In this work we have discussed about group preferences in multi-objective optimization evolutionary algorithms and have introduced a novel approach that brings human subjectivity and cognition into the optimization process. MOEAs can take advantage of decision makers' preferences to guide the search through relevant regions of Pareto-optimal front. Suitable techniques of preference-based multi-objective algorithms and interactive EAs were pointed out as an alternative to handle the dynamism not expected by *a priori* methods. But, particularly, the CI-NSGA-II, CI-SMS-EMOA and CI-SPEA2 were presented as an interactive approach supported by dynamic group preferences.

The new algorithms apprehend people's heterogeneity and common sense to improve the successive stages of evolution in a direct crowdsourcing fashion. Consequently, instead of the entire front, it reaches a smaller sub-set of the front and uses the collective preferences to support decisions upon multi-objective situations. The wisdom arisen from the diversity of many individuals is able to enhance MOEAs, overcome their difficulties and discover creative resolutions.

The approaches have been tested successfully in benchmarking problems. Three different performance indicators (Referential Cluster Variance, Convex Hull Volume and Reference Set Distance) were presented with the intention to measure the proportion of occupied area in P_F.

A real-world case study regarding facility location was tested successfully against the algorithms. The multi-objective scenario was reproduced as a game and directed to a collective intelligence support. Results outlined the benefits of collective reference points to unfold solutions designed by a group of people that is more intelligent when is working together.

The injection of COIN within the original NSGA-II, SPEA2 and SMS-EMOA did not make radical changes in their structure. Therefore, the research community should be able to improve and extend the actual achievements.

In the near future, the continuity of this research will explore different features of the evolutionary process. There is a particular interest in more complex scenarios with many constraints and non-explicit objectives hidden in the problem. It is important to validate if the complexity of the environment will favour even more the integration of COIN in MOEAs. Also, this work wants to create an open platform and different web-scenarios to apply collective intelligence in different multi-objective problems.

Furthermore, there are plans to apply directional information along with the projection of the collective reference points during the evolution process. This way, the technique can extract the intelligence of the crowds and, at the same time, optimize the search through preferred regions with a minimal number of interruptions in the algorithm.

7.2 Future Directions

This work uncovered some directions for future research. One further research topic in the field of collective intelligence and interactive multi-objective optimization could be the implementation of aggregation functions to merge different users' collaboration. Once imported in the next population, the individuals produced in the collective environment with rational supervision are modified by the genetic operators. The crossover and mutation combine or alter the individuals based on random operations. There is a need to develop more effective aggregation functions to join cognitive inputs from different people in the optimization process.

In the same context, the participants could collaborate on the same game scenario. Today, each user edits his own scenario. There is no interaction between the participants. But one can think about how to share the same intermediate solution to many collaborators. Individuals produced by one player can be updated by others users, a real-time editing executed by multiple people simultaneously. This way, not only the aggregation function, but also the collaborative environment will explore the potential and benefits of the 3C model [80, 154] – cooperation, coordination and communication – to address rational improvements to the evolution.

All the objectives in the problem are explicit. Another potential research interest in the future could be the analysis of MOPs with non-explicit objectives hidden in the problem. It should be considered if the collective intelligence uses the common sense to extend the declared objective functions and take into account different constraints or goals not stated. To this end, one can investigate if the provided answers satisfy the explicit objectives, but also the hidden ones.

Based on this work, another important development field is the analysis of which kind of previous information affects the COIN contributions or preferences. For instance, if the DM expresses part of his preferences, maybe the results would be different. But, it should be determined which categories of information influence the results and at what level they change the collective preferences.

The gamification of this problem can also be improved. The game platform (Playcanvas) allows the creation of different web-scenarios to apply collective intelligence in distinct multi-objective problems. In this regard, this work motivates the development of an open platform to diffuse multi-objectives problems and receive multiple collaborators to help solving these MOPs.

References

[1] Ivar aasen field development. http://www.offshoreenergytoday.com/norway-approves-ivar-aasen-field-development/. Accessed: 2016-08-02.

[2] Subsea development of an oil and gas field. http://subseaexplorer.net/technip-awarded-important-contract-for-dvalin-subsea-development/. Accessed: 2017-02-17.

[3] Technip business activities. http://www.technip.com/en/media-center/photo-library. Accessed: 2015-07-22.

[4] Technip field development planning. http://www.technip.com/en/our-business/services/field-development-planning. Accessed: 2016-09-28.

[5] Tools for pipelaying and subsea equipment installation. https://www.royalihc.com/en/about-us/specialised-business-units/ihc-iqip/oil-and-gas/subsea-field-development. Accessed: 2017-02-5.

[6] Success stories. http://www.affinnova.com/clients/success-stories, 2014. [Online; accessed 22-June-2014].

[7] Conover test of variances. https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/PASS/Conover_Test_of_Variances-Simulation.pdf, 2017. [Online; accessed 27-Mar-2017].

[8] Service oriented architecture (soa). https://msdn.microsoft.com/en-us/library/bb833022.aspx, 2017. [Online; accessed 1-February-2017].

[9] Well-known users of sqlite. http://www.sqlite.org/famous.html, 2017. [Online; accessed 2-March-2017].

[10] Agency, I. E. *Key World Energy Statistics*. 2016. [Online; accessed 22-Feb-2017].

[11] Aickelin, U. An indirect genetic algorithm for set covering problems. *Journal of the Operational Research Society 53*, 10 (2002), 1118–1126.

[12] Arrow, K. J. A difficulty in the concept of social welfare. *The Journal of Political Economy* (1950), 328–346.

[13] Auger, A., Bader, J., Brockhoff, D., Zitzler, E. Articulating user preferences in many-objective problems by sampling the weighted hypervolume. In *Proceedings of the 11th Annual conference on Genetic and evolutionary computation* (2009), ACM, p. 555–562.

[14] Babbar-Sebens, M., Minsker, B. S. Interactive genetic algorithm with mixed initiative interaction for multi-criteria ground water monitoring design. *Applied Soft Computing 12*, 1 (2012), 182–195.

[15] Bäck, T. *Evolutionary algorithms in theory and practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms.* Oxford University Press, New York, 1996.

[16] Back, T. *Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms.* Oxford university press, 1996.

[17] Bag, P. K., Sabourian, H., Winter, E. Multi-stage voting, sequential elimination and condorcet consistency. *Journal of Economic Theory 144*, 3 (2009), 1278–1299.

[18] Barber, C. B., Dobkin, D. P., Huhdanpaa, H. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software (TOMS) 22*, 4 (1996), 469–483.

[19] Barber, D. *Bayesian reasoning and machine learning.* Cambridge University Press, 2012.

[20] Bartle, R. Hearts, clubs, diamonds, spades: Players who suit muds. *Journal of MUD research 1*, 1 (1996), 19.

[21] Bartle, R. A. *Designing virtual worlds.* New Riders, 2004.

[22] Basgalupp, M. P., Barros, R. C., De Carvalho, A. C., Freitas, A. A. Evolving decision trees with beam search-based initialization and lexicographic multi-objective evaluation. *Information Sciences 258* (2014), 160–181.

[23] Beasley, D., Martin, R., Bull, D. An overview of genetic algorithms: Part 1. fundamentals. *University computing 15* (1993), 58–58.

[24] Bechikh, S., Said, L. B., Ghédira, K. 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 5*, 278-288 (2013), 71.

[25] Ben Said, L., Bechikh, S., Ghédira, K. The r-dominance: a new dominance relation for interactive evolutionary multicriteria decision making. *Evolutionary Computation, IEEE Transactions on 14*, 5 (2010), 801–818.

[26] Benamara, F., Kaci, S., Pigozzi, G. Collective decision making with individual confidence scores in the decision rule. In *Annales du Lamsade, Proceedings of the DIMACS-LAMSADE Workshop on Algorithmic Decision Theory* (2008), p. 29–45.

[27] Benson, H. P. An outer approximation algorithm for generating all efficient extreme points in the outcome set of a multiple objective linear programming problem. *Journal of Global Optimization 13*, 1 (1998), 1–24.

[28] Beume, N., Naujoks, B., Emmerich, M. Sms-emoa: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research 181*, 3 (2007), 1653–1669.

[29] Bigham, J. P., Jayant, C., Ji, H., Little, G., Miller, A., Miller, R. C., Miller, R., Tatarowicz, A., White, B., White, S., others. Vizwiz: nearly real-time answers to visual questions. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology* (2010), ACM, p. 333–342.

[30] Bischoff, M., Fleischmann, T., Klamroth, K. The multi-facility location–allocation problem with polyhedral barriers. *Computers & Operations Research 36*, 5 (2009), 1376–1392.

[31] Bleuler, S., Laumanns, M., Thiele, L., Zitzler, E. PISA — a platform and programming language independent interface for search algorithms. In *Evolutionary Multi-Criterion Optimization (EMO 2003)* (Berlin, 2003), C. M. Fonseca, P. J. Fleming, E. Zitzler, K. Deb, and L. Thiele, Eds., Lecture Notes in Computer Science, Springer, p. 494 – 508.

[32] Blohm, I., Riedl, C., Leimeister, J. M., Krcmar, H. Idea evaluation mechanisms for collective intelligence in open innovation communities: Do traders outperform raters? In *ICIS* (2011).

[33] Bodenhofer, U. Genetic algorithms: theory and applications, 2003.

[34] Bosman, P. A., Thierens, D. The balance between proximity and diversity in multiobjective evolutionary algorithms. *Evolutionary Computation, IEEE Transactions on 7*, 2 (2003), 174–188.

[35] Boyd, S., Vandenberghe, L. *Convex optimization*. Cambridge university press, 2004.

[36] Branke, J., Kaussler, T., Schmeck, H. Guidance in evolutionary multi-objective optimization. *Advances in Engineering Software 32*, 6 (2001), 499–507.

[37] Brockhoff, D., Bader, J., Thiele, L., Zitzler, E. Directed Multiobjective Optimization Based on the Hypervolume Indicator. *Journal of Multi-Criteria Decision Analysis 20* (2013), 291ï¿½–317.

[38] Ceria, S., Nobili, P., Sassano, A. A lagrangian-based heuristic for large-scale set covering problems. *Mathematical Programming 81*, 2 (1998), 215–228.

[39] Cheng, R., Jin, Y., Olhofer, M., Sendhoff, B. Test problems for large-scale multiobjective and many-objective optimization. *IEEE Transactions on Cybernetics* (2016).

[40] Chvatal, V. A greedy heuristic for the set-covering problem. *Mathematics of operations research 4*, 3 (1979), 233–235.

[41] Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. Collaborative preferences in multi-objective evolutionary algorithms. In *Proceedings of the 12th Brazilian Symposium on Collaborative Systems - Research Paper*, SBSC '15.

[42] Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. Preference-based interactive moea on continuous problem of facility location. In *Workshop de Pesquisa e Desenvolvimento em Inteligência Artificial* (2016).

[43] Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. Extending collective intelligence evolutionary algorithms: A facility location problem application. In *11th Learning and Intelligent Optimization Conference* (2017), LION '17.

[44] Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. B. Collective preferences in evolutionary multi-objective optimization: Techniques and potential contributions of collective intelligence. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing* (New York, NY, USA, 2015), SAC '15, ACM, p. 133–138.

[45] Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. B. Integrating collective intelligence into evolutionary multi-objective algorithms: Interactive preferences. In *LA-CCI (Latin American) Congress on Computational Intelligence* (2015), LA-CCI.

[46] Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. B. Using collective intelligence to support multi-objective decisions: collaborative and online preferences. In *IEEE/ACM 30th International Conference on Automated Software Engineering Workshops* (2015), ASEW.

[47] Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. B. Bio-inspired algorithms and preferences for multi-objective problems. In *11th International Conference on Hybrid Artificial Intelligence Systems* (2016), HAIS '16.

[48] Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. B. *Evolutionary Multi-Objective System Design: Theory and Applications*. Chapman & Hall/CRC Computer and Information Science Series. CRC Press, 2017, cap. Hybrid Multi-Objective Evolutionary Algorithms with Collective Intelligence.

[49] Coello, C. A. C., Dhaenens, C., Jourdan, L. Multi-objective combinatorial optimization: Problematic and context. In *Advances in multi-objective nature inspired computing*. Springer, 2010, p. 1–21.

[50] Coello, C. C., Lamont, G. B., Van Veldhuizen, D. A. *Evolutionary algorithms for solving multi-objective problems*. Springer, 2007.

[51] Collette, Y., Siarry, P. *Multiobjective optimization: principles and case studies*. Springer, 2003.

[52] Conitzer, V. Eliciting single-peaked preferences using comparison queries. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems* (2007), ACM, p. 65.

[53] Conover, W. *Practical Nonparametric Statistics*. John Wiley and Sons, New York, 1999.

[54] Current, J., Daskin, M., Schilling, D. 3 d iscrete n etw ork location m odels. *Facility Location Applications and Theory. Springer* (2004), 81–118.

[55] Custódio, A., Emmerich, M., Maderia, J. Recent developments in derivative-free multiobjective optimization. *Computational Technology Reviews 5* (2012), 1–30.

[56] Cvetkovic, D., Parmee, I. C. Evolutionary design and multi–objective optimisation. In *6th European Congress on Intelligent Techniques and Soft Computing EUFIT* (1998), vol. 98, Citeseer, p. 397–401.

[57] Czyzzak, P., Jaszkiewicz, A. Pareto simulated annealing: a metaheuristic technique for multiple-objective combinatorial optimization. *Journal of Multi-Criteria Decision Analysis 7*, 1 (1998), 34–47.

[58] Darwin, C. *The origin of species.* Lulu. com, 1872.

[59] Das, I., Dennis, J. E. Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems. *SIAM Journal on Optimization 8*, 3 (1998), 631–657.

[60] Deb, K., others. *Multi-objective optimization using evolutionary algorithms*, vol. 2012. John Wiley & Sons Chichester, 2001.

[61] Deb, K., Jain, H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach: solving problems with box constraints. *Evolutionary Computation, IEEE Transactions 18*, 4 (2014), 577–601.

[62] Deb, K., Kumar, A. Interactive evolutionary multi-objective optimization and decision-making using reference direction method. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation* (2007), ACM, p. 781–788.

[63] Deb, K., Kumar, A. Light beam search based multi-objective optimization using evolutionary algorithms. In *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on* (2007), IEEE, p. 2125–2132.

[64] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Evolutionary Computation, IEEE Transactions on 6*, 2 (2002), 182–197.

[65] Deb, K., Sinha, A., Korhonen, P. J., Wallenius, J. An interactive evolutionary multiobjective optimization method based on progressively approximated value functions. *Evolutionary Computation, IEEE Transactions on 14*, 5 (2010), 723–739.

[66] Deb, K., Sundar, J., Udaya Bhaskara Rao, N., Chaudhuri, S. Reference point based multi-objective optimization using evolutionary algorithms. *International Journal of Computational Intelligence Research 2*, 3 (2006), 273–286.

[67] Deb, K., Thiele, L., Laumanns, M., Zitzler, E. *Scalable test problems for evolutionary multiobjective optimization.* Springer, 2005.

[68] Dempster, A. P., Laird, N. M., Rubin, D. B. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)* (1977), 1–38.

[69] Domínguez, A., Saenz-de Navarrete, J., De-Marcos, L., Fernández-Sanz, L., Pagés, C., Martínez-Herráiz, J.-J. Gamifying learning experiences: Practical implications and outcomes. *Computers & Education 63* (2013), 380–392.

[70] Drezner, Z., Hamacher, H. W. *Facility location: applications and theory.* Springer Science & Business Media, 2001.

[71] Ebert, J. Soa with rest: principles, patterns & constraints for building enterprise solutions with rest by thomas erl, benjamin carlyle, cesare pautasso, raj balasubramanian. *ACM SIGSOFT Software Engineering Notes 38*, 3 (2013), 32–33.

[72] Edelsbrunner, H. Alpha shapes a survey. *Tessellations in the Sciences 27* (2010).

[73] Ehrgott, M. *Multicriteria optimization*, vol. 2. Springer, 2005.

[74] Ehrgott, M., Gandibleux, X. A survey and annotated bibliography of multiobjective combinatorial optimization. *OR-Spektrum 22*, 4 (2000), 425–460.

[75] Emmerich, M., Beume, N., Naujoks, B. An emo algorithm using the hypervolume measure as selection criterion. In *Evolutionary Multi-Criterion Optimization* (2005), Springer, p. 62–76.

[76] Filatovas, E., Kurasova, O., Sindhya, K. Synchronous r-nsga-ii: An extended preference-based evolutionary algorithm for multi-objective optimization. *Informatica 26*, 1 (2015), 33–50.

[77] Fleischer, M. The measure of pareto optima applications to multi-objective metaheuristics. In *International Conference on Evolutionary Multi-Criterion Optimization* (2003), Springer, p. 519–533.

[78] Fort, K., Adda, G., Cohen, K. B. Amazon mechanical turk: Gold mine or coal mine? *Computational Linguistics 37*, 2 (2011), 413–420.

[79] Fortin, F.-A., De Rainville, F.-M., Gardner, M.-A., Parizeau, M., Gagné, C. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research 13* (jul 2012), 2171–2175.

[80] Fuks, H., Raposo, A., Gerosa, M. A., others. The 3c collaboration model. In *Encyclopedia of E-collaboration*. IGI Global, 2008, p. 637–644.

[81] Gendreau, M., Potvin, J.-Y. *Handbook of metaheuristics*, vol. 2. Springer, 2010.

[82] Gong, M., Liu, F., Zhang, W., Jiao, L., Zhang, Q. Interactive moea/d for multi-objective decision making. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation* (2011), ACM, p. 721–728.

[83] Greenstein, S., Zhu, F. Collective intelligence and neutral point of view: The case of wikipedia. Relatório Técnico, National Bureau of Economic Research, 2012.

[84] Grinstead, C. M., Snell, J. L. *Introduction to probability.* American Mathematical Soc., 2012.

[85] Gulley, N. Patterns of innovation: a web-based matlab programming contest. In *CHI'01 Extended Abstracts on Human Factors in Computing Systems* (2001), ACM, p. 337–338.

[86] Haldar, S. *Inside sqlite.* " O'Reilly Media, Inc.", 2007.

[87] Hamacher, H. W., Nickel, S. Classification of location models. *Location Science 6*, 1 (1998), 229–242.

[88] Hansen, M. P., Jaszkiewicz, A. *Evaluating the quality of approximations to the non-dominated set.* IMM, Department of Mathematical Modelling, Technical Universityof Denmark, 1998.

[89] Haouari, M., Chaouachi, J. A probabilistic greedy search algorithm for combinatorial optimisation with application to the set covering problem. *Journal of the Operational Research Society 53*, 7 (2002), 792–799.

[90] Hettenhausen, J., Lewis, A., Mostaghim, S. Interactive multi-objective particle swarm optimization with heatmap-visualization-based user interface. *Engineering Optimization 42*, 2 (2010), 119–139.

[91] Heylighen, F. Collective intelligence and its implementation on the web: algorithms to develop a collective mental map. *Computational & Mathematical Organization Theory 5*, 3 (1999), 253–280.

[92] Heylighen, F. Self-organization in communicating groups: the emergence of coordination, shared references and collective intelligence. In *Complexity Perspectives on Language, Communication and Society.* Springer, 2013, p. 117–149.

[93] Holland, J. H. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence.* U Michigan Press, 1975.

[94] Huband, S., Hingston, P., Barone, L., While, L. A review of multiobjective test problems and a scalable test problem toolkit. *Evolutionary Computation, IEEE Transactions on 10*, 5 (2006), 477–506.

[95] Igel, C., Hansen, N., Roth, S. Covariance matrix adaptation for multi-objective optimization. *Evolutionary computation 15*, 1 (2007), 1–28.

[96] Ishibuchi, H., Akedo, N., Nojima, Y. Behavior of multiobjective evolutionary algorithms on many-objective knapsack problems. *IEEE Transactions on Evolutionary Computation 19*, 2 (2015), 264–283.

[97] Ishibuchi, H., Akedo, N., Ohyanagi, H., Nojima, Y. Behavior of emo algorithms on many-objective optimization problems with correlated objectives. In *Evolutionary Computation (CEC), 2011 IEEE Congress on* (2011), IEEE, p. 1465–1472.

[98] Ishibuchi, H., Masuda, H., Nojima, Y. Pareto fronts of many-objective degenerate test problems. *IEEE Transactions on Evolutionary Computation 20*, 5 (2016), 807–813.

[99] Ishibuchi, H., Tsukamoto, N., Nojima, Y. Evolutionary many-objective optimization: A short review. In *Evolutionary Computation, 2008. CEC 2008.(IEEE World Congress on Computational Intelligence). IEEE Congress on* (2008), IEEE, p. 2419–2426.

[100] Jaimes, A. L., Martinez, S. Z., Coello, C. A. C. An introduction to multiobjective optimization techniques. *Optimization in Polymer Processing* (2009), 29–57.

[101] Jiang, S., Ong, Y.-S., Zhang, J., Feng, L. Consistencies and contradictions of performance metrics in multiobjective optimization. *IEEE transactions on cybernetics 44*, 12 (2014), 2391–2404.

[102] Karp, R. M. Reducibility among combinatorial problems. In *Complexity of computer computations*. Springer, 1972, p. 85–103.

[103] Khare, V., Yao, X., Deb, K. Performance scaling of multi-objective evolutionary algorithms. In *International Conference on Evolutionary Multi-Criterion Optimization* (2003), Springer, p. 376–390.

[104] Khatib, F., DiMaio, F., Cooper, S., Kazmierczyk, M., Gilski, M., Krzywda, S., Zabranska, H., Pichova, I., Thompson, J., Popović, Z., others. Crystal structure of a monomeric retroviral protease solved by protein folding game players. *Nature structural & molecular biology 18*, 10 (2011), 1175–1177.

[105] Klein, M. How to harvest collective wisdom on complex problems: An introduction to the mit deliberatorium. *Center for Collective Intelligence working paper* (2011).

[106] Klose, A., Drexl, A. Facility location models for distribution system design. *European Journal of Operational Research 162*, 1 (2005), 4–29.

[107] Knowles, J., Thiele, L., Zitzler, E. A tutorial on the performance assessment of stochastic multiobjective optimizers. *Tik report 214* (2006), 327–332.

[108] Krafzig, D., Banke, K., Slama, D. *Enterprise SOA: service-oriented architecture best practices*. Prentice Hall Professional, 2005.

[109] Kulin, H. W., Kuenne, R. E. An efficient algorithm for the numerical solution of the generalized weber problem in spatial economics. *Journal of Regional Science 4*, 2 (1962), 21–33.

[110] Lan, G., DePuy, G. W., Whitehouse, G. E. An effective and simple heuristic for the set covering problem. *European journal of operational research 176*, 3 (2007), 1387–1403.

[111] Lévy, P., Bonomo, R. *Collective intelligence: Mankind's emerging world in cyberspace*. Perseus Publishing, 1999.

[112] Li, K., Deb, K. Performance assessment for preference-based evolutionary multiobjective optimization using reference points. *COIN Report*, 2016001 (2016), 1–23.

[113] Li, X. A non-dominated sorting particle swarm optimizer for multiobjective optimization. In *Genetic and Evolutionary Computation – GECCO 2003* (2003), Springer, p. 37–48.

[114] Li, X. Better spread and convergence: Particle swarm multiobjective optimization using the maximin fitness function. In *Genetic and Evolutionary Computation–GECCO 2004* (2004), Springer, p. 117–128.

[115] Logsdon, J. M., Williamson, R. A., Launius, R. D., Acker, R. J., Garber, S. J., Friedman, J. L. Exploring the unknown: Selected documents in the history of the us civil space program. volume 4; accessing space.

[116] López-Ibáñez, M., Knowles, J. Machine decision makers as a laboratory for interactive emo. In *International Conference on Evolutionary Multi-Criterion Optimization* (2015), Springer, p. 295–309.

[117] Malone, T. W., Laubacher, R., Dellarocas, C. Harnessing crowds: Mapping the genome of collective intelligence.

[118] Marquis, J., Gel, E. S., Fowler, J. W., Köksalan, M., Korhonen, P., Wallenius, J. Impact of number of interactions, different interaction patterns, and human inconsistencies on some hybrid evolutionary multiobjective optimization algorithms. *Decision Sciences 46*, 5 (2015), 981–1006.

[119] Martí, L. *Scalable multi-objective optimization.* Tese de Doutorado, Ph. D. Thesis, Departmento de InformÃatica, Universidad Carlos III de Madrid, Colmenarejo, Spain, 2011.

[120] Martí, L. Evolutionary multi-objective optimization. `http://lmarti.github.io/emo-course-lncc/EvolutionaryMulti-ObjectiveOptimization.slides.html`, 2016. [Online; accessed 2016-08-21].

[121] Martí, L., Grimme, C., Kerschke, P., Trautmann, H., Rudolph, G. Averaged hausdorff approximations of pareto fronts based on multiobjective estimation of distribution algorithms. In *Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation* (2015), ACM, p. 1427–1428.

[122] Martí, L., Sanchez-Pi, N. Convergence detection and stopping criteria for evolutionary multi-objective optimization. Tutorial presented at the 2015 IEEE Congress on Evolutionary Computation (CEC2015), Sendai, Japan.

[123] Martinez, S. Z., Coello, C. A. C. An archiving strategy based on the convex hull of individual minima for moeas.

[124] Mavrotas, G. Effective implementation of the ε-constraint method in multi-objective mathematical programming problems. *Applied mathematics and computation 213*, 2 (2009), 455–465.

[125] Megiddo, N., Supowit, K. J. On the complexity of some common geometric location problems. *SIAM journal on computing 13*, 1 (1984), 182–196.

[126] Megiddo, N., Tamir, A. On the complexity of locating linear facilities in the plane. *Operations research letters 1*, 5 (1982), 194–197.

[127] Mitchell, M. *An introduction to genetic algorithms.* MIT press, 1998.

[128] Mohammadi, A., Omidvar, M. N., Li, X. A new performance metric for user-preference based multi-objective evolutionary algorithms. In *2013 IEEE Congress on Evolutionary Computation* (2013), IEEE, p. 2825–2832.

[129] Mohan, C. K., Mehrotra, K. G. Reference set metrics for multi-objective algorithms. In *International Conference on Swarm, Evolutionary, and Memetic Computing* (2011), Springer, p. 723–730.

[130] Monfared, M. D., Mohades, A., Rezaei, J. Convex hull ranking algorithm for multi-objective evolutionary algorithms. *Scientia Iranica 18*, 6 (2011), 1435–1442.

[131] Mukhlisullina, D., Passerini, A., Battiti, R. Learning to diversify in complex interactive multiobjective optimization. In *Metaheuristics International Conference (MIC 2013)* (2013).

[132] Mukhopadhyay, S., Banerjee, S. Cooperating swarms: a paradigm for collective intelligence and its application in finance. *International Journal of Computer Applications 6*, 10 (2010), 31–41.

[133] Nagar, Y. *Combining human and machine intelligence for making predictions.* Tese de Doutorado, Massachusetts Institute of Technology, 2013.

[134] Ohlsson, M., Peterson, C., Söderberg, B. An efficient mean field approach to the set covering problem. *European Journal of Operational Research 133*, 3 (2001), 583–595.

[135] Ojalehto, V., Podkopaev, D., Miettinen, K. Towards automatic testing of reference point based interactive methods. In *International Conference on Parallel Problem Solving from Nature* (2016), Springer, p. 483–492.

[136] Pelleg, D., Moore, A. W., others. X-means: Extending k-means with efficient estimation of the number of clusters. In *ICML* (2000), p. 727–734.

[137] Pfeiffer, J., Golle, U., Rothlauf, F. Reference point based multi-objective evolutionary algorithms for group decisions. In *Proceedings of the 10th annual conference on Genetic and evolutionary computation* (2008), ACM, p. 697–704.

[138] Riquelme, N., Von Lücken, C., Baran, B. Performance metrics in multi-objective optimization. In *Computing Conference (CLEI), 2015 Latin American* (2015), IEEE, p. 1–11.

[139] Rudolph, G., Schütze, O., Grimme, C., Trautmann, H. An aspiration set emoa based on averaged hausdorff distances. In *International Conference on Learning and Intelligent Optimization* (2014), Springer, p. 153–156.

[140] Sastry, K., Goldberg, D. E., Kendall, G. Genetic algorithms. In *Search methodologies.* Springer, 2014, p. 93–117.

[141] Sawaragi, Y., Nakayam, H., Tanino, T. *Theory of multiobjective optimization.* Elsevier, 1985.

[142] Schutze, O., Esquivel, X., Lara, A., Coello, C. A. C. Using the averaged hausdorff distance as a performance measure in evolutionary multiobjective optimization. *IEEE Transactions on Evolutionary Computation 16*, 4 (2012), 504–522.

[143] Schütze, O., Laumanns, M., Tantar, E., Coello, C. A. C., Talbi, E.-G. Computing gap free pareto front approximations with stochastic search algorithms. *Evol. Comput. 18*, 1 (marï¿½ de 2010), 65–96.

[144] Shan-Fan, J., Xiong, S.-W., Zhuo-Wang, J. The multi-objective differential evolution algorithm based on quick convex hull algorithms. In *Natural Computation, 2009. ICNC'09. Fifth International Conference on* (2009), vol. 4, IEEE, p. 469–473.

[145] Silverman, B. W. *Density estimation for statistics and data analysis*, vol. 26. CRC press, 1986.

[146] Smyth, P. The em algorithm for gaussian mixtures. `http://www.ics.uci.edu/~smyth/courses/cs274/background_notes.html`, 2014. [Online; accessed 11-Oct-2014].

[147] Stewart, T. Goal programming and cognitive biases in decision-making. *Journal of the Operational Research Society 56*, 10 (2005), 1166–1175.

[148] Sun, X., Yang, L., Gong, D., Li, M. Interactive genetic algorithm assisted with collective intelligence from group decision making. In *Evolutionary Computation (CEC), 2012 IEEE Congress on* (2012), IEEE, p. 1–8.

[149] Surowiecki, J. *The wisdom of crowds.* Random House LLC, 2005.

[150] Tan, K. C., Lee, T. H., Khor, E. F. Evolutionary algorithms for multi-objective optimization: performance assessments and comparisons. *Artificial intelligence review 17*, 4 (2002), 251–290.

[151] Tan, P.-N., Steinbach, M., Kumar, V. *Introduction to Data Mining.* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2006.

[152] Taylor, R. N., Medvidovic, N., Dashofy, E. M. *Software architecture: foundations, theory, and practice.* Wiley Publishing, 2009.

[153] Tellier, L.-N. The weber problem: solution and interpretation. *Geographical Analysis 4*, 3 (1972), 215–233.

[154] Thalheim, B., Jaakkola, H., Nakanishi, T., Sasaki, S., Schewe, K.-D. Conceptual modelling of collaboration for information systems. In *EJC* (2013), p. 272–305.

[155] Thiele, L., Miettinen, K., Korhonen, P. J., Molina, J. A preference-based evolutionary algorithm for multi-objective optimization. *Evolutionary Computation 17*, 3 (2009), 411–436.

[156] Van Veldhuizen, D. A. Multiobjective evolutionary algorithms: classifications, analyses, and new innovations. Relatório Técnico, DTIC Document, 1999.

[157] Van Veldhuizen, D. A., Lamont, G. B. On measuring multiobjective evolutionary algorithm performance. In *Evolutionary Computation, 2000. Proceedings of the 2000 Congress on* (2000), vol. 1, IEEE, p. 204–211.

[158] Voss, T., Beume, N., Rudolph, G., Igel, C. Scalarization versus indicator-based selection in multi-objective cma evolution strategies. In *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)* (2008), IEEE, p. 3036–3043.

[159] Vygen, J. *Approximation algorithms facility location problems.* Forschungsinstitut für Diskrete Mathematik, Rheinische Friedrich-Wilhelms-Universität, 2005.

[160] Wagner, T., Trautmann, H. Integration of preferences in hypervolume-based multiobjective evolutionary algorithms by means of desirability functions. *Evolutionary Computation, IEEE Transactions on 14*, 5 (2010), 688–701.

[161] Wang, P., Emmerich, M., Li, R., Tang, K., Baeck, T., others. Convex hull-based multi-objective genetic programming for maximizing receiver operating characteristic performance.

[162] Wang, P., Emmerich, M., Li, R., Tang, K., Yao, X., others. Convex hull-based multi-objective genetic programming for maximizing roc performance. *arXiv preprint arXiv:1303.3145* (2013).

[163] Watkins, J. H., Rodriguez, M. A. A survey of web-based collective decision making systems. In *Evolution of the Web in Artificial Intelligence Environments.* Springer, 2008, p. 243–277.

[164] Weiszfeld, E. Sur le point pour lequel la somme des distances de n points donnés est minimum. *Tohoku Mathematical Journal, First Series 43* (1937), 355–386.

[165] Wickramasinghe, U. K., Li, X. Integrating user preferences with particle swarms for multi-objective optimization. In *Proceedings of the 10th annual conference on Genetic and evolutionary computation* (2008), ACM, p. 745–752.

[166] Wickramasinghe, U. K., Li, X. Using a distance metric to guide pso algorithms for many-objective optimization. In *Proceedings of the 11th Annual conference on Genetic and evolutionary computation* (2009), ACM, p. 667–674.

[167] Wierzbicki, A. P. The use of reference objectives in multiobjective optimization. In *Multiple criteria decision making theory and application.* Springer, 1980, p. 468–486.

[168] Wierzbicki, A. P. Reference point approaches and objective ranking. *Practical approaches to multi-objective optimization*, 06501 (2006).

[169] Yang, X.-S. *Nature-inspired metaheuristic algorithms.* Luniver press, 2010.

[170] Yazdani, H., Kwasnicka, H., Ortiz-Arroyo, D. Multiobjective particle swarm optimization using fuzzy logic. In *Computational Collective Intelligence. Technologies and Applications*. Springer, 2011, p. 224–233.

[171] Yen, G. G., He, Z. Performance metric ensemble for multiobjective evolutionary algorithms. *IEEE Transactions on Evolutionary Computation 18*, 1 (2014), 131–144.

[172] Zapotecas Martínez, S., Coello Coello, C. A. A novel diversification strategy for multi-objective evolutionary algorithms. In *Proceedings of the 12th annual conference companion on Genetic and evolutionary computation* (2010), ACM, p. 2031–2034.

[173] Zeng, F., Decraene, J., Low, M. Y.-H., Hingston, P., Wentong, C., Suiping, Z., Chandramohan, M. Autonomous bee colony optimization for multi-objective function. In *Evolutionary Computation (CEC), 2010 IEEE Congress on* (2010), IEEE, p. 1–8.

[174] Zhang, Q., Li, H. Moea/d: A multiobjective evolutionary algorithm based on decomposition. *Evolutionary Computation, IEEE Transactions on 11*, 6 (2007), 712–731.

[175] Zhao, J., Fernandes, V. B., Jiao, L., Yevseyeva, I., Maulana, A., Li, R., Bäck, T., Emmerich, M. Multiobjective optimization of classifiers by means of 3-d convex hull based evolutionary algorithm. *arXiv preprint arXiv:1412.5710* (2014).

[176] Zitzler, E. Evolutionary algorithms for multiobjective optimization: Methods and applications.

[177] Zitzler, E., Brockhoff, D., Thiele, L. The hypervolume indicator revisited: On the design of pareto-compliant indicators via weighted integration. In *International Conference on Evolutionary Multi-Criterion Optimization* (2007), Springer, p. 862–876.

[178] Zitzler, E., Deb, K., Thiele, L. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary computation 8*, 2 (2000), 173–195.

[179] Zitzler, E., Laumanns, M., Thiele, L., Zitzler, E., Zitzler, E., Thiele, L., Thiele, L. Spea2: Improving the strength pareto evolutionary algorithm, 2001.

[180] Zitzler, E., Thiele, L. Multiobjective optimization using evolutionary algorithms: a comparative case study. In *International Conference on Parallel Problem Solving from Nature* (1998), Springer, p. 292–301.

[181] Zitzler, E., Thiele, L. Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *Evolutionary Computation, IEEE Transactions on 3*, 4 (1999), 257–271.

[182] Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., Da Fonseca, V. G. Performance assessment of multiobjective optimizers: an analysis and review. *IEEE transactions on evolutionary computation 7*, 2 (2003), 117–132.

[183] Zujevs, A., Eiduks, J. New decision maker model for multiobjective optimization interactive methods. *Proceedings of the Information Technologies* (2011), 51–58.

APPENDIX A – Published Results

A.1 Conference Proceedings

- Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. B. Collective preferences in evolutionary multi-objective optimization: Techniques and potential contributions of collective intelligence. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing* (New York, NY, USA, 2015), SAC '15, ACM, p. 133–138
- Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. B. Integrating collective intelligence into evolutionary multi-objective algorithms: Interactive preferences. In *LA-CCI (Latin American) Congress on Computational Intelligence* (2015), LA-CCI
- Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. Collaborative preferences in multi-objective evolutionary algorithms. In *Proceedings of the 12th Brazilian Symposium on Collaborative Systems - Research Paper*, SBSC '15
- Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. B. Using collective intelligence to support multi-objective decisions: collaborative and online preferences. In *IEEE/ACM 30th International Conference on Automated Software Engineering Workshops* (2015), ASEW
- Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. B. Bio-inspired algorithms and preferences for multi-objective problems. In *11th International Conference on Hybrid Artificial Intelligence Systems* (2016), HAIS '16
- Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. Preference-based interactive moea on continuous problem of facility location. In *Workshop de Pesquisa e Desenvolvimento em Inteligência Artificial* (2016)

A.2 Book Chapter

- Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. B. *Evolutionary Multi-Objective System Design: Theory and Applications.* Chapman & Hall/CRC Computer and Information Science Series. CRC Press, 2017, cap. Hybrid Multi-Objective Evolutionary Algorithms with Collective Intelligence

APPENDIX B – Submitted Material for Publication

Submitted materials waiting for approval.

B.1 Conference

- Cinalli, D., Martí, L., Sanchez-Pi, N., Garcia, A. C. Extending collective intelligence evolutionary algorithms: A facility location problem application. In *11th Learning and Intelligent Optimization Conference* (2017), LION '17

APPENDIX C - Multi-Objective Test Problem

This appendix describes the test problems used in the experiments of the thesis: DTLZ [67], ZDT [178] and WFG [94]. .

C.1 The DTLZ Problem Set

The DTLZ problems are part of the Deb–Thiele–Laumann–Zitzler (DTLZ) family of scalable multi-objective test problems to analyse and compare the performance of MOEAs.

C.1.1 DTLZ1

DTLZ1 is an M-objective problem with a linear Pareto-optimal front. The main difficulty of this problem is to converge to the hyper-plane: $\sum_{m=1}^{M} f_m = 0.5$. The fitness landscape contains a large number of local PFs $(11^K - 1)$. A value of $k = 5$ is recommended.

In the case of $M > 3$, 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. The objective functions are expressed as:

$$
\begin{aligned}
f_1(\boldsymbol{x}) &= \tfrac{1}{2}(1 + g(\boldsymbol{x}_M))x_1 x_2 \dots x_{M-1}, \\
f_2(\boldsymbol{x}) &= \tfrac{1}{2}(1 + g(\boldsymbol{x}_M))x_1 x_2 \dots (1 - x_{M-1}), \\
&\vdots \\
f_{M-1}(\boldsymbol{x}) &= \tfrac{1}{2}(1 + g(\boldsymbol{x}_M))x_1(1 - x_2), \\
f_M(\boldsymbol{x}) &= \tfrac{1}{2}(1 + g(\boldsymbol{x}_M))(1 - x_1),
\end{aligned}
$$

having $g(\boldsymbol{x}_M)$ defined as

$$
g(\boldsymbol{x}_M) = 100 \left[|\boldsymbol{x}_M| + \sum_{x_i \in \boldsymbol{x}_M} (x_i - 0.5)^2 - \cos 20\pi(x_i - 0.5) \right],
$$

C.1.2 DTLZ2

This DTLZ2 problem can be used to investigate an MOEA's ability to scale up its performance in a large number of objectives. The test problem has a simple unimodal fitness landscape. In [39], the authors declare that the fitness landscape is too simple to distinguish decision variables of different scales and, therefore, DTLZ2 is not suited for large-scale optimization.

In the case of $M > 3$, 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. The parameter $k = |x_M| = 10$ is suggested and the total number of variables is $n = M + k - 1$. The objective functions are expressed as:

$$
\begin{aligned}
f_1(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \textstyle\prod_{i=1}^{M-1} \cos(x_i \frac{\pi}{2}), \\
&\vdots \\
f_m(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \textstyle\prod_{i=1}^{M-m} \cos(x_i \frac{\pi}{2}) \sin(x_{M-m+1} \frac{\pi}{2}), \\
&\vdots \\
f_M(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \sin(x_1 \frac{\pi}{2})
\end{aligned}
$$

with $g(\boldsymbol{x}_M)$ defined as

$$
g(\boldsymbol{x}_M) = \sum_{x_i \in \boldsymbol{x}_M} (x_i - 0.5)^2 ;
$$

C.1.3 DTLZ3

The DTLZ3 problem is a M-objective problem with a n-dimensional decision vector based on the DTLZ2. The Pareto-optimal front lies on the first orthant of a unit hypersphere. This problem was introduced to test the ability of a MOEA to converge to the global Pareto–optimal front, since there are $3^{n-M+1} - 1$ suboptimal fronts parallel to the optimal one. It is formulated as:

$$
\begin{aligned}
f_1(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \textstyle\prod_{i=1}^{M-1} \cos(x_i \frac{\pi}{2}), \\
&\vdots \\
f_m(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \textstyle\prod_{i=1}^{M-m} \cos(x_i \frac{\pi}{2}) \sin(x_{M-m+1} \frac{\pi}{2}), \\
&\vdots \\
f_M(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \sin(x_1 \frac{\pi}{2})
\end{aligned}
$$

having $g(\boldsymbol{x}_M)$ defined as

$$g(\boldsymbol{x}_M) = 100\left[|\boldsymbol{x}_M| + \sum_{x_i \in \boldsymbol{x}_M} (x_i - 0.5)^2 - \cos 20\pi(x_i - 0.5)\right],$$

where $\boldsymbol{x}_M$ represents the last $n - M + 1$ features of $\boldsymbol{x} \in [0,1]^n$.

C.1.4 DTLZ4

The DTLZ4 modified the DTLZ2 problem with a different meta-variable mapping. It presents a highly nonuniform distribution of the Pareto optimal solutions.

The objective functions are expressed as:

$$\begin{aligned}
f_1(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \textstyle\prod_{i=1}^{M-1} \cos(x_i^\alpha \frac{\pi}{2}),\\
&\vdots\\
f_m(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \textstyle\prod_{i=1}^{M-m} \cos(x_i^\alpha \frac{\pi}{2}) \sin(x_{M-m+1}^\alpha \frac{\pi}{2}),\\
&\vdots\\
f_M(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \sin(x_1^\alpha \frac{\pi}{2})
\end{aligned}$$

having $g(\boldsymbol{x}_M)$ defined as

$$g(\boldsymbol{x}_M) = \sum_{x_i \in \boldsymbol{x}_M} (x_i - 0.5)^2\,;$$

The parameter $\alpha = 100$ is suggested and all variables x_1 to x_{M-1} are varied in $[0,1]$.

C.1.5 DTLZ5

This problem tests the MOEA's ability to converge to a curve. It is recommended to use a higher-objective ($M \in [5,10]$) version of this problem to study the computational time complexity of an MOEA.

It is formulated as:

$$\begin{aligned}
f_1(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \textstyle\prod_{i=1}^{M-1} \cos(\theta_i \frac{\pi}{2}),\\
&\vdots\\
f_m(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \textstyle\prod_{i=1}^{M-m} \cos(\theta_i \frac{\pi}{2}) \sin(\theta_{M-m+1} \frac{\pi}{2}),\\
&\vdots\\
f_M(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \sin(\theta_1 \frac{\pi}{2})
\end{aligned}$$

with $g(\boldsymbol{x}_M)$ defined as

$$g(\boldsymbol{x}_M) = \sum_{x_i \in \boldsymbol{x}_M} (x_i - 0.5)^2 \, ;$$

and $\theta_1, \ldots, \theta_{M-1}$ as

$$\begin{aligned} \theta_1 &= x_1 \tfrac{\pi}{2} \\ \theta_i &= \tfrac{\pi}{4(1+g(\boldsymbol{x}_M))}(1 + 2g(\boldsymbol{x}_M)x_i) \end{aligned}$$

C.1.6 DTLZ6

The DTLZ6 problem is also based on a simpler problem, in this case, the DTLZ5 problem. As in the previous case, suboptimal fronts are also present with the intention of deceiving the optimizer.

The objective functions are expressed as:

$$\begin{aligned} f_1(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \textstyle\prod_{i=1}^{M-1} \cos(\theta_i \frac{\pi}{2}), \\ &\vdots \\ f_m(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \textstyle\prod_{i=1}^{M-m} \cos(\theta_i \frac{\pi}{2}) \sin(\theta_{M-m+1} \frac{\pi}{2}), \\ &\vdots \\ f_M(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \sin(\theta_1 \tfrac{\pi}{2}) \end{aligned}$$

with $g(\boldsymbol{x}_M)$ defined as

$$g(\boldsymbol{x}_M) = \sum_{x_i \in \boldsymbol{x}_M} x_i^{0.1} \, ;$$

and $\theta_1, \ldots, \theta_{M-1}$ as

$$\begin{aligned} \theta_1 &= x_1 \tfrac{\pi}{2} \\ \theta_i &= \tfrac{\pi}{4(1+g(\boldsymbol{x}_M))}(1 + 2g(\boldsymbol{x}_M)x_i) \end{aligned}$$

The Pareto-optimal front corresponds to $x_i = 0$ for $x_i \in \boldsymbol{x}_M$.

C.1.7 DTLZ7

The DTLZ7 problem has a Pareto-optimal front that consists of a heavily disconnected set of 2^{M-1} Pareto-optimal regions. This problem is intended to test an algorithm's ability to maintain a robust coverage of all optimal regions. It is formulated as:

$$\begin{aligned} f_m(\boldsymbol{x}) &= x_m, \text{ for } m = 1, \ldots, M-1; \\ f_M(\boldsymbol{x}) &= (1 + g(\boldsymbol{x}_M)) \left[M - \textstyle\sum_{i=1}^{M-1} \frac{f_i}{1+g(\boldsymbol{x}_M)}(1 + \sin 3\pi f_i) \right] \end{aligned}$$

with g defined as

$$g = 1 + \frac{9}{|\boldsymbol{x}_M|} \sum_{x_i \in \boldsymbol{x}_M} x_i.$$

The Pareto-optimal front corresponds to $x_i = 0$ for $x_i \in \boldsymbol{x}_M$.

C.2 The ZDT Problem Set

The ZDT problems are part of the Zitzler–Deb–Thiele (ZDT) family of scalable multi-objective test problems to analyse and compare the performance of MOEAs.

All the test functions are consist of three main functions: f_1, g, h:

$$\begin{aligned} \text{minimize } & \tau(\boldsymbol{x}) = (f_1(x_1), f_2(\boldsymbol{x})) \\ \text{subject to } & f_2(\boldsymbol{x}) = g\,(x_2, \ldots, x_m)\, h(f_1(x_1), g\,(x_2, \ldots, x_m)) \\ \text{where } & \boldsymbol{x} = (x_1, \ldots, x_m) \end{aligned}$$

The function f_1 assigns the first decision variable only, whereas g uses the remaining $m-1$ variables. The function h receives the function values of f_1 and g as parameters.

C.2.1 ZDT1

The ZDT1 test function has a convex Pareto-optimal front:

$$\begin{aligned} f_1(x_1) &= x_1 \\ g\,(x_2, \ldots, x_m) &= 1 + 9 \sum_{i=2}^{m} x_i/(m-1) \\ h(f_1, g) &= 1 - \sqrt{f_1/g} \end{aligned}$$

where $m = 30$, and $x_i \in [0, 1]$. The Pareto-optimal front is formed with $g(\boldsymbol{x}) = 1$.

C.2.2 ZDT2

The ZDT2 test function is the nonconvex counterpart to ZDT1:

$$
\begin{aligned}
f_1(x_1) &= x_1 \\
g(x_2, \ldots, x_m) &= 1 + 9\sum_{i=2}^{m} x_i/(m-1) \\
h(f_1, g) &= 1 - (f_1/g)^2
\end{aligned}
$$

where $m = 30$, and $x_i \in [0, 1]$. The Pareto-optimal front is formed with $g(\boldsymbol{x}) = 1$.

C.2.3 ZDT3

The ZDT3 consists of several noncontiguous convex parts:

$$
\begin{aligned}
f_1(x_1) &= x_1 \\
g(x_2, \ldots, x_m) &= 1 + 9\sum_{i=2}^{m} x_i/(m-1) \\
h(f_1, g) &= 1 - \sqrt{f_1/g} - (f_1/g)\sin(10\pi f_1)
\end{aligned}
$$

where $m = 30$, and $x_i \in [0, 1]$. The Pareto-optimal front is formed with $g(\boldsymbol{x}) = 1$.

C.2.4 ZDT4

The ZDT4 test function contains 21^9 local Pareto-optimal fronts. This problem was introduced to test the ability of a MOEA to deal with multimodality:

$$
\begin{aligned}
f_1(x_1) &= x_1 \\
g\left(x_2, \ldots, x_m\right) &= 1 + 10(m-1) + \sum_{i=2}^{m} (x_i^2 - 10\cos(4\pi x_i)) \\
h(f_1, g) &= 1 - \sqrt{f_1/g}
\end{aligned}
$$

where $m = 10$, and $x_1 \in [0,1]$ and $x_2, \ldots, x_m \in [-5,5]$. The Pareto-optimal front is formed with $g(\boldsymbol{x}) = 1$.

C.2.5 ZDT6

The ZDT6 test function has a nonconvex Pareto-optimal front:

$$
\begin{aligned}
f_1(x_1) &= 1 - \exp(-4x_1)\sin^6(6\pi x_1) \\
g\left(x_2, \ldots, x_m\right) &= 1 + 9\Big(\big(\sum_{i=2}^{m} x_i\big)/(m-1)\Big)^{0.25} \\
h(f_1, g) &= 1 - (f_1/g)^2
\end{aligned}
$$

where $m = 10$, and $x_i \in [0,1]$. The Pareto-optimal front is formed with $g(\boldsymbol{x}) = 1$.

C.3 The Walking Fish Group Problem Set

The problems to be addressed are part of the Walking Fish Group problem toolkit (WFG). This is a toolkit for creating complex synthetic multi-objective test problems that can be devised to exhibit a given set of target features.

Unlike previous test suites where complexity is embedded in the problem, a test problem designer using the WFG toolkit has access to a series of components to control specific test problem features (e.g., separability, modality, etc.). The WFG toolkit was used to construct a suite of test problems that provides a thorough test for optimizers. This set of nine problems, WFG1 to WFG9, are formulated in such manner that each poses a different type of challenge to multi-objective optimizers.

The WFG test suite exceeds the functionality of previous existing test suites. In particular, it includes a number of problems that exhibit properties not evident in other commonly used test suites such as the DTLZ and the Zitzler–Deb–Thiele (ZDT) test suites. These differences include: non-separable problems, deceptive problems, a truly degenerate problem, a mixed shape Pareto front problem, problems scalable by the number of position-related parameters, and problems with dependencies between position- and distance-related parameters. The WFG test suite provides a better form of assessing the performance of optimization algorithms on a wide range of different problems.

WFG problems are constructed by combining functions that define the shape of the Pareto-optimal front and a set of transformation functions. The shape functions are [119]

$$
\begin{aligned}
\text{linear}_1(x_1,\ldots,x_{M-1}) &= \prod_{i=1}^{M-1} x_i\,;\\
\text{linear}_{m=2,\ldots,M-1}(x_1,\ldots,x_{M-1}) &= \left(\prod_{i=1}^{M-m} x_i\right)(1-x_{M-m+1})\,;\\
\text{linear}_M(x_1,\ldots,x_{M-1}) &= 1-x_1\,;\\
\text{convex}_1(x_1,\ldots,x_{M-1}) &= \prod_{i=1}^{M-1}\left(1-\cos\left(x_i\tfrac{\pi}{2}\right)\right)\,;\\
\text{convex}_{m=2,\ldots,M-1}(x_1,\ldots,x_{M-1}) &= \left[\prod_{i=1}^{M-1}\left(1-\cos\left(x_i\tfrac{\pi}{2}\right)\right)\right]\left(1-\sin\left(x_{M-m+1}\tfrac{\pi}{2}\right)\right)\,;\\
\text{convex}_M(x_1,\ldots,x_{M-1}) &= \left(1-\sin\left(x_{M-m+1}\tfrac{\pi}{2}\right)\right)\,;\\
\text{concave}_1(x_1,\ldots,x_{M-1}) &= \prod_{i=1}^{M-1}\left(1-\sin\left(x_i\tfrac{\pi}{2}\right)\right)\,;\\
\text{concave}_{m=2,\ldots,M-1}(x_1,\ldots,x_{M-1}) &= \left[\prod_{i=1}^{M-1}\left(1-\sin\left(x_i\tfrac{\pi}{2}\right)\right)\right]\left(1-\cos\left(x_{M-m+1}\tfrac{\pi}{2}\right)\right)\,;\\
\text{concave}_M(x_1,\ldots,x_{M-1}) &= \cos\left(x_{M-m+1}\tfrac{\pi}{2}\right)\,;\\
\text{mixed}_M(x_1,\ldots,x_{M-1}) &= \left(1-x_1-\frac{\cos 2A\pi x_1+\pi/2}{2A\pi}\right)^{\alpha}\,;\\
\text{disc}_M(x_1,\ldots,x_{M-1}) &= 1-x_1^{\alpha}\cos^2\left(Ax_1^{\beta}\pi\right)\,.
\end{aligned}
$$

Similarly, the transformation functions are formulated as:

$$
\begin{aligned}
\mathrm{bPoly}(y,\alpha) =& y^{\alpha}\,;\\
\mathrm{bFlat}(y,A,B,C) =& A + \min\left(0, |y-B|\right)\frac{A(B-y)}{B} - \min\left(0,|y-C|\right)\frac{(1-A)(y-C)}{1-C}\,;\\
\mathrm{bParam}(y,u(\boldsymbol{y}'),A,B,C) =& y^{B+(C-B)(A-(1-2u(\boldsymbol{y}'))|\lfloor 0.5-u(\boldsymbol{y}')\rfloor+A|}\,;\\
\mathrm{sLinear}(y,A) =& \frac{|y-A|}{|\lfloor A-y\rfloor + A|}\,;\\
\mathrm{sDecept}(y,A,B,C) =& 1 + (|y-A|-B)\\
& \left(\frac{\lfloor y-A+B\rfloor\left(1-C\frac{A-B}{B}\right)}{A-B} + \frac{\lfloor A+B-y\rfloor\left(1-C\frac{1-A-B}{B}\right)}{1-A-B} + \frac{1}{B}\right)\,;\\
\mathrm{sMulti}(y,A,B,C) =& \frac{1+\cos\left[(4A+2)\,\pi\left(0.5-\frac{|y-C|}{2(\lfloor C-y\rfloor + C)}\right)\right] + 4B\left(\frac{|y-C|}{2(\lfloor C-y\rfloor+C)}\right)}{B+2}\,;\\
\mathrm{rSum}(\boldsymbol{y},\boldsymbol{w}) =& \frac{\sum_{i=1}^{|\boldsymbol{y}|} w_1 y_i}{\sum_{i=1}^{|\boldsymbol{y}|} w_i}\,;\\
\mathrm{rNonSep}(\boldsymbol{y},A) =& \frac{\sum_{j=1}^{|\boldsymbol{y}|}\left(y_j + \sum_{k=0}^{A-2}\left|y_j - y_{1+j+k \mod |\boldsymbol{y}|}\right|\right)}{\frac{|\boldsymbol{y}|}{A}\lceil\frac{A}{2}\rceil\left(1+2A-2\lceil\frac{A}{2}\rceil\right)}\,.
\end{aligned}
$$

There are some other common features for all problems. For example, their decision vector is

$$
\boldsymbol{z} = [z_1,\ldots,z_k,z_{k+1},\ldots,z_n]\,, 0 \le z_i \le z_{i,\max}\,.
$$

and

$$
\begin{aligned}
z_{i=1:n,\max} &= 2i\,;\\
z_{i=1:n,[0,1]} &= \frac{z_i}{z_{i=1:n,\max}}\,;\\
x_{i=1:M-1} &= \max(y_M, A_i)(y_i - 0.5) + 0.5\,;\\
x_M &= y_M\,;\\
S_{m=1:M} &= 2m\,;\\
A_{i=1:M} &= 1\,.
\end{aligned}
$$

C.3.1 WFG1

WFG1 skews the relative significance of different parameters by employing dissimilar weights in its weighted sum reduction. It is separable and unimodal.

$$\begin{aligned} \text{minimize } f_m(\boldsymbol{x}) &= x_M + S_m \text{convex}_m(x_1, \dots, x_{M-1});\ m = 1, \dots, M-1; \\ f_M(\boldsymbol{x}) &= x_M + S_M \text{mixed}_M(x_1, \dots, x_{M-1}),\ \alpha = 1,\ A = 5\,. \end{aligned}$$

where

$$\begin{aligned} y_{i=1:M-1} &= \text{rSum}\left(\left[y'_{(i-1)k/(M-1)+1}, \dots, \quad y'_{ik/(M-1)}\right],\right. \\ &\qquad \left.[2\left((i-1)k(M-1)+1\right), \dots, 2ik/(M-1)]\right)\ ; \\ y_M &= \text{rSum}\left(\left[y'_{k+1}, \dots, y'_n\right], [2(k+1), \dots, 2n]\right)\ ; \\ y'_{i=1:n} &= \text{bPoly}(y''_i, 0.02)\,; \\ y''_{i=1:k} &= y'''_i\,; \\ y''_{i=k+1:n} &= \text{bFlat}(y'''_i, 0.8, 0.75, 0.85)\,; \\ y'''_{i=1:k} &= z_{i,[0,1]}\,; \\ y'''_{i=k+1:n} &= \text{sLinear}(z_{i,[0,1]}, 0.35)\,. \end{aligned}$$

C.3.2 WFG2

This is a non-separable problem with a disconnected Pareto-optimal front.

$$\begin{aligned} \text{minimize } f_m(\boldsymbol{x}) &= x_M + S_m \text{convex}_m(x_1, \dots, x_{M-1});\ m = 1, \dots, M-1; \\ f_M(\boldsymbol{x}) &= x_M + S_M \text{disc}_M(x_1, \dots, x_{M-1}),\ \alpha = \beta = 1,\ A = 5\,. \end{aligned}$$

where

$$\begin{aligned}
y_{i=1:M-1} &= \text{rSum}\left(\left[y'_{(i-1)k/(M-1)+1}, \ldots, y'_{ik/(M-1)}\right], \mathbf{1}\right) \ ; \\
y_M &= \text{rSum}\left(\left[y'_{k+1}, \ldots, y'k + l/2\right], \mathbf{1}\right) \ ; \\
y'_{i=1:k} &= y''_i \ ; \\
y'_{i=k+1:k+l/2} &= \text{rNonSep}\left(\left[y''_{k+2(i-k)-1}, y''_{k+2(i-k)}\right], 2\right) \ ; \\
y''_{i=1:k} &= z_{i,[0,1]} \ ; \\
y''_{i=k+1:n} &= \text{sLinear}(z_{i,[0,1]}, 0.35) \ .
\end{aligned}$$

C.3.3 WFG3

This is a non-separable and unimodal problem [98].

$$\text{minimize } f_m(\boldsymbol{x}) = x_M + S_m(1 - x_{M-m+1})x_1x_2 \ldots x_{M-m}; \ m = 1, \ldots, M-1$$

C.3.4 WFG4

WFG4 is a separable and strongly multi-modal problem that, like the remaining problems, has a concave Pareto-optimal front. This front lies on the first orthant of a hypersphere of radius one located at the origin.

$$\text{minimize } f_m = x_M + S_m \text{concave}_m(x_1, \ldots, x_{M-1}); \ m = 1, \ldots, M-1;$$

where

$$\begin{aligned}
y_{1:M-1} &= \text{rSum}\left(\left[y'_{(i-1)k/(M-1)+1}, \ldots, y'_{ik/(M-1)}\right], \mathbf{1}\right) \ ; \\
y_M &= \text{rSum}\left(\left[y'_{k+1}, \ldots, y'k + l/2\right], \mathbf{1}\right) \ ; \\
y'_{i=1:n} &= \text{sMulti}(z_{i,[0,1]}, 30, 10, 0.35) \ .
\end{aligned}$$

C.3.5 WFG5

WFG5 is also a separable problem but it has a set of deceptive locally optimal fronts. This feature is meant to evaluate the capacity of the optimizers to avoid getting trapped in local optima.

$$\text{minimize } f_m = x_M + S_m \text{concave}_m(x_1, \ldots, x_{M-1});\ m = 1, \ldots, M-1;$$

where

$$\begin{aligned}
y_{1:M-1} &= \text{rSum}\left(\left[y'_{(i-1)k/(M-1)+1}, \ldots, y'_{ik/(M-1)}\right], \mathbf{1}\right)\ ; \\
y_M &= \text{rSum}\left(\left[y'_{k+1}, \ldots, y'k + l/2\right], \mathbf{1}\right)\ ; \\
y'_{i=1:n} &= \text{sDecept}(z_{i,[0,1]}, 0.35, 0.001, 0.05)\,.
\end{aligned}$$

C.3.6 WFG6

WFG6 is a non-separable problem without the strong multi-modality of WFG4 but with a simpler non-separable reduction when compared to WFG2.

$$\text{minimize } f_m = x_M + S_m \text{concave}_m(x_1, \ldots, x_{M-1});\ m = 1, \ldots, M-1;$$

where

$$\begin{aligned}
y_{1:M-1} &= \text{rNonSep}\left(\left[y'_{(i-1)k/(M-1)+1}, \ldots, y'_{ik/(M-1)}\right], k/(M-1)\right)\ ; \\
y_M &= \text{rNonSep}\left(\left[y'_{k+1}, \ldots, y'k + l/2\right], l\right)\ ; \\
y'_{i=1:k} &= z_{i,[0,1]}\ ; \\
y'_{i=k+1:n} &= \text{sLinear}(z_{i,[0,1]}, 0.35)\,.
\end{aligned}$$

C.3.7 WFG7

The WFG7 problem is uni-modal and separable.

$$\text{minimize } f_m = x_M + S_m \text{concave}_m(x_1, \ldots, x_{M-1});\ m = 1, \ldots, M-1;$$

where

$$
\begin{aligned}
y_{1:M-1} &= \text{rSum}\left(\left[y'_{(i-1)k/(M-1)+1}, \ldots, y'_{ik/(M-1)}\right], \mathbf{1}\right) ; \\
y_M &= \text{rSum}\left(\left[y'_{k+1}, \ldots, y'k + l/2\right], \mathbf{1}\right) ; \\
y'_{i=1:k} &= y''_i ; \\
y'_{i=k+1:n} &= \text{sLinear}\left(y''_i, 0.35\right) ; \\
y''_{i=1:k} &= \text{bParam}\left(z_{i,[0,1]}, \text{rSum}\left([z_{i+1,[0,1]}, \ldots, z_{n,[0,1]}], \mathbf{1}\right), 0.98/49.98, 0.02, 50\right) ; \\
y''_{i=k+1:n} &= z_{i,[0,1]} .
\end{aligned}
$$

C.3.8 WFG8

WFG8 is a non-separable problem.

$$\text{minimize } f_m = x_M + S_m \text{concave}_m(x_1, \ldots, x_{M-1});\ m = 1, \ldots, M-1;$$

where

$$
\begin{aligned}
y_{1:M-1} &= \text{rSum}\left(\left[y'_{(i-1)k/(M-1)+1}, \ldots, y'_{ik/(M-1)}\right], \mathbf{1}\right) ; \\
y_M &= \text{rSum}\left(\left[y'_{k+1}, \ldots, y'k + l/2\right], \mathbf{1}\right) ; \\
y'_{i=1:k} &= y''_i ; \\
y'_{i=k+1:n} &= \text{sLinear}\left(y''_i, 0.35\right) ; \\
y''_{i=1:k} &= z_{i,[0,1]} ; \\
y''_{i=k+1:n} &= \text{bParam}\left(z_{i,[0,1]}, \text{rSum}\left([z_{1,[0,1]}, \ldots, z_{i-1,[0,1]}], \mathbf{1}\right), 0.98/49.98, 0.02, 50\right) .
\end{aligned}
$$

C.3.9 WFG9

WFG9 is non-separable, multi-modal and has deceptive local optima. These properties probably make WFG9 the hardest problem of all the problems of the WFG set.

$$\text{minimize } f_m = x_M + S_m \text{concave}_m(x_1, \ldots, x_{M-1});\ m = 1, \ldots, M-1;$$

where

$$
\begin{aligned}
y_{1:M-1} &= \text{rNonSep}\left(\left[y'_{(i-1)k/(M-1)+1}, \ldots, y'_{ik/(M-1)}\right], k/(M-1)\right) \ ; \\
y_M &= \text{rNonSep}\left(\left[y'_{k+1}, \ldots, y'k + l/2\right], l\right) \ ; \\
y'_{i=1:k} &= \text{sDecept}(y''_i, 0.35, 0.001, 0.05) \, ; \\
y'_{i=k+1:n} &= \text{sMulti}\,(y''_i, 30, 95, 0.35) \ ; \\
y''_{i=1:n-1} &= \text{bParam}\left(z_{i,[0,1]}, \text{rSum}\left([z_{i+1,[0,1]}, \ldots, z_{n,[0,1]}], \mathbf{1}\right), 0.98/49.98, 0.02, 50\right) \ ; \\
y''_{i=n} &= z_{n,[0,1]} \, .
\end{aligned}
$$

APPENDIX D – Decision-Making

D.1 Collective Decision-Making Techniques

Besides the current collective scenario, decision-making in large communities is still starting to harvest the so-called “wisdom of the crowds”. It faces difficulties to manage the interactions and get valuable knowledge concealed or dispersed in the group. There are three stages in the formation of groups: sharing information, cooperation of knowledge and collective action. The flow and exchange of information happen openly in the first two stages. Collective decision-making is handled by the last stage and engages all the knowledge generated in the firsts stages.

In general terms, collective decision-making systems work on aggregation engines to combine singular inputs into a global perspective for decision. Many alternatives were elaborated and well studied along the past years [163], but each one of them addresses a particular class of questions. A first class has to do with the representation of a group opinion or preferences and suggests a pool of techniques: voting, recommendation systems, judgement aggregation, averaging, prediction markets, aggregation market techniques, rating scales and ranking. Collecting the best information available is the second category and covers the wikis, document ranking and deliberation maps for arguments. The last category is a compilation of group’s thoughts: brainstorm session and elicitation of ideas.

The first category represents groups opinion. Voting is a distinct approach to aggregate individual opinions [92]. It is the most used method for opinion or preferences aggregation and, also, the most common choice after ideas elicitation. Each participant analyses and ranks the possible alternatives. Then, based on the frequency of votes, an aggregation is made and the winner or a ranking list is produced. It arrives at a social choice from particular preferences. Judge aggregation [26] is a composition of individual votes to develop a collective decision. Those problems examine a group of people declaring their votes (1 or 0, yes or no) on a set X of alternatives.

	P	Q	$(P \wedge Q) \Leftrightarrow R$
Judge A	1	0	0
Judge B	0	1	0
Judge C	1	1	1
Majority	1	1	0

Table D.1: Doctrinal Paradox example [26]. the majority of proposition P and Q independently says R is 1 (true). But, considering the answer of each judge, R is 0 (false).

However, besides the popularity and simplicity to apply these techniques, voting and judge aggregation come up with some problems. The Arrow's paradox, the Condorcet's paradox and the discursive dilemma may result in an inconsistent and undesired collective outcomes. The Arrow's impossibility theorem or Arrow's paradox [12] states that three or more alternatives to a majority-wins strategy do not guarantee a community-wide rank order while also satisfying a set of "fairness criteria". Hence, the result might be a selection that nobody wanted, but have voted for it. Condorcet's paradox [17, 52] demonstrates a situation where the preferences of voters form a cycle in which every preference is overcome by another candidate. Lastly, doctrinal paradox or discursive dilemma [26] is an inconsistent collective result from judge aggregation scenarios.

Table D.1 shows a hypothetical situation to represent the doctrinal paradox. Let P and Q be propositions to satisfy a rule $(P \wedge Q) \Leftrightarrow R$. Each member (judge) must express his judgement on P. According to the table, the majority of proposition P and Q independently says that the conclusion R is 1 (true). But, if the conclusion is taken from the answer of each judge, it yields an ambiguous outcome saying R is 0 (false).

Rating scale is another technique based on the voting strategy. Participants assign numerical values or choose an item from the rating scale to represent two intentions: a vote over that criteria and the intensity of his preference. This mechanism improves the reliability on the final overall ranking. Averaging can be used to achieve a final rate with good results already reported [92, 117]. Some authors criticize the quality of binary rates (thumbs up or down) [32] and suggest a more granular rating scale, like 5-star scale, to get better rankings.

Prediction markets are used to evaluate new products, ideas or preferences. The process of future events estimation can be applied to collect, aggregate and appraise disperse information. Transforming the alternatives in a virtual market place and trade with their probabilities of occurrence or preferences is an interesting plan [133]. Nevertheless, the complexity and game rules of prediction markets may disturb the users' interaction for such a simple action of choosing a couple of items. In this sense, rating scale leads to higher satisfaction and accuracy than prediction market [32].

The second category of decision-making retrieves available information from users' interaction and contributions. Wikis achieve consensus after an edit war and a strong effort to build a neutral point of view. Grenstein developed a statistical approach to measure Wikipedia's neutrality [83]. The deliberatorium from Klein [105] organizes the arguments (pros and cons) of a discussion of the right subject and avoids duplication.

The last category of decision-making discovers and aggregates group thoughts. Eliciting ideas has the mission to involve all the available participants and make their opinions or preferences converge as a consensus. Web sites such as: Dell's Ideastorm.com, MyStarbuckIdea.com, Reddit.com, Obama's government, Google's project 10to100 and others; are internet social communities that register massive levels of activity and ideas. Google's project elicited 150.000 ideas along almost 2 years, Dell's far more than 10.000 and Obama's site around 70.000 in just three weeks [105].

APPENDIX E – Supporting Tools

In this work, the tools used as part of the system architecture are: SQLite, Playcanvas and PythonAnywhere.

E.1 SQLite

SQLite is a relational database management system. This work uses the SQLite database to store all the population and offsprings of the evolution process. Also, the collective contributions and preferences are stored in the database. A full description of the database schema is presented in Figure E.1.

The *gameworld* table control the positions of the obstacles and the boundaries of the scenario. *Experiment* table contains several attributes used in the evolution process: crossover and mutation rate, the algorithm identification, the number of generations before the first interactive interruption, etc. In addition, the population and the collective collaboration are stored in the *generation* table. Users are identified and kept in the *player* table.

E.2 Playcanvas

Playcanvas is an open source 3D WebGL game engine. The application handles rigid-body physics simulation, 3D animations and scripting via JavaScript programming language. It exposes a framework with all the main components to write high-quality games. Figure E.2 shows the Playcanvas editor with the object hierarchy on the left pane and the object properties/attributes on the right pane.

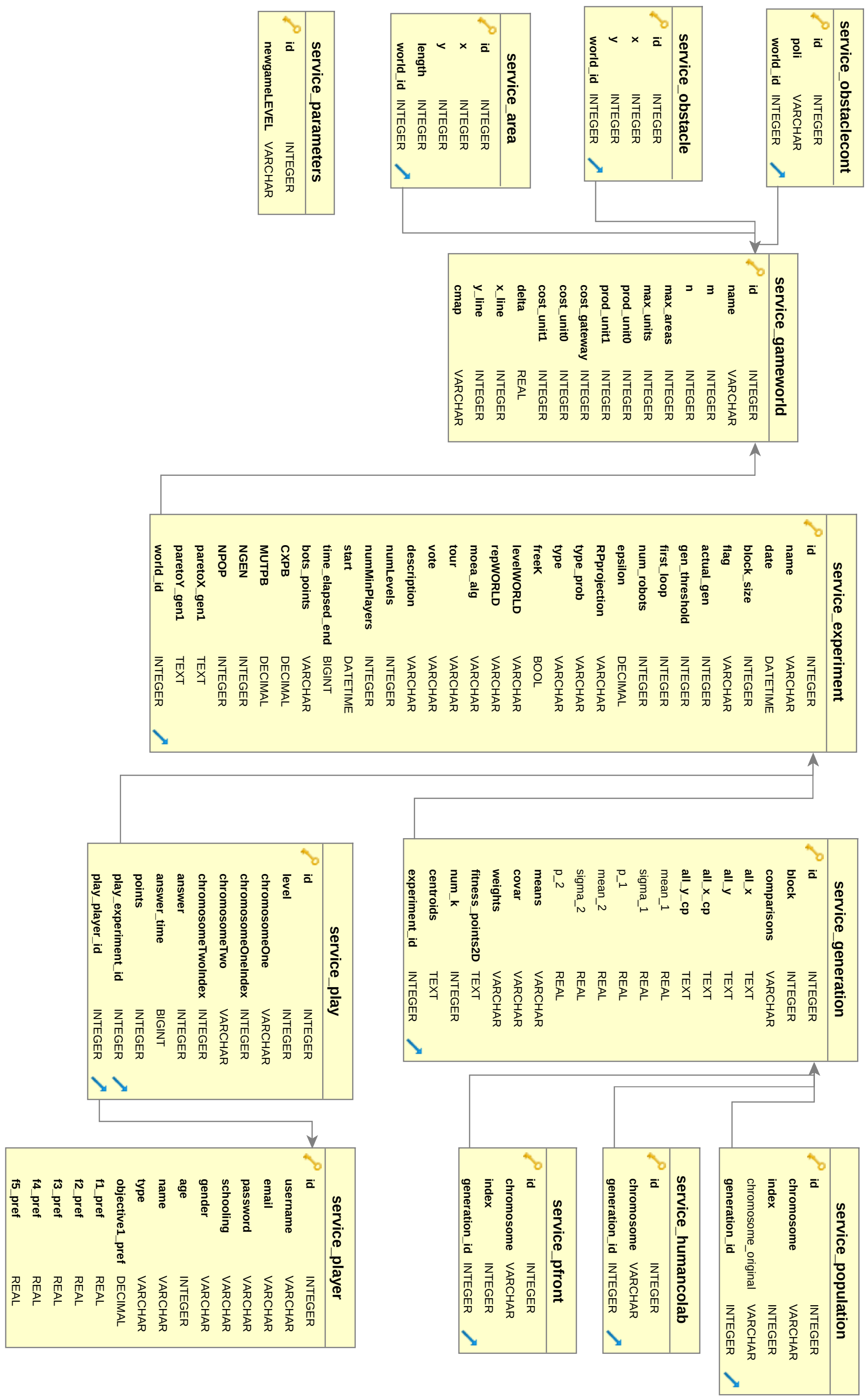


Figure E.1: Database schema of the game.

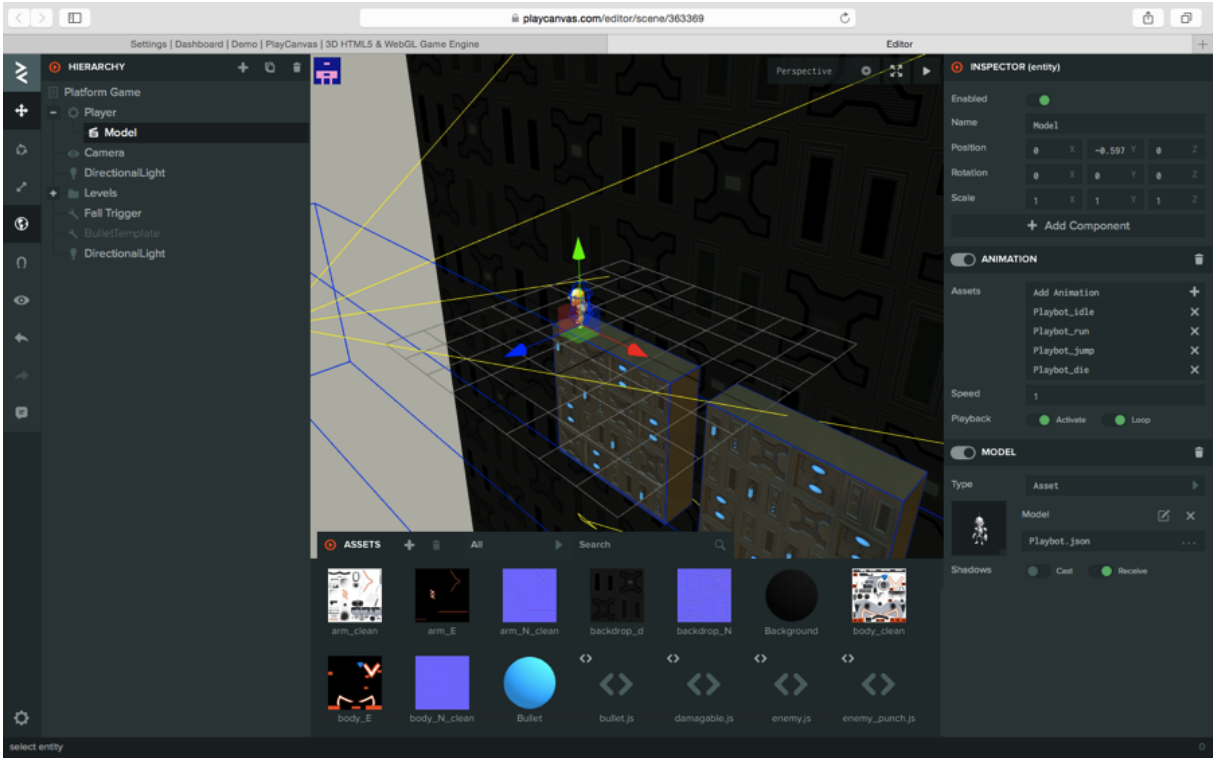


Figure E.2: Playcanvas editor. On the left pane, there is the object hierarchy and, on the right pane, the object properties/attributes.

E.3 PythonAnywhere

PythonAnywhere is a popular and growing internet web server. It provides a development environment and supports many Python versions (2.6, 2.7, 3.3, 3.4 and 3.5). PythonAnywhere is placed on the market as a Platform as a Service (PaaS) and makes it simple for programmers to create, store and run their software in the cloud with easy scalability. Figure E.3 displays some files hosted on the web server. The configuration environment allows to insert, delete or edit any python files.

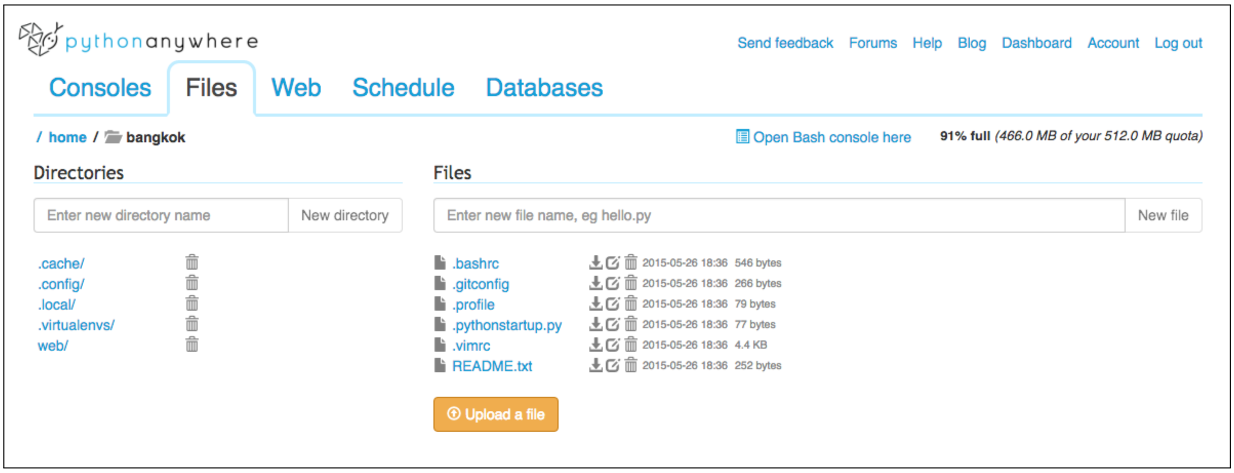


Figure E.3: PythonAnywhere internet web server.

APPENDIX F – Hypothesis Test

F.1 Conover Test

The Conover-Inman procedure [52] is a non-parametric method for testing equality of population medians. It is a test of homogeneity (equal variance) based on ranks and can be implemented in a pairwise manner to determine if the results of one algorithm were significantly better than those of the other. The test does not assume that all populations are normally distributed and is recommended when the normality assumption is not viable [7].

Let g groups each have a normal distribution with possibly different means $\sigma_1, \ldots, \sigma_g$. The number of subjects in each group is: $n_1, \ldots, n_g$, Y_{ki} denote response values and N is the total sample size of all groups. The formula to calculate the Conover test is:

$$T = \frac{1}{D^2}\left[\sum_{k=1}^{g} \frac{S_2^k}{n_k} - N\bar{S}^2\right]$$

where

$$\begin{aligned}
Z_{ki} &= |Y_{ki} - \bar{Y}_k|;\\
R_{ki} &= \mathrm{Rank}(Z_{ki});\\
S_k &= \sum_{i=1}^{n_k} R_{ki}^2;\\
\bar{S} &= \frac{1}{N}\sum_{k=1}^{g} S_k;\\
D^2 = \frac{1}{N-1}\left[\sum_{k=1}^{g}\sum_{i=1}^{n_k} R_{ki}^4 - N\bar{S}^2\right]&.
\end{aligned}$$