THE UNIVERSITY OF WESTERN ONTARIO
DEPARTMENT OF CIVIL AND
ENVIRONMENTAL ENGINEERING

Water Resources Research Report

Multi-objective Evolutionary Algorithms for Water Resources Management

By:

Andre Schardong
and
S. P. Simonovic

Report No: 078
Date: December 2011

ISSN: (print) 1913-3200; (online) 1913-3219;
ISBN: (print) 978-0-7714-2907-1; (online) 978-0-7714-2908-8;

Multi-objective Evolutionary Algorithms for Water Resource Management

By

Andre Schardong
and
Slobodan P. Simonovic

Department of Civil and Environmental Engineering
The University of Western Ontario
London, Ontario, Canada

December, 2011

Executive Summary

This report presents an application of evolutionary algorithms to multi-objective analysis of water resources systems and their integration into a decision support system (DSS). Two DSSs are presented, namely AcquaNet and ModSim that take advantage of two multi-objective evolutionary algorithms: MoDE-NS and MoPSO-NS. Their comparison with NSGA-II is presented.

The algorithms are developed in the form of DSS which enables generalized multi-objective analysis with a focus on water resources systems. The possibilities for integration with AcquaNet and ModSim DSS, either by importing network flow directly from them or by integrated optimization/simulation are also presented. A graphical visualization tool (Trade-off Graph - TG) for easy analysis of the so called non-dominated solutions is included in the DSS.

The algorithms are applied to common set of test problems for validation by comparing their results to the NSGA-II. The possibilities of application of the developed DSS and multi-objective evolutionary algorithms are initially exploited by multi-objective analysis of a hydrological rainfall-runoff model Smap, with two and five objectives. Then, the analysis is extended to a complex water resources system, the Cantareira System, that provides water supply of $33m^3/s$ for nearly half of the Sao Paulo Metropolitan Reion (SPMR). The analysis is done by comparing two pairs of objective functions: minimization of demand deficits versus minimization of pumping cost and minimization of demand deficits versus minimization of the deviation from the water quality standards. The results show that the multi-objective evolutionary algorithms are suitable for application to integrated water resources management and represent a good alternative to the “classical” methods. The MoDE-NS and MoPSO-NS developed in this study, outperformed NSGA-II by obtaining a better coverage of the Pareto

optimal front, especially in the water resources system case study.

Table of Contents

2.4.1 Weighting Method 23

2.4.2 ε-Constraint Method 23

2.6.1 Genetic algorithms - GA 29

2.6.2 Differential Evolution - DE 31

2.6.3 Particle Swarm Optimization - PSO 36

2.7.1 Compromise Programming 40

2.7.2 Graphical Visualization 41

2.8.1 Generational Distance - GD 44

2.8.2 Inverted Generational Distance - iGD 45

2.8.3 Spacing Metric - SP 46

2.8.4 Diversity Metric - ID 47

2.8.5 Dominance Degree - DD 48

2.11.1 AcquaNet DSS 53

2.11.2 ModSim DSS 53

3.2.1 MoDE-NS 59

3.2.2 MoPSO-NS 64

3.2.3 NSGA-II 67

4.1.1 Test 1 – Kursawe's Test 86

4.1.2 Test 2 - Kita's Test 89

4.1.3 Test 3 – ZDT1's Test 93

4.1.4 Test 4 – DTLZ5 96

4.1.5 Test 5 – Viennet 4's Test 99

4.1.6 Test problem results discussion 103

4.2.1 Analysis using two objectives 111

4.2.2 Calibration analysis with five Objectives ... 114

4.3.1 Metropolitan Region of São Paulo Water Supply System ... 119

4.3.2 Multi-objective analysis ... 127

List of Tables

Table 2.1 – Applications of multi-objective evolutionary algorithms in water resources 50

Table 3.1 – MoDE-NS strategies 59

Table 3.2 – MoDE-NS Parameters 63

Table 3.3 – MoPSO-NS parameters 66

Table 3.4 –NSGA-II Parameters 68

Table 3.5 – Objective Function coordinates for 3 Objectives 79

Table 4.1 – Parameters used on MOEAs for Test Problems 85

Table 4.2 – Metrics for Kursawe's test problem 88

Table 4.3 – Metrics for Kita's test problems 92

Table 4.4 – Metrics for ZDT1's test problem 95

Table 4.5 – Metrics for DTLZ5's test problem 99

Table 4.6 – Metrics for Viennet 4's test problem 102

Table 4.7 – Objective Functions Equations used with SMAP 108

Table 4.8 – Parameter values obtained with Objective Function SSR e PWRMSE – Using single objective optimization 110

Table 4.9 – Parameter values obtained for one compromise solution with $\alpha = 0.5$ 113

Table 4.10 – Calibration quality measure from previous solution analysed 113

Table 4.11 – Parameter values obtained for one compromise solution with $\alpha = 0.5$, from analysis with 5 objectives 115

Table 4.12 – Calibration quality measure from previous solution analysed 115

Table 4.13 – Operational data of Cantareira System 119

Table 4.14 – Cantareira system minimum and maximum downstream restrictions flows..... 119

Table 4.15. Water quality model coefficients 123

Table 4.16. Withdraw and loads for each reach 124

Table 4.17 – Algorithms metrics for scenario f_1 vs f_3 129

Table 4.18 – Compromise solution for $\alpha(f_1) = 1$ e $\alpha(f_3) = 0$ 132

Table 4.19 – Compromise solution for $\alpha(f_1) = 0.5$ e $\alpha(f_2) = 0.5$ 135

Table 4.20 – Compromise solution for $\alpha(f_1) = 0$ e $\alpha(f_2) = 1$ 135

Table 4.21 – Solutions with best values for f_2 for both MoDE-NS and AcquaNet+MoDE-NS 140

Table 4.22 – Compromise solutions for three objectives and algorithms with $\alpha = 0,5$ 144

List of Figures

Figure 2.1 – Typical Pareto front for two conflicting objectives (minimization functions) 18

Figure 2.2 – Pareto fronts examples - Source: Deb (2009) 18

Figure 2.3 – Dominance relation in solution space for a two-objective problem 20

Figure 2.4 – Example of a set of solutions found very close to the real Pareto Front 21

Figure 2.5 – Non-dominated solution ranking schema 27

Figure 2.6 – *Crowding Distance* calculation schema 28

Figure 2.7 – *Fitness Sharing calculation schema* 28

Figure 2.8 – General D.E. Flowchart 34

Figure 2.9 – DE’s pseudo code 35

Figure 2.10 – Flowchart of a PSO based algorithm 38

Figure 2.11 – Bar chart example – Source: Deb (2009) 42

Figure 2.12 – Line Chart – Source: Deb (2009) 42

Figure 2.13 –*Star Coordinate* method 43

Figure 2.14 – Generational Distance – GD metric schema 45

Figure 2.15 – Inverted Generational Distance – iGD schema 46

Figure 2.16 – Space metric – SP schema 47

Figure 2.17 - Diversity Metric – ID schema......48

Figure 2.18 – Dominance Degree – DD schema......49

Figure 2.19 – Typical structure of a Decision Support System, Source: Porto et al. (2003) ...53

Figure 3.1 – Non-dominated solution ranking schema......55

Figure 3.2 – *Crowding Distance* calculation schema......56

Figure 3.3 – Procedure for MOEA developed......58

Figure 3.4 – Pseudo code of MOEA developed......59

Figure 3.5 – Offspring population generation schema for MoDE-NS......63

Figure 3.6 – MoDE-NS parameter configuration screen......64

Figure 3.7 – MoPSO-NS Flowchart......66

Figure 3.8 –MoPSO-NS dos Parameters configuration......67

Figure 3.9 –NSGA-II Chart flow......68

Figure 3.10 –Parameters do NSGA-II configuration Screen......69

Figure 3.11 – DSS Flowchart......72

Figure 3.12 – DSS – Main Screen......73

Figure 3.13 – Problem Setup......73

Figure 3.14 – Data setup for multi-objective analysis......74

Figure 3.15 – Algorithms configuration......75

Figure 3.16 – Objective Function for problem with tow objectives......76

Figure 3.17 – Objective function definition......76

Figure 3.18 – Optimization progress......77

Figure 3.19 –Results visualization......77

Figure 3.20 – Schema for circle used to project non-dominated solutions......79

Figure 3.21 – Schema for centroid calculation for a non-dominated solution......81

Figure 3.22 – Example of Trade-off Graph (TG) usage......82

Figure 3.23 – Result screen of the developed DSS 82
Figure 3.24 – Flowchart of Integration between the DSS and AcquaNet / ModSim DSS....... 84
Figure 4.1 – Result for Kursawe’s Test – NSGA-II 87
Figure 4.2 – Result for Kursawe’s Test – MoDE-NS 88
Figure 4.3 – Result for Kursawe’s Test – MoPSO-NS 88
Figure 4.4 – Non-dominated solutions for Kita’s test problem – MoDE-NS 91
Figure 4.5 – Non-dominated solutions for Kita’s test problem – MoPSO-NS 91
Figure 4.6 – Non-dominated solutions for Kita’s test problem – NSGA-II 92
Figure 4.7 – Non-dominated solutions for ZDT1’s test problem – MoDE-NS 94
Figure 4.8 – Non-dominated solutions for ZDT1’s test problem – MoPSO-NS 94
Figure 4.9 – Non-dominated solutions for ZDT1’s test problem – NSGA-II 95
Figure 4.10 – Non-dominated solutions for DTLZ5’s test problem – MoDE-NS 97
Figure 4.11 – Non-dominated solutions for DTLZ5’s test problem – MoPSO-NS 98
Figure 4.12 – Non-dominated solutions for DTLZ5’s test problem – NSGA-II 98
Figure 4.13 – Result do Teste de Viennet 4 – MoDE-NS 101
Figure 4.14 – Non-dominated solutions for Viennet 4’s Test – MoPSO-NS 102
Figure 4.15 – Non-dominated solutions for Viennet 4’s Test – NSGA-II 102
Figure 4.16 – TG graph for Viennet 4 test non-dominated solutions – MoDE-NS 103
Figure 4.17 – User Interface of Smap.Net Model 105
Figure 4.18 – Smap Model Schema - Source: Lopes et al. (1982) 105
Figure 4.19 –Piriqui River basin location in Parana State - Brazil 110
Figure 4.20 – Modeled and calculated flow for Objective Functions SSR and PWRMSE ... 111
Figure 4.21 – Non-dominated solutions for two Objectives: SSR e PWRMSE 112
Figure 4.22 – Modeled and calculated flow for compromise solution selected, with α = 0.5 for all objectives (MoDE-NS) 114

Figure 4.23 – Modeled and calculated flow for compromise solution selected with com α = 0.5 for all objectives (MoDE-NS) ... 116

Figure 4.24 – Trade-off Graph for MoDE-NS with five Objective, and α = 0.5 (for all objectives) ... 117

Figure 4.25 – Trade-off Graph for MoPSO-NS with five Objective, and α = 0.5 (for all objectives) ... 117

Figure 4.26 – Trade-off Graph for NSGA-II with five Objective, and α = 0.5 (for all objectives) ... 118

Figure 4.27 – Cantareira System location and main components. Source: Sabesp (2009a) .. 120

Figure 4.28 – Schematic for Cantareira System Water Supply ... 121

Figure 4.29 – True Pareto front for scenario f_1 *vs* f_3 – for 24 months simulation period ... 128

Figure 4.30 – Non-dominated solutions for scenario f_1 *vs* f_3 – MoDE-NS ... 130

Figure 4.31 – Non-dominated solutions for scenario f_1 *vs* f_3 – MoPSO-NS ... 131

Figure 4.32 – Non-dominated solutions for scenario f_1 *vs* f_3 – NSGA-II ... 131

Figure 4.33 – Non-dominated solutions for scenario f_1 *vs* f_3 for three algorithms ... 132

Figure 4.34 – Monthly flow provided to SPMR ... 133

Figure 4.35 – Non-dominated solutions for scenario f_1 vs f_2 for three algorithms ... 134

Figure 4.36 – 1. Value of BOD at N3 for Best Compromise solution with $\alpha(f_1) = 0.5$ and $\alpha(f_2) = 0.5$ (Comp. Sol.); 2. Solutions with best value of BOD - $\alpha(f_1) = 0$ e $\alpha(f_2) = 1$ (Best value for BOD) ... 135

Figure 4.37 – Flowchart for M.O. analysis using MOEAs + AcquaNet DSS ... 137

Figure 4.38 –AcquaNet DSS networkflow representation ... 138

Figure 4.39 – Non-dominated solutions for scenario f_1 vs f_2, MoDE-NS and AcquaNet+MoDE-NS ... 140

Figure 4.40 – Non-dominated solutions for three different BOD scenarios for downstream flow {1, 3, 5} mg/L 141

Figure 4.41 – 3D graph for non-dominates solutions for three objectives using MoDE-NS . 144

Figure 4.42 – Trade-off graph for three objectives using MoDE-NS with α = 0,5 144

1 Introduction

1.1 Background

The inadequate availability of water in quantity and quality to meet the various demands is undoubtedly one of the most important issues to be addressed in the water resource management in a watershed. It is often necessary to build dams to regulate flows to ensure the human water supply, irrigation, hydropower generation, environmental control, water quality control, recreation and other demands. In the hydrological system, reservoirs are usually physically connected or act as components of an integrated system.

The availability of water is discussed globally and there is a consensus that water is a scarce good and endowed with economic value. Such compliance must be present at all stages of planning, since conflicts of interest regarding the use of water demonstrate the need for inter-institutional coordination and the adoption of integrated water resources management policies.

The Brazilian Federal Law No. 9.433/97, establishing the National Water Resources Policy (NWRP) and created the National Water Resources Management System, which contributes significantly to the water resource regulating and planning process. This law establishes that the NWRP should be based on the following points:

• Water is a public good;

• Water is a limited natural resource with economic value;

• In situations of scarcity, the priority of water use is for human consumption and watering animals;

• The management of water resources should always consider multi-objective analysis;

• The watershed is the basic unit for implementation of the National Water Resources and activities of the National Water Resources Management;

• The water resources management should be decentralized and rely on the participation of government and society.

The fundamentals mentioned naturally require that the water management should be focused on the multiple uses of water. Among the various management activities, complex problems exist and arise, as example, the operation of water systems for water allocation. Generally the water allocation for meeting demands is subject to availability, to operational constraints and conflicts among various uses.

These problems are inherently of multi-objective nature, being necessary to consider, in addition to operational constraints, a number of other restrictions of any nature: economic, cultural, technological, among others, which may be difficult to deal with using single-objective optimization techniques. According to Loucks and Beek (2005), the traditional approach to cost / benefit, whose objectives are converted and expressed in a single unit, can be extremely difficult and lead to distortions in the set of optimal solutions.

In single-objective optimization techniques, the solution algorithms aim to find a single global optimum value within the search space defined by the constraints of the problem. The multi-objective optimization techniques ideally generate a set of non-dominated solutions, and for this reason the concept of global optimum no longer makes sense. In general this set of solutions is known as the set of non-dominated solutions, or non-inferior or Pareto front. A solution is said to be dominated when it cannot be improved without worsen another solution of the solutions found so far.

Traditional multi-objective analysis techniques use mathematical programming to generate the set of non-dominated solutions. For this purpose, several methods exist where the multi-objective problem is transformed into a single objective problem and solved successively until the non-dominated set of solutions is determined. Among them, the weighting method and e-Constraint method are some of the techniques used for multi-objective analysis that make use of mathematical programming (either linear or nonlinear) to find the set of non-dominated solutions. These methods are called classics here so they can be differentiated from Evolutionary Algorithms, discussed in this report.

Evolutionary algorithms raised in the 80s initially with the Genetic Algorithm (GA) as well as its first multi-purpose version, Deb (2009). Authors like Simonovic (2009), Deb (2009) highlight some characteristics that make the evolutionary algorithms very attractive to multi-objective analysis in relation to the classical methods:

- Based on populations, and for this reason are able to generate multiple non-dominated solutions in a single generation;
- Are less susceptible to the shape or continuity of the Pareto front.
- Extremely flexible regarding problem formulation (objective functions and constraints);
- Do not require deep knowledge of optimization models, and are generally easy to implement;

Great attention has been given to the evolutionary algorithms in the last decade, and the raise of alternative algorithms for GA, many researchers have been developing more efficient algorithms for multi-objective analysis. Among them are DE (Differential Evolution) and PSO (Particle Swarm Optimization), whose versions for multi-objective analysis are the focus of this report, as well as its application in multi-objective analysis of water resources systems.

The multi-objective evolutionary algorithms (MOEA) developed, MoDE-NS: Multi-Objective Differential Evolution and MoPSO-NS: Multi-objective particle swarm optimization, are compared with a version of multi-objective GA, called Non-dominated Sorting Genetic Algorithm - NSGA-II for the validation and result benchmark purpose. The three algorithms were summarized in a Decision Support Systems (DSS) that consists of a user interface (UI), and a library of algorithms, that can be used separately from the UI and applied to solve any type of Multi-objective problem. NSGA-II was translated into the same programming language used in the development of this DSS, but all its original features were preserved.

The developed Decision Support System is a very flexible tool that allows multi-objective analysis of problems across the board, without additional tools and can also be integrated to other DSS for water resource management as AcquaNet and ModSim DSS. AcquaNet (Porto et al., 2003 and 2005) and ModSim (Labadie, 2010), that can solve efficiently large problems using networkflow algorithms. Moreover, the developed DSS provides a tool for visualization of the set of non-dominated solutions based on a methodology proposed by Baltar (2007). This tool is called Trade-off Graph (TG) and uses a coordinate system method to project non-dominated solutions on a 2D graph and compromise programming to highlight the best (or most robust) solutions based on user preferences. The TG is described in the methodology and examples are presented in Section 3.

The integration of evolutionary algorithms discussed and Decision Support System – DSS developed is a tool that can help decision makers and watershed comities to resolve conflicts and assist in the management and planning of water resources systems. The possibility of integrating the DSS with other systems as AcquaNet and ModSim DSS enables multi-objective analysis using the MOEAs to complex and large water resources problems.

1.2 Organization of the Repot

The report is organized into five chapters in this sequence: (1) introduction and objectives, (2) literature review, (3) methodology, (4) application of the methodology, followed by (5) conclusions and recommendations. On Annex 1 contains instructions for installation of the Decision Support System developed and described in this report.

2 Literature Review

Optimization techniques have been employed in recent decades to deal with water resource management problems. Optimization models are based on mathematical programming techniques, which include a variety of algorithms, whose choice depends on the characteristics of the problem to be analyzed, the data availability, the objectives and constraints. Authors such as Yeh (1985) and Labadie (2004) present a detailed review of the most common techniques employed in the management of water resources systems.

Rarely the management of complex water resources systems involves a single objective and therefore, the multi-objective analysis is recommended (Simonovic, 2009). On single-objective optimization techniques, the solution algorithms aim to find a single optimal value (ideally the global optima), within the search space defined by the constraints of the problem. Since multi-objective techniques ideally generate a set of non-dominated solutions, and for this reason the concept of global optimal does not make sense. In fact, it is common in multi-objective analysis that the term “optimization” is absent, since there is no unique “optimal” solution for the problem (Deb, 2009 and Simonovic, 2009).

Traditional techniques of multi-objective analysis use mathematical programming to generate the set of non-dominated solutions. There are several methods in which the multi-objective problem is decomposed into a single objective problem and solved successively until the non-dominated set of solutions is found (Baltar, 2007 and Deb, 2009). Among them, the weighting method and ε-constraint method are the most commonly used techniques use for multi-objective analysis in case of non-linear problems.

With the rise of evolutionary algorithms (EA) in the 80's and their multi-objective versions (MOEA), new analytical methods have been proposed multi-objective. Initially, the

MOEA were derived from the genetic algorithm (GA) (Deb, 2008 and 2009). But with the rise of new algorithms, as alternatives to GA, like DE (Differential Evolution) and PSO (Particle Swarm Optimization), many researchers have been developing more efficient algorithms for multi-objective analysis.

This session presents the basic concepts related to multi-objective analysis, a quick review of two very popular classic methods (the weigh, the basic structure for GA, DE, PSO and some versions of the same multi-objective. Visualization and presentation of non-dominated solutions are also discussed. Finally, a review of some applications of MOEA in analysis of water resource systems is presented.

2.1 Basic Concepts

The process of modeling and solving problems with two or more conflicting and non-measurable goals is called Multi-objective analysis. The multi-objective analysis aims to produce a set of optimal solutions that represent the best trade-off relations between objectives which are called the set of non-dominated solutions or simply, the Pareto front.

The so called trade-offs are inevitable when it comes to multi-objective analysis, which arise precisely because there are conflicting among different uses in a water resources system, and thus objectives cannot be simultaneously fully met.

Figure 2.1 presents in graphical form a typical relationship between two conflicting objectives, where f_1 and f_2, are both minimization functions. This graph also presents the concept non-dominated solutions and Pareto front ($\mathcal{F}$), that are formally presented below. In Figure 2.2 other examples of Pareto fronts are presented.

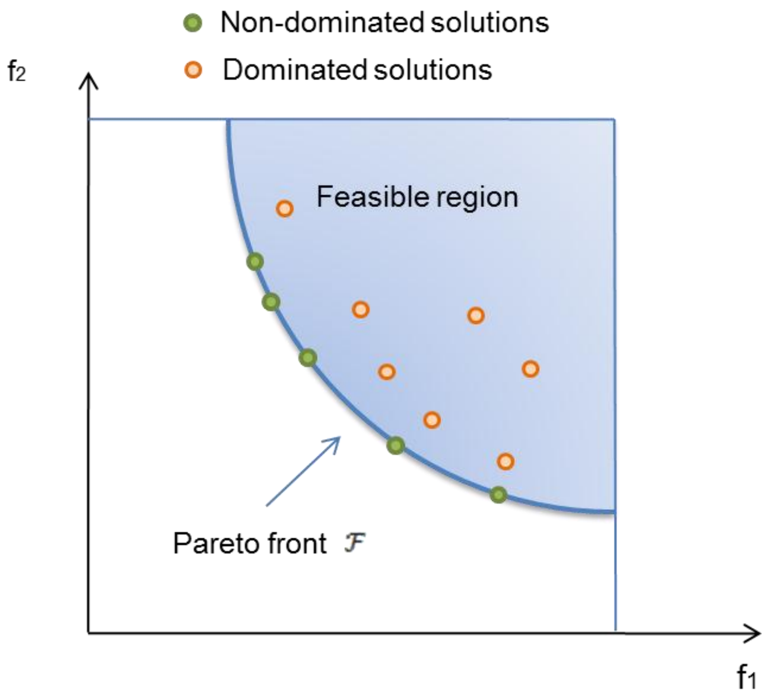


Figure 2.1 – Typical Pareto front for two conflicting objectives (minimization functions)

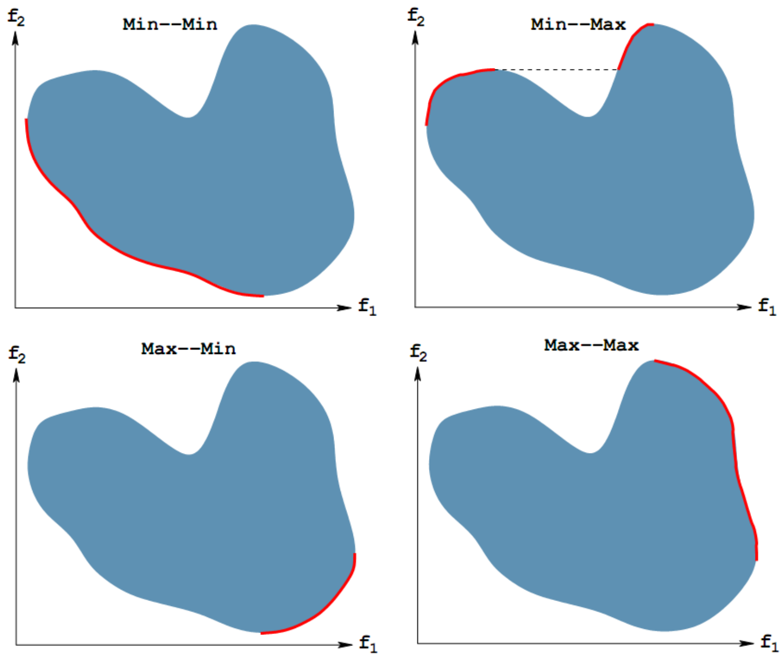


Figure 2.2 – Pareto fronts examples - Source: Deb (2009)

A multi-objective problem can be written as:

$$Minimize\ f(\vec{x}) = [\ f_1(\vec{x}),\ f_2(\vec{x}), \dots,\ f_k(\vec{x})] \quad with\ k = 1, \dots, M$$
$$Subject\ to \quad g_i(\vec{x}) \leq 0 \qquad with\ i = 1, \dots, n \tag{2.1}$$
$$x_j \geq 0 \qquad with\ j = 1, \dots, p$$

Where vector $\vec{x}^* = [x_1^*, \dots, x_n^*]$ that satisfies the equation (2.1), and is optimal solution for the optimization problem, $g_i(\vec{x})$ are the problem constrains, M the number of objectives, n the number of constrains and p the number of the problem variables. The constraints in eq. (2.1) define the feasible space **Ω** of the problem. Constraints of type ≤ can be converted in constraints of type ≥ by multiplying by -1. The same procedure can be used to convert a Maximization problem into a Minimization problem.

2.2 Non-dominance concept

Most multi-objective algorithms use the non-dominance concept in the optimization process. In these algorithms, one of the basic processes is the comparison (for sorting procedure, selection, and others) of two solutions or individuals for their dominance.

One solution is said to be non-dominate over other, if two conditions are met:

1 – The solution $\vec{x}$ is no worse than $\vec{y}$ in any objectives, that is $f_j(\vec{x}) \ntrianglerighteq f_j(\vec{y})$, for j ∈ {1, 2, ..., M}, where M is the number of objectives.

2 – The solution $\vec{x}$ is slitly better than $\vec{y}$ in at least one of the objectives, $f_j(\vec{x}) \triangleleft f_j(\vec{y})$, for at least one j ∈ {1, 2, ..., M} where M is the number of objectives.

If both conditions above are satisfied then $\vec{x}$ dominates $\vec{y}$, that also written as $\vec{x} \preccurlyeq \vec{y}$. If one of conditions 1 and 2 is not met, $\vec{x}$ do not dominates $\vec{y}$.

The non-dominated solutions define the optimal set of solutions or the Pareto-optimal Front $\mathcal{F}$. Thus, and individual or solution $\vec{x}$ belongs to $\mathcal{F}$ only if no other solutions $\vec{y}$ exists, that dominates $\vec{x}$.

The Figure 2.3 shows an example of non-dominance solutions, in which solutions 1 and 2 are non-dominated and 3, 4, 5 and 6 are dominated solutions.

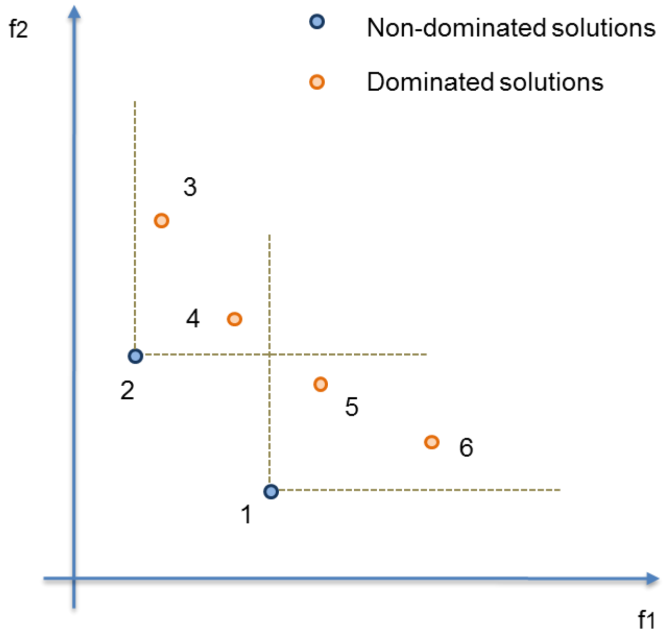


Figure 2.3 – Dominance relation in solution space for a two-objective problem

The set of non-dominated solutions is conceptually equivalent in multi-objective analysis, to the global optimal in single-objective optimization (Simonovic, 2009).

In multi-objective analysis, two important goals must be achieved (regardless of which technique or algorithm is used):

(i) finding solutions as close as possible to the true Pareto-optimal front; and

(ii) finding solutions that reasonably cover the whole extension of the Pareto-optimal front

The ideal situation is shown in the example in Figure 2.4 where both goals are met satisfactorily, i.e., the applied technique found solutions very close to the real Pareto front and properly distributed to cover the whole extension of the front.

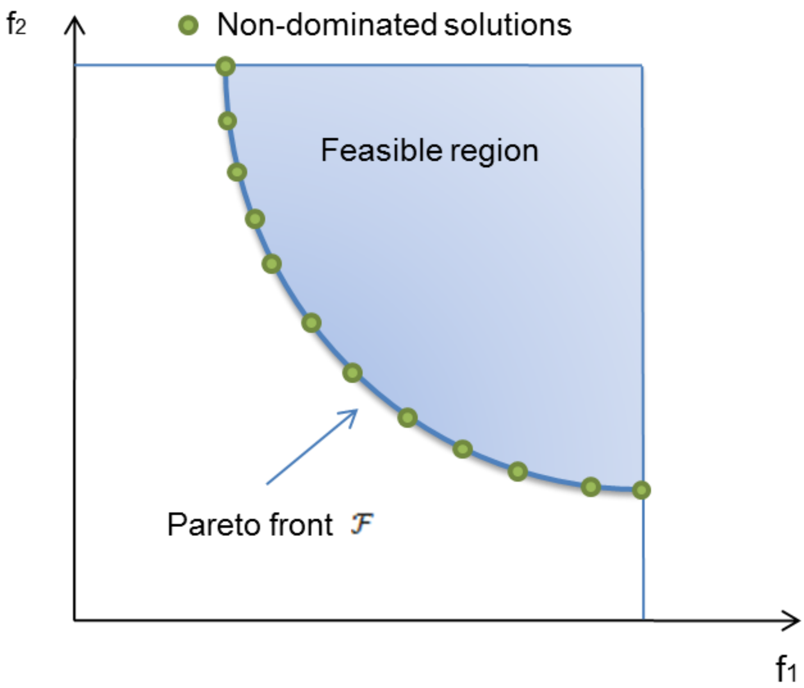


Figure 2.4 – Example of a set of solutions found very close to the real Pareto Front

2.3 Classification of Multi-Objective techniques

The techniques for multi-objective analysis can be classified into two major groups (Deb, 2009):

(i) generating techniques of non-dominated set of solutions: methods in this category solve the multi-objective problem by trying to find the whole set of non-dominated solutions without any intervention from the user or Decision Maker. DM's intervention is required, by introducing its preferences on a second step, to identify best compromise solutions. These techniques are classified by Deb (2009) as the ideal approach to solver multi-objective analysis problems.

(ii) techniques with prior articulation of preferences: methods in this category requires a prior definition of DM's preferences to convert the multiple objective problem into a single objective one, which then can be solved using a single-objective optimizer.

It is also very common classification using a third category of techniques which rely on progressive articulation of preferences (Simonovic, 2009). These methods start trying to find a set of non-dominated solutions that is presented to the user or DM. The problem is modified to reflect DM’s preferences and solved again. These two steps are repeated until a satisfactory solution is obtained.

The evolutionary algorithms are classified as the first category as well as some classical methods as the weighting method and ε-Constraint method that try to find the complete set of non-dominated solutions. Deb (2009) refers to the generating methods as the ideal approach since they provide more effective and comprehensive decision support information. Baltar (2007) highlights that a set of solutions with their respective trade-offs could provide more valuable information to DM than a single optimal solution.

2.4 Classical Methods

This section will describe two mathematical programming methods commonly used in solving multi-objective problems. These are called “classics” to differentiate them from evolutionary algorithms presented in this report.

An important feature of these methods is that they to multiple-objective problem in a single-objective one, enabling thus the use of mathematical programming techniques to solve it.

This report doesn’t aim to detail these classical methods, only briefly review them. Interested readers should refer to a series of author that present a very comprehensive and detailed review on these methods: Loucks and Beek (2005), Baltar (2007), Coello Coello (2007), Deb (2009) and Simonovic (2009).

2.4.1 Weighting Method

This method aims to obtain the set of non-dominated solutions by parametric variation of weights (w_i) applied to objective functions of the problem, as presented in equation (2.2). The weights convert the multi-objective problem into single-objective that can be solved using mathematical programming techniques (linear or nonlinear programming).

$$Minimize\ f(x) = \sum_{i=1}^{M} w_i \times f_i(x),\ \text{where } M \text{ is the number of objectives}$$
$$Subjet\ to: x \in X \tag{2.2}$$

These weights w_i must be supplied to the solution algorithm and its choice can directly affect the results, i.e., not all non-dominated solutions can be obtained with this method.

Deb (2009) and Simonovic (2009) further highlighted the need for parametric variation of weights can result in large computational effort, since for each set of weights is necessary to solve the problem and hopefully find a new non-dominated solution. Another problem with this method is that different sets of weight my result in the same solution.

2.4.2 ε-Constraint Method

This method also belongs to the first category generating of techniques for, and as one of the objectives is optimized, the others are converted into constraints of the problem, according to equation (2.3). This is how the multi-objective problem is converted into a single-objective problem to be solved with the use of mathematical programming.

$$Minimize\ f(x) = f_i(x)$$
$$Subjet\ to\ \ x\ \in X\ e\ f_i(x) \leq \varepsilon_j\ \ for\ \ \ i\ \neq j,\ \ i, \qquad (2.3)$$
$$j\ \in \{1,2,\dots,M\}, where\ \ M\ is\ the\ number\ of\ objectives$$

The challenge in this method is to find or set correctly the ε_j values for each of the functions within the maximum and minimum boundaries, thus ensuring that the set of non-dominated solutions is found.

2.5 Why Evolutionary Algorithms?

The classical methods for multiple-objective analysis use a preference based schemes, which ultimately convert the multi-objective problem to single-objective so that can be solved by using mathematical programming. These classical methods generate “point to point” solutions, i.e., at each simulation they are able to produce only one solution, hopefully a new non-dominated solution of the problem, (Deb, 2008).

The main difference between MOEA and the classical methods is that EA are population based. This feature makes them unique to solve multi-objective problems, being able to produce multiple non-dominated solutions in a single generation.

Coello Coello (2007 and 2010), Simonovic (2009), Deb (2008 and 2009) highlight some features that make evolutionary algorithms (EA) very attractive for multi-objective analysis in relation to so-called classical methods. Among them:

- Are able to generate multiple non-dominated solutions in a single generation;
- Are less susceptible to the shape or continuity of the Pareto front.
- Extremely flexible regarding problem formulation, with regard to the objective

functions and constraints;

- Do not require deep knowledge of nonlinear optimization models, and are generally easy to implement.
- Are not influenced by the initial solution, unlike many methods for non-linear mathematical programming, especially those based on derivatives and gradients.

According to Deb (2008 and 2009), Coello Coello et al. (2007) and Price et al. (2005), the use of evolutionary algorithms for multi-objective analysis has grown significantly in recent years, and have been quite effective in a wide variety of optimization problems.

2.6 Multi-objective Evolutionary algorithms - MOEA

Multi-objective Evolutionary algorithms - MOEAs are able to optimize problems that would be difficult to be solved using conventional techniques such as linear and nonlinear programming.

MOEAs initiates with a population of solutions usually randomly generated within the limits imposed on variables (lower and upper limits). From the initial population, the off-spring populations are generated using selection, mutation and crossover operators. To preserve the best individuals, an elitism operator is also commonly used (Deb, 2008).

The MOEA raised in the 80's, initially with a version of Genetic Algorithm (GA), also known as "Vector Evaluated Genetic Algorithm" or VEGA, proposed by Schaffer (1984) apud Deb (2009). According to Deb (2008) little attention was given to this algorithm, as it used to converge prematurely to specific solutions and could not maintain a desired diversity of the non-dominated set of solutions, which is the second goal to be reached in multi-objective analysis, as discussed earlier. Also according to the author, MOEAs have gained more attention after the publication of Goldberg's (1989) work, where he suggested a revolutionary procedure using non-dominance concept (Non-dominated Sorting – NS procedure) for generation of offspring solutions. In this work the author also suggest a selection operator to maintain diversity of solutions, using information from the vector solution or directly from the values of the objective function. These two operators are the basis of all modern MOEAs.

From this work many algorithms for multi-objective analysis have been proposed in the 90's, initially based on GA. An example is the NSGA-II algorithm, that is also widespread in the scientific community (which can be noticed by the large number of published papers in which it is used), and its results and application serve as a benchmark for new algorithms proposed or developed.

With the raise of other evolutionary algorithms such as Differential Evolution (DE) originally proposed by Storn and Price (1995) and Price and Storn (1997) and Particle Swarm Optimization (PSO) proposed by Kennedy and Eberhart (1995), multi-objective algorithms based on both have been proposed.

By extending an EA, either GA, PSO or DE for multi-objective analysis is necessary to consider two important factors in the MOEA (Chakraborty et al. 2008):

1. how to maintain and select the best individuals, i.e., how to consider elitism, and
2. how to maintain the diversity of the population.

Elitism is generally considered the classification of solutions by its non-dominance, as suggested by Goldberg. This method has variations, but its basic procedure consists on the classification of solutions by assigning each one a rank according to their “dominance: the first non-dominated solutions receive Rank 1 and are removed from the population. Once these solutions are removed the new non-dominated solutions are sought and receive a rank 2. This new solutions are removed and this process is repeated until the entire population is classified as in Figure 2.5.

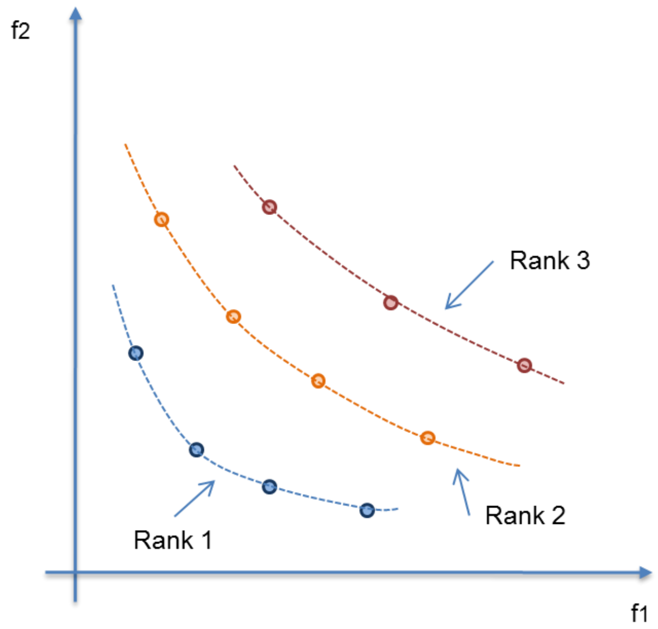


Figure 2.5 – Non-dominated solution ranking schema

To maintain the diversity two common operators are used: Crowding Distance (CD) and Fitness Sharing (FS). Both are used as a selection criteria of non-dominated solutions for the generation of child population, and its main goal is to keep solutions "well" spread as possible on the Pareto front. Both are a measure of how densely the neighborhood of a solution is populated.

CD is calculated using the objective function value of neighboring solutions, as Figure 2.6 and is an indicator of how far apart they are. Thus, solutions with higher values of CD are preferred as they are in a less densely populated region. The great advantage of using CD over FS is that no additional parameter is required.

The FS calculates the density of the solutions into a circle of radius σ_{share} around each solution, as Figure 2.7. Unlike the CD, the lower the value of FS, the more likely the solution is to be chosen because it will be in a less densely populated region. The disadvantage of this metric is that the value of σ_{share} must be defined and can affect significantly the result of optimization, Deb (2009).

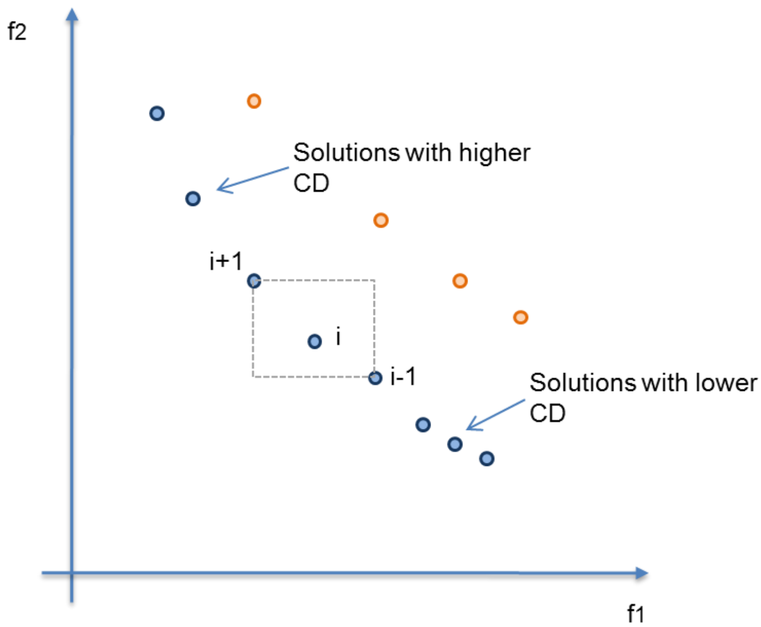


Figure 2.6 – *Crowding Distance* calculation schema

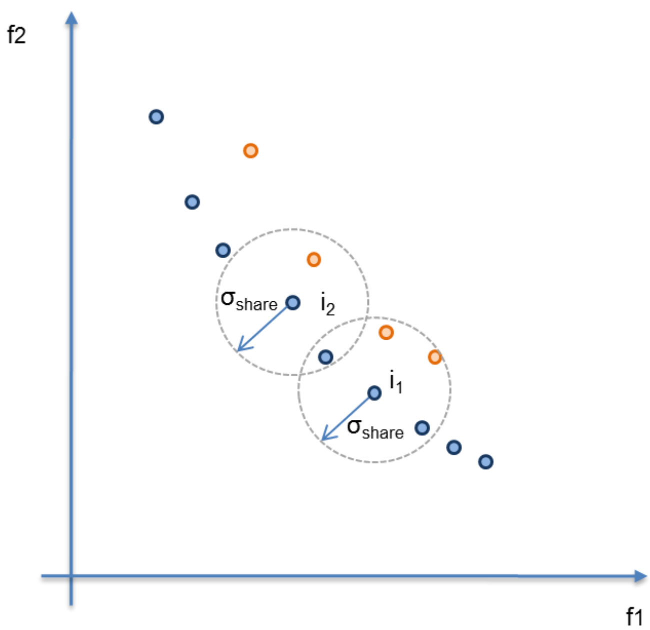


Figure 2.7 – *Fitness Sharing calculation schema*

2.6.1 Genetic algorithms - *GA*

The Genetic Algorithm (GA) was one of the first algorithms of AE category and makes analogy to natural selection mechanisms. It was proposed by Holland (1975) and became widely popular after the publication of Goldberg's book (1989).

GA works with an initial population, consisting of chromosomes, which are formed by genes (decision variables). The chromosomes may be binary or real representation, thus defining two types of algorithms, the real representation and binary representation. The binary representation is rarely used, because for problems with large number of variables, the encoding and decoding process requires lots of memory and processing time.

In GA there are three operators: selection, crossover and mutation, to generate a new population of individuals from a population of parents.

The selection operator is used to select the chromosomes on which is applied the of crossover operator. There exist different selection operators (Deb, 1999), and generally chromosomes with higher value of objective function value are more likely to be selected and survive.

A widely used crossover operator in real coded GA is the Simulated Binary Crossover (SBX, Deb and Agrawal, 1995, Deb and Berger, 2001, and Deb, 2009) that simulates the crossover at 1 (one) point of the binary encoding and uses a probability distribution around two parents to create offspring solutions.

There are two important aspects that make this version of the GA (SBX operator) more efficient than previous versions: offspring solutions are usually generated closer to parents, and the distribution of offspring solutions is proportional to the distribution of the parent solutions (Deb and Berger , 2001).

After selection and crossover, mutation is performed to preserve the diversity of the population. The mutation probability must be kept low because higher values may compromise good solutions (Deb, 1999). Mutation operators can be uniform or non-uniform or polynomial, and may vary with the number of generation, Deb (2009).

Multi-objective versions

After Golbert's work, who suggested the concept of non-dominance for the generation of new population, a first generation of algorithms has emerged. Three of these are shortly described as follows:

- NSPGA (Niched Pareto Genetic Algorithm), proposed by Horn et al. (1994): In this algorithm two solutions are selected at random in the population and its dominance is compared to another group of solutions that is also selected randomly at the start of generation. The size of this group is controlled by a user-defined parameter. Besides the selection operator described, this algorithm uses an operator of diversity (Fitness Sharing) if the two solutions selected for comparison are both dominated or both not dominated.

- MOGA (Multi-objective Genetic Algorithm) proposed by Fonseca and Fleming (1993): In this algorithm all non-dominated solutions receive a ranking and the other solutions given the value of ranking according to the number of solutions that dominate them. The selection is made of solutions with the lowest ranking and an operator of diversity, CD, is applied to solutions with similar objective function values.

- NSGA (non-dominated Sorting GA) proposed by Srinivas and Deb (1994): the first version of this algorithm had some differences with respect to MOGA. In the ranking is assigned to all solutions. In addition, to calculate all the solutions is an operator of diversity, the CD, from the variables of the solutions (and not the values of the objective function).

According to Deb (2008), these three algorithms were important to encourage research for the MOGA, but its efficiency was extremely low. Thus the second generation of MOEA raised that combine elitist conservation mechanisms with very efficient ranking algorithms and operators of diversity. These represent the state of the art in MOEA, and two of them are briefly described:

- NSGA-II proposed by Deb et al. (2000 and 2002): This algorithm uses a mechanism of non-dominated sorting, assigning them a level (ranking) to each solution. The algorithm generates the offspring population from the population of parents and the sorting procedure is performed in the population resulting from the union of the two. To generate a new population of size N (where N is the number of individuals) individuals that belong to the lower ranks (with best ranks) are first selected and if necessary to complete population or exclude individuals, the CD metric is used as tiebreaker. These processes used in NSGA-II are described in detail in methodology.

- SPEA and SPEA2 (Strength Pareto Evolutionary Algorithm) proposed by Zitzler et al. (1999 and 2001): unlike the NSGA-II algorithm that uses an external file to store the non-dominated solutions. Moreover, the mechanism of classification of non-dominated solutions is a bit different from NSGA-II, but also uses the idea of ranking by levels. An operator of diversity is used to prevent solutions from the external file that are close together, being used to generate the new population.

Other MOEAs using GA has been proposed and an extensive review of them can be found in Deb (2008 and 2009) and Nicklow et al. (2010).

2.6.2 Differential Evolution - DE

Differential Evolution (DE) was proposed by Price and Storn (1997) and is one of the most recent evolutionary algorithms for solving real-parameter optimization problems. DE exhibits an overall excellent performance for a wide range of benchmark problems. Furthermore, because of its simple but powerful searching capability, DE has got numerous real-world applications (Price et al., 2005). It uses a simple mutation operator based on the differences between pairs of solutions (also called vectors) in order to find the search direction based on the distribution of solutions in current population.

Unlike Genetic Algorithm (GA), which strongly depends on the crossover operator, DE uses mutation as the primary search mechanism and selection to indicate the direction within the feasible region of decision space

The basic idea of the DE is a scheme for the generation of the so called Test vector. DE generates this new vector from at least three members (depends on the strategy used) of the population. If the Test vector has a better value of objective function over a predetermined vector of the population, it will replace the vector which is being compared. In addition, the vector with the best objective function is always stored to keep track of progress throughout the optimization process. Price et al. (2005) point out that the use of distance information and direction between members of the population to generate random derivations results in an adaptive scheme with excellent convergence properties.

The original implementation maintains a pair of vectors of the population, both containing Np individuals and D size (number of variables). The current population, represented by $P_{x,t}$ is composed of vectors, $X_{i,t}$, equation (2.4), $P_{u,t}$ represents the test population that is composed of test vectors, represented by $U_{i,t}$, equation (2.5).

$$\boldsymbol{P_{x,t} = (X_{i,t}),\ \ i = 0, 1, \dots, Np - 1, \qquad t = 0, 1, \dots, gmax} \\ \boldsymbol{X_{i,t} = (x_{j,i,t}),\ \ j = 0, 1, \dots, D - 1} \tag{2.4}$$

$$\boldsymbol{P_{u,t} = (U_{i,t}),\ \ i = 0, 1, \dots, Np - 1, \qquad t = 0, 1, \dots, gmax} \\ \boldsymbol{U_{i,t} = (x_{j,i,t}),\ \ j = 0, 1, \dots, D - 1} \tag{2.5}$$

where $x_{j,i,t}$ is the decision variable

The initial population is randomly generated from the upper and lower limits of each decision variable (if there is no problem information available), as equation (2.6).

$$x_j = rand(0,1) \times (b_{j,U} - b_{j,L}) + b_{j,L}, \;\; j = 0,1,\dots,D-1 \tag{2.6}$$

where x_j is the decision variable and $b_{j,L}$ the lower limit of j variable and $b_{j,U}$ the upper limit of variable j.

Mutation in DE is used to create a population of NP test vectors. The original formulation uses three vectors chosen randomly to create the mutant vector $V_{i,t}$, as described in equation (2.7). The F parameter is a real number, usually between 0 and 1 (Vassan and Simonovic, 2008) and determines the mutation rate of the population over the generations. The indices of the vectors r0, r1, r2 are chosen randomly from position 0 to Np-1, for each generation and they are different from the current vector i.

$$V_{i,t} = X_{ro,t} + F(X_{r1,t} - X_{r2,t}), \;\; j = 0,1,\dots,D-1, \;\; t = 0,1,\dots,gmax \tag{2.7}$$

After mutation, a uniform crossover is applied to the test vectors generated in the mutation, as described in equation (2.8). The crossover probability CR determines the number or fraction of the variables combined (used) in the process. To determine whether a variable is considered for crossover, a random number is generated between 0 and 1, and compared to the crossover probability CR. If CR is greater than the random number generated, the variable $v_{j,i,t}$ is used to compose the trial vector, otherwise the variable $x_{j,i,t}$ is used. In addition, the variable j_{rand} randomly selected for comparison, is excluded from the trial vector to ensure that the variable $x_{j,i,t}$ will not be present twice in off-spring population.

$$U_{i,t} = u_{j,i,t} = \begin{cases} v_{j,i,t} & se \quad rand(0,1) \leq Cr \; ou \quad j = jrand \\ x_{j,i,t} & othrtwise \end{cases} \tag{2.8}$$

The selection consists on checking if the vector $U_{i,t}$ has a better objective function value than the vector $X_{i,t}$. If so $U_{i,t}$ is used for next population, otherwise $X_{i,t}$ is selected, as in equation (2.9).

$$X_{i,t} = \begin{cases} U_{i,t} & if \;\; f(U_{i,t}) \leq f(X_{i,t}) \\ X_{i,t} & otherwise \end{cases} \tag{2.9}$$

The Figure 2.8 presents a simplified flow chart of steps performed by an DE algorithm in the optimization process.

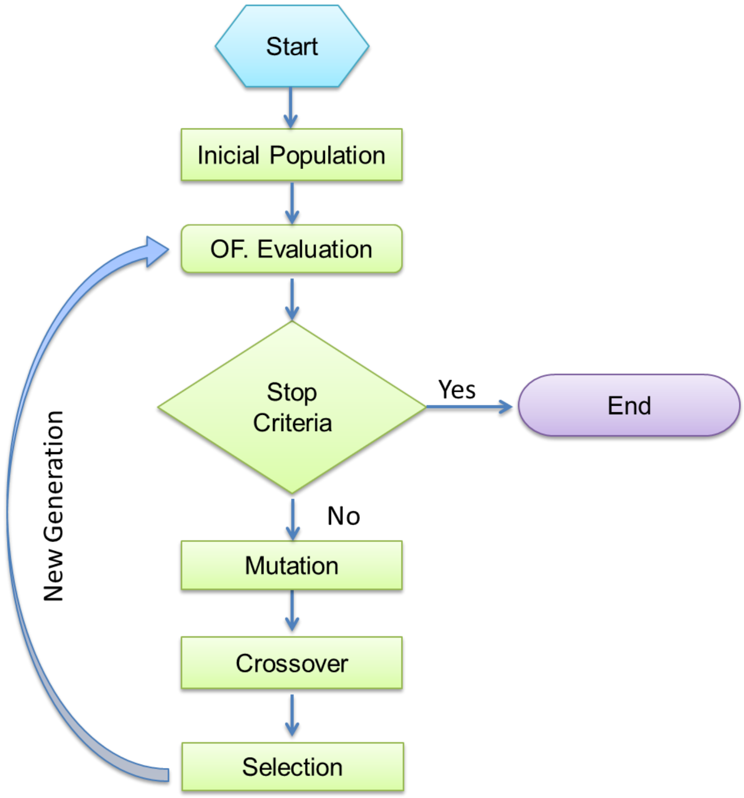


Figure 2.8 – General D.E. Flowchart

The general convention used by DE is DE/α/β/γ, where α is the vector to be perturbed (randomly selected), β the number of vectors considered for the mutation and γ the type of crossover used (can be exp: exponential or bin: binomial).

The variant described above is DE/rand/1/bin, also known as the classic DE, and is the original strategy proposed by Price and Storn (1997). The same authors have proposed other strategies listed below:

- DE/best/1/exp
- DE/rand/1/exp
- DE/rand-to-best/1/exp
- DE/best/2/exp
- DE/rand/2/exp
- DE/best/1/bin
- DE/rand/1/bin
- DE/rand-to-best/1/bin
- DE/best/2/bin
- DE/rand/2/bin

However, as noted by Vasan and Simonovic (2008) and Schardong et al. (2009), the strategy DE/rand/1/bin seems always gets the best results for most single-objective problems. A pseudo-DE is shown in the table of Figure 3.10.

Generation of Initial Population
While k < Max.Gen.
{
  While i < Np
  {
    Generate random numbers r0,r1,r2 in {1, … , Np}
    Calculate mutation, $\boldsymbol{V}_{i,t}$ as **equation (2.7)**
    Perform Crossover, $\boldsymbol{U}_{i,t}$ as **equation (2.8).**
    Perform Selection, $\boldsymbol{X}_{i,t}$, as **equation (2.9)**
  }
  $\boldsymbol{X}_{.,t+1} = \boldsymbol{X}_{.,t}$ is the new population
}
End

Figure 2.9 – DE’s pseudo code

Multi-objective versions

Like in GA, the most popular multi-objective based in DE are those that use the concept of non-dominance sorting and one diversity operator for the generation of new populations. In fact, some authors as Filipc and Robic (2005), Kwan et al. (2007) and Regulwar et al. (2010) have suggested the use of the mechanisms of NSGA-II with the DE. Kukkonen and Lampinen (2005) proposed the GDE3 (Generalized Differential Evolution), in which an external file is used to store non-dominated solutions and that are used in the process of generating the new population. Another algorithm, the PDE (Pareto Differential Evolution) proposed by Abbas and Sarker (2002), which also uses an external file to store the non-dominated solutions, and an diversity operator to eliminate solutions very close together in the repository.

Price et al. (2005) and Chakraborty (2008) present a detailed review of the existing MOEA based on DE, and can be noticed that many of the algorithms use the non-dominated sorting procedure and diversity operator Crowding Distance, as in the NSGA-II.

2.6.3 Particle Swarm Optimization - PSO

Like DE, Particle Swarm Optimization (PSO) is relatively recent algorithm and was initially proposed by Kennedy and Eberhart (1995). It is based on analogy with the flight formation of a flock of birds. As the DE and GA, the PSO is population based, and use a set of particles (individuals) that compose the population, in which the optimal solutions are sought through a combination of individual by learning and social behavior.

PSO is a very simple algorithm and the basic formulation for calculating the velocity and position (variables) update, proposed by Kennedy and Eberhart (1995) is presented in equations (2.10) and (2.11).

$$v_i(t+1) = w \times v_i(t) + c_1 \times r_1 \times [Pbi(t) - x_i(t)] + c_2 \times r_2 \times [Pg(t) - x_i(t)] \quad (2.10)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2.11)$$

where:

- *w* is the inertia coefficient which has an important role in determining whether the search will be around a local optimal (low values of *w*) or global (for higher value)
- c_1 and c_2 are constants that vary from 1.5 to 2.0, with the sum of not more than 4;
- r_1 and r_2 are random numbers uniformly distributed in the range 0 to 1;
- *Pbi(t)* is the best position of the particle *t* to date, also called the local best;
- *Pg(t)* is the particle with the best overall position of all particles, so far, also called the global best;
- $x_i(t)$ is the position vector of particle *i*, and
- $v_i(t)$ is the velocity of particle *i*.

Interestingly, the PSO does not have the crossover mechanism as GA and DE, and is common in PSO algorithms only accept the new particle if its objective function value is better than the previous one, featuring a selection process. However, this procedure can negatively affect the convergence of the algorithm. Pbi and Pg, the best local and best global particles, respectively, are updated at each generation, and Pbi ∈ Pg. The inertia coefficient *w* seems to be the one that mostly affects the performance of PSO (Coello Coello et al., 2004). Some authors recommend starting the optimization with high values of *w* and decrease them over. The suggested values for *w* are between 0.4 and 1.4. Benitez et al. (2005) suggest values of $c_1 = c_2 = 1$ and *w* = 0.5. Jung and Karnery (2006) suggest that high initial values of c_1 and c_2 should be used and the values should decrease over the generations. Suggested values range from 1.4 to 0.5.

The flowchart in Figure 2.10 shows the basic structure of a PSO base algorithm.

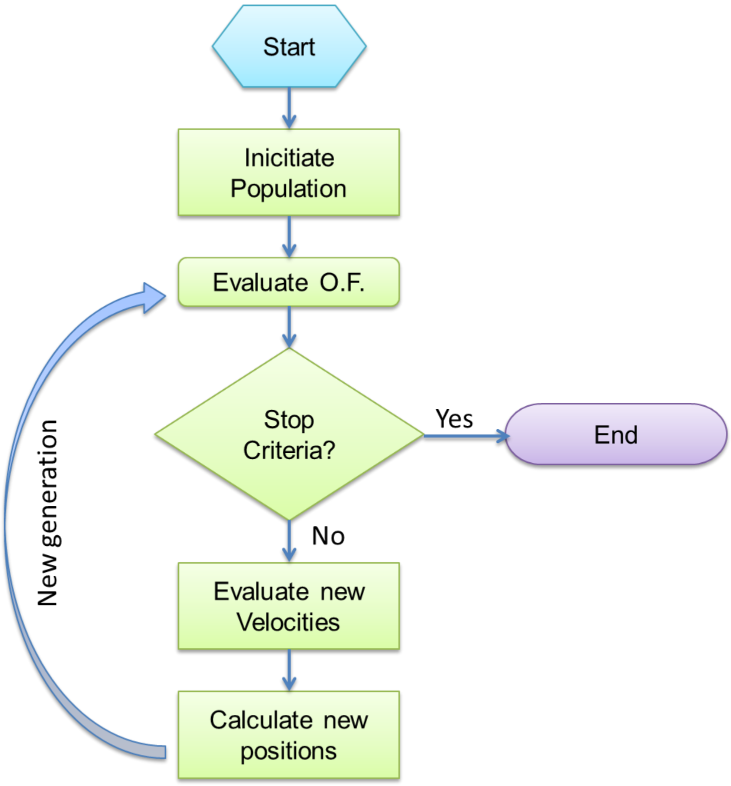


Figure 2.10 – Flowchart of a PSO based algorithm

Multi-objective Versions

The main difference between PSO and MOPSO is the way how of storage and selection of global best (P_g) particles. In fact, P_g in equation above makes no sense in multi-objective optimization, once there is no such thing as the global best on multi-objective. There are several strategies for storing non-dominated solutions and selecting one to calculate the next velocities and positions. Some authors suggest the use of an external repository to store non-dominated solutions (Coello Coello and Lechuga, 2002 and Coello Coello et al., 2004). To upgrade the velocity of each particle, this algorithm has to use a procedure to define how the best global P_g will be selected from the non-dominated external archive, also known as leader. There are several criteria to select a particle from the repository, and this strategy has the function of maintaining the diversity of solutions. Coello Coello et al. (2007) and Pulido (2005) propose a method of choosing the solution from the external repository combining randomness and proximity between solutions, with the goal of promoting diversity and accelerating the convergence. Lechuga and Rowe (2005) proposed a PSO based multi-objective algorithm that use an external repository, making the selection of the leader based on the value of fitness sharing. Baltar (2007) used an external file to store the non-dominated solutions as well and a method that calculates the density of points around each solution and apply a probability factor for the selection of the P_g particle. Chiu et al. (2007) proposed a new strategy, called cross-search strategy, based on the proximity of the particles and uses a mutation operator to maintain diversity. Benitez et al. (2005) suggests the use of an external repository without the use of diversity operators, that is, suggests that the leader P_g, should be done by choosing a random solution from the repository.

Algorithms that use external archives to store non-dominated solutions are also popular, but require an additional parameter, which is the size of the archive.

2.7 Non-dominated solution visualization

The multi-objective analysis process generates, ideally, a set of non-dominated solutions which is composed of complete solutions of the problem in analyse. Thus it is necessary to provide to the Decision Maker ways to evaluate/visualize and choose a solution of this set. This choice should reflect the manager's personal preferences, or preferences of individuals and groups involved in decision-making process. Compromise programming (Simonovic, 2009 and 2011) can assist DM in choosing the best compromise solution to a given set of weights for each objective. The compromise programming is used to generate the classification and thus determine the best compromise, or at least, the most roust solutions. Some existing graphical methods of visualization of the non-dominated solutions are also reviewed.

2.7.1 Compromise Programming

Compromise programming uses the measure of the distance of each objective value to the ideal point, for creating a rank according to the weights chosen by the DM for each objective function that should reflect its preferences.

The calculation of distances is done according to equation (3.13).

$$L_s(j) = \sum_{i=1}^{n} w_i^p \left(\frac{z_i^* - z_i(j)}{z_i^* - z_i^{**}} \right)^p \qquad (2.12)$$

Where n is the number of criteria (objectives, in this case), z_i^* the best value for criteria i and z_i^{**} the worst value for criteria i and $z_i(j)$ the value of criteria i for individual j.

The values of p define the norms and / or distances used, and as practical recommendation (Simonovic, 2009) the values of *p* are: 1, 2 and 100. The values of *w* should reflect user or DM preferences. In general a normalization criterion for *w* is adopted, for example, the sum of all *w* must be equal to 1.

After defining the values *p* and *w*, the Ls distance values are calculated and a descending sorting of the solution is done according to the values calculated of Ls, i.e., the lower Ls the better the solutions rank. The compromise programming is used to aid in choosing the best (or more robust) compromise solutions of the non-dominated set generated in a multi-objective analysis, when this is necessary or desired. Again, the better term here reflects the DM preferences expressed by the values of the weights (*w*) values provided for each objective.

2.7.2 Graphical Visualization

The graphical visualization of non-dominated set of solutions without any kind of transformation is only possible in of problems with up to three goals. But even with three objectives the visualization of non-dominated solutions can be difficult, since the computational tools available are still limited and the presentation and analysis of solutions in three-dimensional graphs is not always trivial.

Some authors present methods for graphical visualization of the non-dominated set of solutions. Deb (2009) presents some of these methods:

(i) Bar Chart: In this method the values of each objective are plotted as vertical bars in M sets of bars on the horizontal axis and are separated by a space. A typical example of this type of graph is shown in Figure 2.11. The values of each objective can be normalized and each solution presented in a different color. This method is inefficient for large number of solutions.

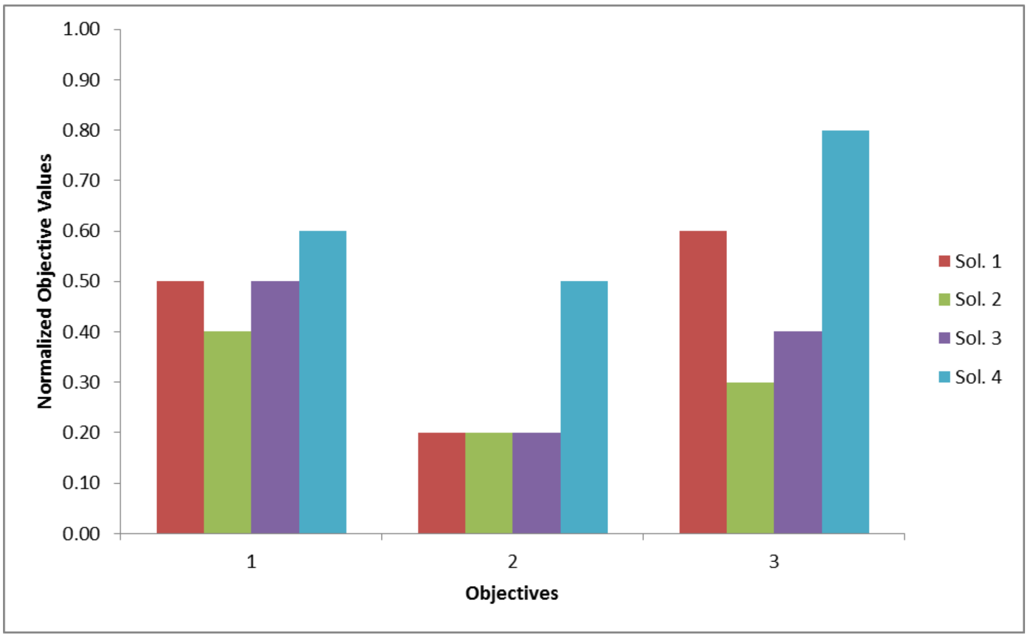


Figure 2.11 – Bar chart example – Source: Deb (2009)

(ii) Line Chart (Deb, 2009): In this method the values of objectives are plotted on the horizontal axis at position 1.2,..., M, where M is the number of objects, and values each for each solution are plotted on the axis and connected by lines. An example is shown in Figure 2.12.

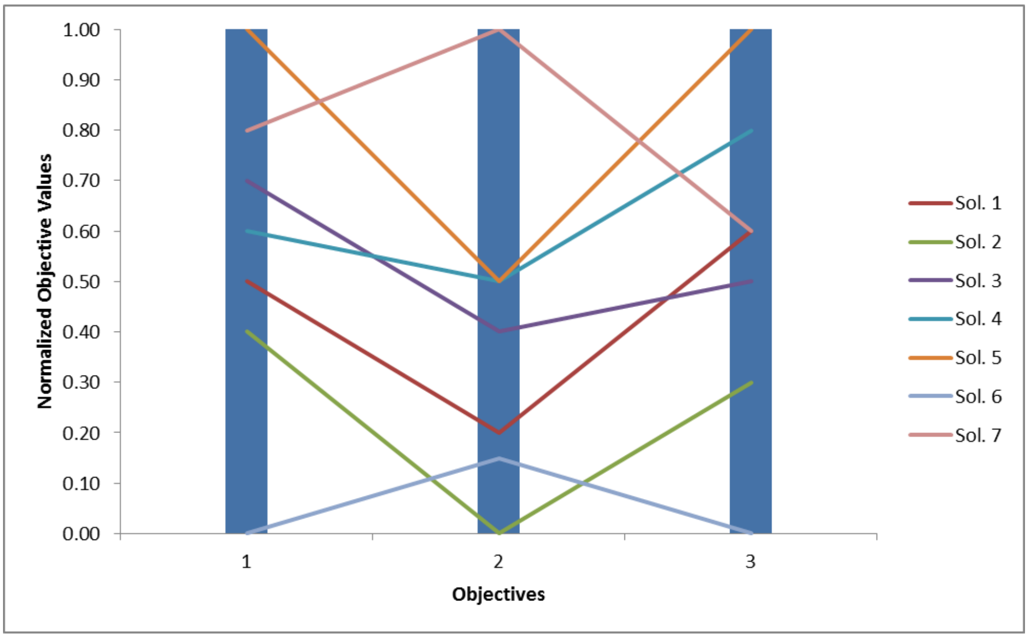


Figure 2.12 – Line Chart – Source: Deb (2009)

(iii) Star Coordinate: Mañas (1987) cited in Deb (2009) suggests that a circle is divided into equal parts corresponding to the number of objectives (M) of the problem. Thus the circle has M parts, which defines M radial lines connecting the center point to the edge of the circle, according to the example of Figure 2.13 with three objectives. Next, the author suggests plotting the points of each solution on these lines, and the center of the circle corresponds to the minimum value of each function and the maximum is plot at the end. These points (f1*, f2* and f3* in Figure 2.13) are connected defining a polygon, whose area defines the classification of the solution. Solutions with larger area are preferred over those with smaller area. The difficulty with this method is that each solution generates a graph, and thus in a problem with large number of solutions, the process of choosing the best one can be difficult.

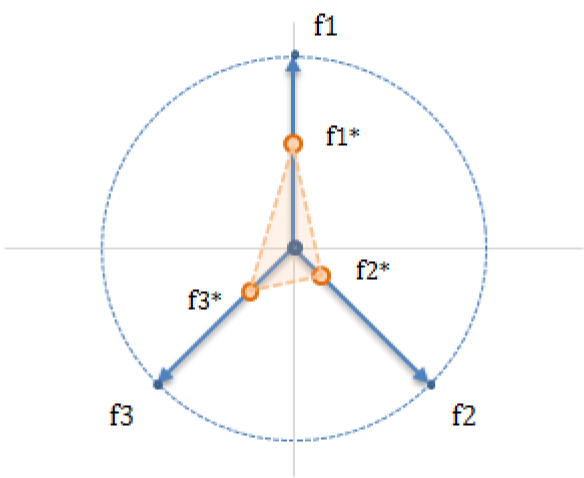


Figure 2.13 –*Star Coordinate* method

The technique of 2D data projection using the Star Coordinate method is quite common (Shaik and Yeasin, 2006). Baltar (2007) proposed a method that calculates the projection of the solutions (regardless of the number of objectives) in a circle by calculating the centroids of the polygon defined by the coordinates of the projected objectives. Similar methodology was used in this report, with a small variation in the way of highlighting the best compromise non-dominated solutions. This method was introduced in Decision Support System and is call Trade-off Graph (TG) and its use is exemplified in the applications of the algorithms.

2.8 Performance Metrics

The performance metrics are used to evaluate the performance of algorithms by comparing the results by applying them to standard test problems, of which the real Pareto front is known, although this is not a prerequisite for all of the metrics. The metrics presented are: a generation distance (GD), inverted distance indicator (IGD), spacing metric (SP), diversity index (DI) and Dominance Degree (DD). A good algorithm should get good results for all metrics, and a bad value for some of them may indicate some deficiency in determining the non-dominated solutions set.

2.8.1 Generational Distance - GD

This metric provides a measure of how close the set of tested solutions is to the Pareto front, Deb (2009). The metric formula is presented in equation (2.13) and a schematic with an example in presented in Figure 2.14.

$$GD = \frac{\sqrt{\sum_{i=1}^{n} d_i^2}}{n} \tag{2.13}$$

Where n is the number of solutions tested, d_i the distance between solution i and the nearest point of the known Pareto front. The lower the value of GD, the better the set of tested solutions.

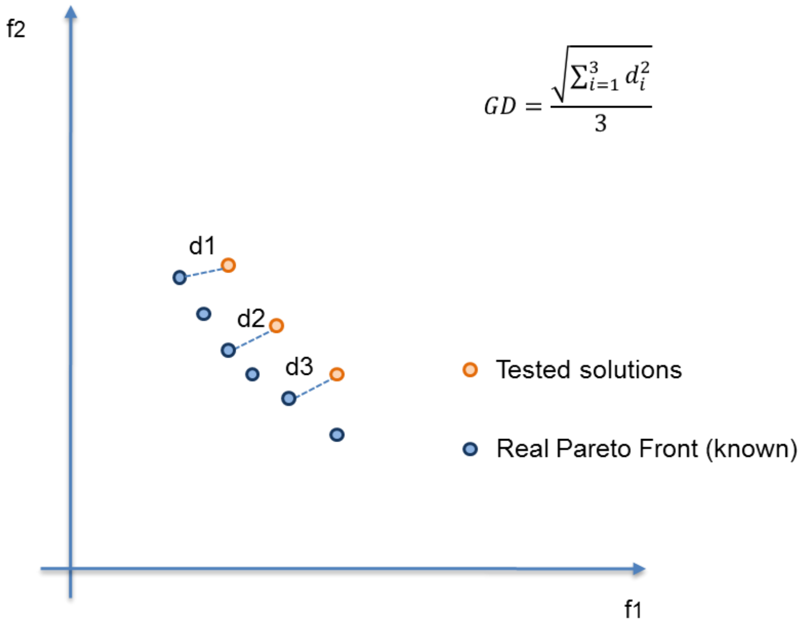


Figure 2.14 – Generational Distance – GD metric schema

2.8.2 *Inverted Generational Distance - iGD*

This metric is similar to the GD, but measures the distance from the points of the known Pareto front to the points of the set of tested solutions and was proposed by Pulido et al. (2005). The metric formula is presented in equation (2.14) and a schematic with an example in presented in Figure 2.15.

$$iGD = \frac{\sqrt{\sum_{i=1}^{n} dp_i^2}}{np} \tag{2.14}$$

Where np is the number of solutions of the known Pareto front, dp_i the distance between solution i of the known Pareto front and the nearest point of the test set. GD and iGD are complementary.

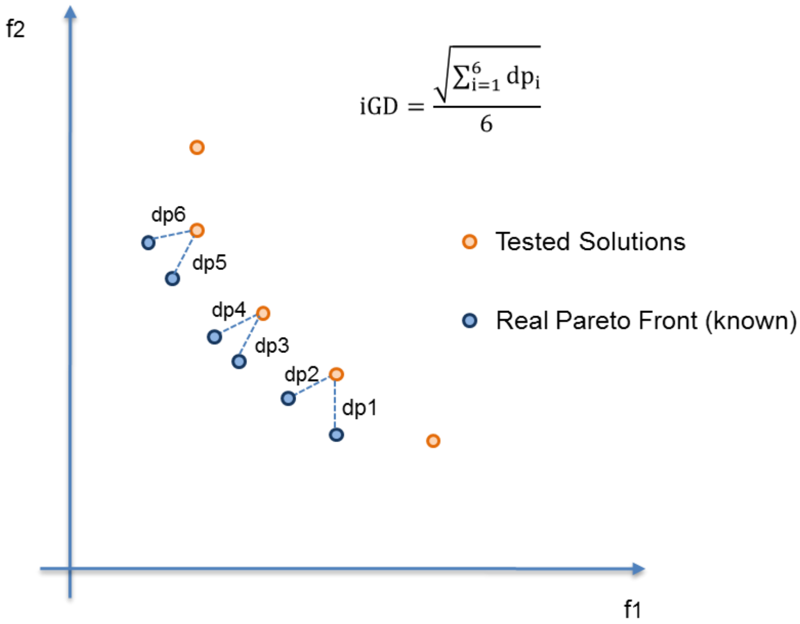


Figure 2.15 – Inverted Generational Distance – iGD schema

2.8.3 Spacing Metric - SP

This metric provides a measure of the spacing of the solutions set tested (Coello Coello et al., 2007 and Deb, 2009). The lower the value of SP, the better and more evenly spaced in the Pareto front solutions are. The metric formula is presented in equation (2.15) and a schematic with an example in presented in Figure 2.16. This indicator provides no information about the proximity of the solutions set tested to the real Pareto front, and thus it is not necessary to calculate this metric.

$$SP = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(\bar{d} - d_i\right)^2} \qquad (2.15)$$

Where n is the number of solutions of the test set, d_i the distance between the solution *i* and the nearest point on the known Pareto front and $\bar{d}$ the average of all d_i. The calculation of d_i is presented in equation (2.16).

$$d_i = \min\{|f_1^i(\vec{x}) - f_1^j(\vec{x})| + |f_2^i(\vec{x}) - f_2^j(\vec{x})|\}, \quad i,j = 1,2,\dots,n \tag{2.16}$$

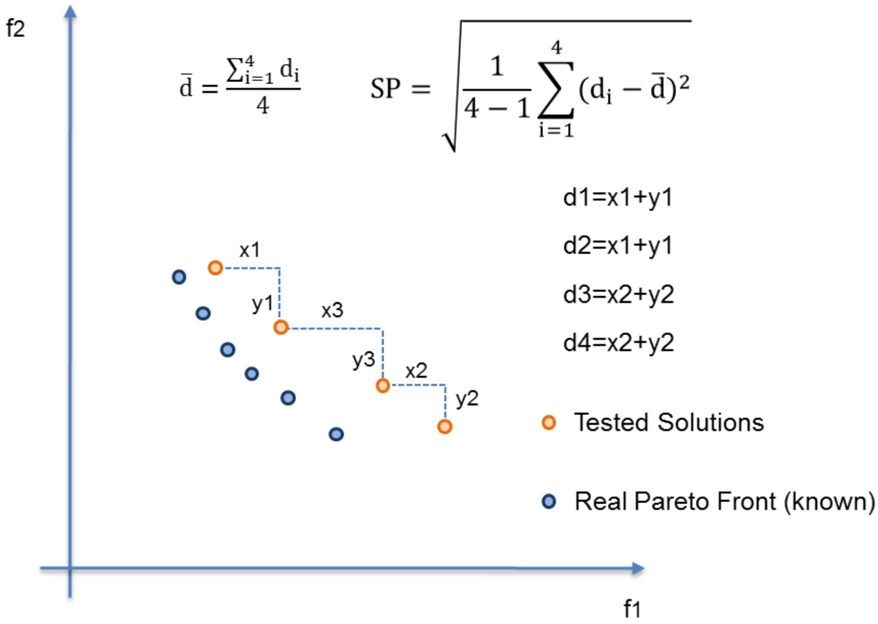


Figure 2.16 – Space metric – SP schema

2.8.4 Diversity Metric - ID

This metric provides a measure of the distribution of non-dominated solutions tested against a known Pareto front (Deb, 2009). The lower the values of ID, the better the solutions are distributed on the non-dominated front. The metric formula is presented in equation (2.17) and the scheme with an example is shown in Figure 2.17.

$$ID = \frac{df + dl + \sum_{i=1}^{n-1}|\bar{d} - d_i|}{df + dl + (n-1).\bar{d}} \tag{2.17}$$

Where n is the number of solutions of the test set, d_i the Euclidean distance between two solutions of the test set, $\bar{d}$ is the average of all d_i, dl and df are the distances of the extreme points of the test set to the extreme points known Pareto front.

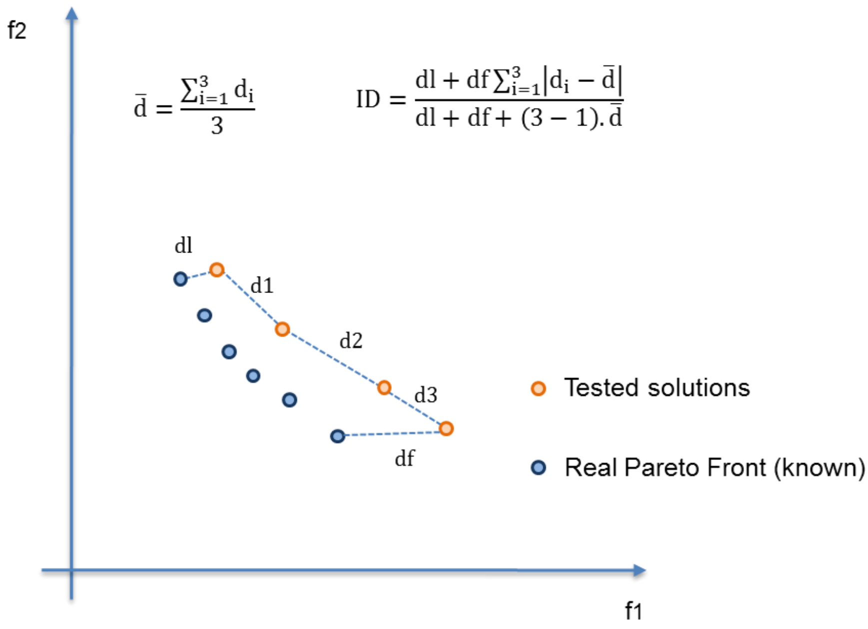


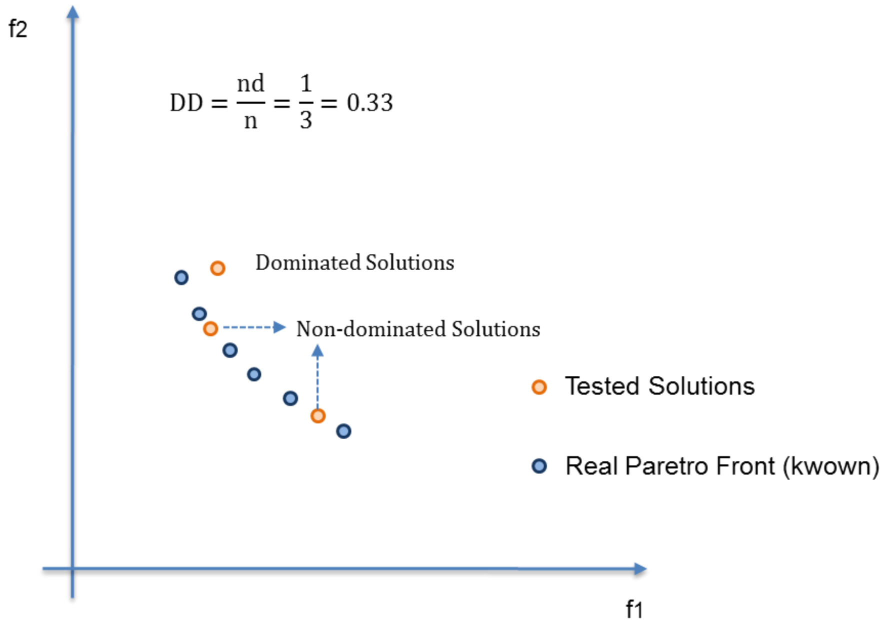
Figure 2.17 - Diversity Metric – ID schema

2.8.5 Dominance Degree - DD

This metric provides a measure of how many solutions of the test set that are dominated by the known Pareto front (Deb, 2009). The lower the value of DD, the closer to the Pareto front of test set is. The metric formula is presented in equation (2.18) and a schematic with an example in presented in Figure 2.18.

$$GRD = \frac{nd}{n} \tag{2.18}$$

Where *n* is the number of solutions of the test set, *nd* the number of solutions in the test set that are dominated by the known Pareto front.



or

Figure 2.18 – Dominance Degree – DD schema

2.9 Constraint handling in MOEA

Handling constraints within a MOEA represents a great challenge, particularly when dealing with real-world problems. Most real-world problems have constraints that need to be incorporated into MOEAs in order to avoid infeasible solutions. Water resources management usually have both "hard" (i.e., they must be satisfied) and "soft" (i.e., they can be relaxed) constraints. A classic example of a hard constraint on water systems is the mass balance in reservoirs.

Deb (2009), Coello Coello et al. (2007) and Chakraborty et al. (2008) present a review of techniques used to handle constraints. A common technique is the penalty function, which adds a penalty to the objective function when a constraint is not met. This penalty must be multiplied by a coefficient with magnitude of about 100 times the value of the objective function, Deb (2009). The difficulty with this method is to define the values of these penalty coefficients, since incorrect values for the coefficients can lead to slow convergence of the algorithm.

Another technique to handle constraint, that proves to be very efficient and is preferred by these authors, is to favor solutions with lower infeasibility in the selection process (for single-objective EA) and on the non-sorting procedure, for the MOEA (Deb, 2009).

2.10 Application of MOEA in water resource management

Evolutionary algorithms have been widely applied in water resources management problems and GA is one of the most used, possibly because it was one of the first to be proposed. Baltar (2007), Nicklow et al. (2010) and Adeyemo (2011) present a comprehensive review of GA applications in water resources management systems.

This report focus on multi-objective optimization and a simplified review of the application of MOEA is presented in Table 2.1, in descending order by year of publication. Some of the references listed describe the application to simplified water resources systems, where the analysis is usually performed considering a short time period of simulation, due to the complexity and difficulty of solving problems with multiple reservoirs using MOEA.

Table 2.1 – Applications of multi-objective evolutionary algorithms in water resources

References	Description
Regulwar et al. (2010)	Application of a Differential Evolution (DE) algorithm for optimal operation of a single reservoir used for hydropower generation and irrigation. The constraints imposed to the problem were: maximum flows for hydropower generation, maximum irrigation flow, reservoir volume limitations and the monthly balance equation. DE/best/1/bin strategy produces better results than NSGA-II.
Baltar (2007)	Application of a Multi-objective PSO and NSGA-II to series of problems related to water resources management, including a simple reservation system.

References	Description
Reddy e Kumar (2007)	Application of a multi-objective Differential Evolution algorithm for optimization of a water resource system. The comparison was made with NSGA-II. The application of case study was done on a single reservoir system with a period of 12 months of simulation, and the objectives were: meet demand for irrigation and hydropower generation.
Kim et al. (2006)	Application of NSGA-II algorithm to a system of multiple reservoirs (3 in cascade) with optimization of two conflicting objectives of the Han River basin. Three different scenarios were analysed. In this study the penalty function was used to consider the constraints of the problem. The simulations were made for a period of 12 months.
Reis et al. (2005 e 2006)	Application of genetic algorithms combined with linear programming for operation of a reservoir system considering multiple objectives.
Yandamuri et al. (2005)	Application of NSGA-II for optimization of the location for effluent discharges, considering multiple objectives: cost of treatment, even distribution of releases and performance measurement for check the violation of Max limit of dissolved oxygen.
Tang et al. (2005)	Review of the application of evolutionary algorithms for the calibration of hydrologic models, considering multiple objectives.
Suen et al. (2005)	Application of NSGA-II in the Dahan River Basin in Taiwan for multi-objective analysis. The objectives analysed were: demand for urban water supply and maintenance of downstream minimum flow (ecological flow)
Prasad e Park (2004)	Application of NSGA-II to the optimal design of water distribution networks. The objectives were: minimization of cost and maximize network reliability.

2.11 Decision Support System - DSS

Despite several decades of intensive research in the application of optimization models the system of reservoirs, authors such as Yeh (1985), Wurbs (1993) and Labadie (2004) have noted a large gap between theoretical advances and applications to real world problems. Some causes are appointed:

- many reservoir system operators are skeptical about models purporting to replace their judgment and prescribe solution strategies and feel more comfortable with use of existing simulation models;
- computer hard-ware and software limitations in the past have required

simplifications and approximations that operators are unwilling to accept;

- optimization models are generally more mathematically complex than simulation models, and therefore more difficult to comprehend;
- many optimization models are not conducive to incorporating risk and uncertainty;
- the enormous range and varieties of optimization methods create confusion as to which to select for a particular application;
- some optimization methods, such as dynamic programming, often require customized program development; and
- many optimization methods can only produce optimal period-of-record solutions rather than more useful conditional operating rules

Also according to Labadie (2004), many of the problems mentioned are being overcome with the use of optimization models integrated into Decision Support Systems (DSS), combined with advances in hardware and software for personal computers.

According to Porto and Azevedo et al. (1997) and Porto et al. (2003), Decision Support Systems is a methodology to aid decision making based on extensive use of databases and mathematical models. The Figure 2.19 shows the typical structure of a Decision Support System.

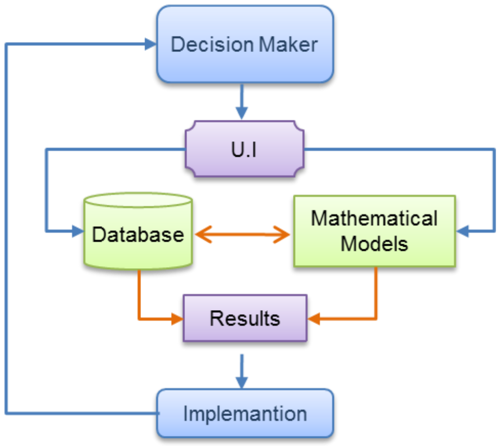


Figure 2.19 – Typical structure of a Decision Support System, Source: Porto et al. (2003)

2.11.1 AcquaNet DSS

AcquaNet is a Decision Support System that uses a network flow algorithm for the simulation/optimization process. AcquaNet has a database that stores the input data of the models, as well as the results generated. AcquaNet has a modular structure incorporating mathematical models developed for analysis of different problems related to the use of water resources. This structure consists of modules for: water allocation, water quality assessment, water allocations for irrigation, hydropower production and economic analysis. The DSS is widely used in Brazil for water resource management (Porto et al., 2003 and 2005). AcquaNet's user interface allows the user to build the entire topology of the problem is formulated using only the mouse and a series of icons (representing reservoirs, canals, we pass, etc.).

2.11.2 ModSim DSS

ModSim DSS is another Decision Support System containing a graphical interface that enables the user to create the topology the represents it's problem in a generalized way

(Labadie, 2010). The topology, as well as in AcquaNet, is built in network flow representation, and data is stored in files that play the role of the database. ModSim also uses an efficient networkflow optimization algorithm called Relax IV that combines simulation with optimization to determine optimal flows on the arcs of the network.

3 Methodology

In this session the two algorithms developed for multi-objective analysis MoDE-NS (Multi-objective Non-dominated Sorting DE) and MoPSO-NS (Multi-objective Non-dominated Sorting PSO) are described. The NSGA-II has also been translated into the programming language used, but its original characteristics were preserved, except for random number generation. A common framework for manipulation of populations and non-dominated solutions based on NSGA-II is used, allowing the three algorithms to be integrated into an optimization library, detailed in the item below.

First, the non-dominated sorting procedure, the diversity operator mechanisms, and the constraint s handling scheme used in the algorithms are described. Then the algorithms are described, including NSGA-II as well as the Decision Support System. Finally, the methodology for the visualization of non-dominated solution set is presented and the integration with the Decision Support Systems as AcquaNet DSS and ModSim DSS is discussed.

The computer codes were developed in C # (C sharp) that is very efficient in terms of time processing and memory usage.

3.1 Non-dominance Sorting and Diversity

For multi-objective version of the GA over the methodology used is the classification of Pareto front, originally proposed by Golbert (1989), also known as the non-dominated sorting procedure - NS. This method, also known as the non-dominated sorting procedure - NS, shown in Figure 3.1, consists of sorting the solutions and assigning each one a rank according to their "dominance", i.e. the first non-dominated solutions receive rank 1 and are removed from the population. Once these non-dominated solutions are removed, the next non-dominated solutions are given a rank value 2, and this process is repeated until the entire population is classified. This procedure is used in NSGA-II and has been extended to the two algorithms developed here, the MoDE-NS and MoPSO-NS.

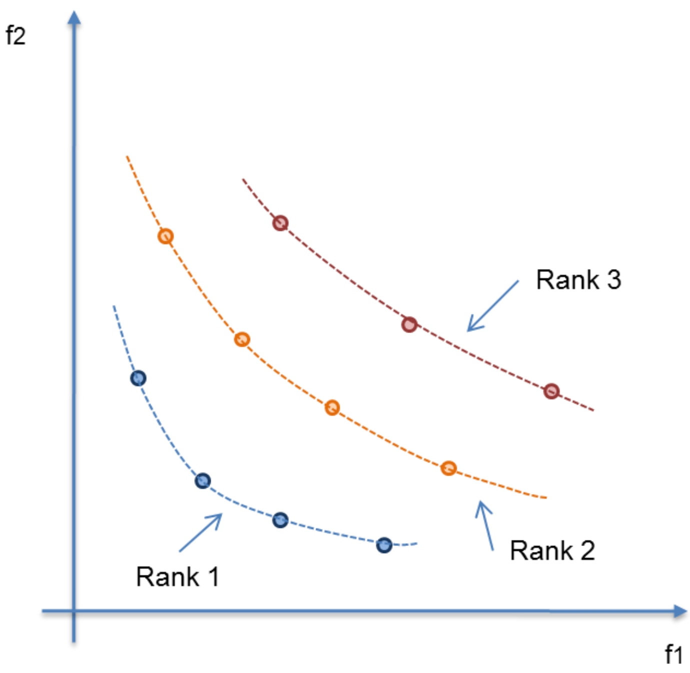


Figure 3.1 – Non-dominated solution ranking schema

To maintain the diversity, the Crowding Distance (CD) is used, which calculates the density of points around each solution, as in Figure 3.2. This operator is used to determine whether a solution will be part of the next generation, i.e., it is used as criteria for sorting set of solutions with the same rank. The CD is calculated according to equation (3.1) that is applied to each solution in the population.

$$CD(k) = \sum_{j=1}^{n} \frac{f_k(i+1) - f_k(i-1)}{{f_k}^{*} - {f_k}^{**}}, \qquad k \in \mathcal{F}^*, \mathcal{F}^* = 1,2,\dots,m \qquad (3.1)$$

Where k is the individual who belongs to the subset of solutions at each level ($\mathcal{F}^*$) and m is the number of fronts that exist in the sorted population j the index of the objective function (OF), n the number of objectives, f_k the value of the OF, ${f_k}^{*}$ e ${f_k}^{**}$, are respectively the best and worst value for the subset of solutions.

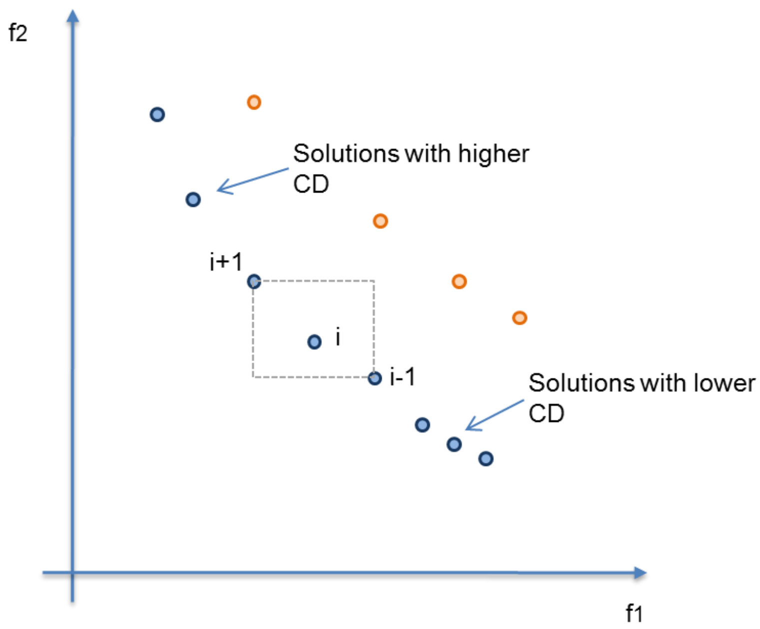


Figure 3.2 – *Crowding Distance* calculation schema

3.2 Algorithms Description

In this section the developed algorithms MoDE-NS and MoPSO-NS are described, which use two sets of populations (parents and child, or offspring) as in NSGA-II. Moreover, the elitism and diversity mechanisms used are those described above: the non-dominated sorting procedure (NS) and crowding distance (CD).

NSGA-II was implemented in the same structure of the MoDE-NS and MoPSO-NS, to compose the so called optimization library, however, its original features were maintained. It was used a benchmark and validation for MoDE-NS and MoPSO-NS, in both test problems and water resources problems.

The structure used is composed by two pairs of populations with size N. The main population, called Pt always contains the non-dominated solutions and the solutions whit lowest (best) rank, using methods described above. Thus, there is no need to maintain an external file for non-dominated solutions as suggested by some authors for MOEA based in DE and PSO. The second population, called Qt, is used to store the offspring (child) population in each generation.

Each generation of AEM at least 6 common steps are executed:

1. Generation of initial population: P0;
2. Non-dominance sorting and CD assigning for P0;
3. Generation of offspring population (Qt): at this step is the reproducing schemes are used NSGA-II, MoDE-NS or MoPSO-NS and determine the efficiency of the result.
4. The two populations are merged Pt U Qt: Rt;
5. Non-dominance sorting and CD assigning to the set Rt;
6. Selection of new population P (t +1)

The steps described above are the basic algorithms and presented schematically in the flowchart in Figure 3.3 and a pseudo code that is shown in Figure 3.4. Note that at the end of the optimization process, the non-dominated sets of solutions are stored in population P (t +1).

Although this population may contain more than one front, and thus, only solutions with rank 1 (non-dominated solutions) will be considered on the result analysis.

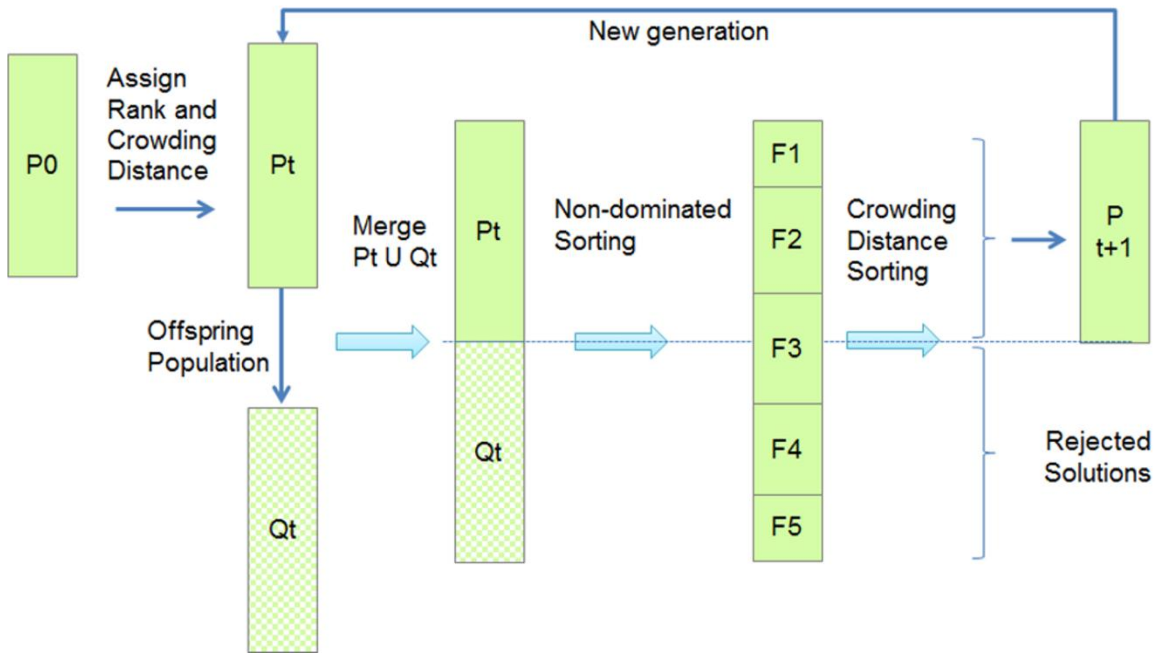


Figure 3.3 – Procedure for MOEA developed

Although simple, this structure proved to be very efficient for the MoDE-NS and MoPSO-NS. In following items the process of generating the offspring population Qt for of each algorithm, as well as their characteristics and specificities are presented. We can note at this point, that the difference between the three algorithms is the way that Qt population is generated, which is a determining factor in the quality of results.

```
Generation of initial population: P0
Non-dominated sorting and CD assigning to (P0)
While i < GerMax
{
   Generation of Q(t)
   Merge P(t) U Q(t) = R(t)
   Non-dominated sorting and CD assigning to(Rt)
   Selection of P(t+1), rejecting Rt – P(t+1)
}
End – P(t+1) stores the non-dominated solutions with better rank.
```

Figure 3.4 – Pseudo code of MOEA developed

3.2.1 MoDE-NS

For the MoDE-NS algorithm developed the most important operators for Multi-objective optimization where used, as well as most of DE strategies for generation of te offspring population. A self-adaptive schema for DE's control parameters F and CR is also used. A mutation procedure on the offspring population is include on MoDE-NS as well as and different elitism selection procedure is tested. These aspects are discussed below.

For the MoDE-NS algorithm the main strategies existing in DE were implemented for the generation of the offspring population Qt from parent Pt. The multi-objective version has some peculiarities compared to the single-objective version, which are related to the choice of the vector α of the DE strategies (DE/α/β/γ). In the developed MoDE-NS, the vector α of the considered strategies can be: *rand (random)*, *best*, *rand-to-best* and *current-to-best* as in Table 3.1, that shows how the vector $V_{i,t}$ if calculate for each strategy, Chakraborty (2008).

Table 3.1 – MoDE-NS strategies

Strategy	Mutation Vector $V_{i,t}$.
DE/rand/β/γ	$V_{i,t} = X_{r0,t} + F\sum_{\beta=1}^{2}\left(X_{r1+2.(\beta-1),t} - X_{r2+2.(\beta-1),t}\right)$
DE/best/β/γ	$V_{i,t} = X_{best,t} + F\sum_{\beta=1}^{2}\left(X_{r1+2.(\beta-1),t} - X_{r2+2.(\beta-1),t}\right)$
DE/rand-to-best/β/γ	$V_{i,t} = X_{r0,\mathrm{t}} + F\sum_{\beta=1}^{2}\left(X_{rbest,t} - X_{r1+2.(\beta-1),t}\right)$
DE/current-to-rand/β/γ	$V_{i,t} = X_{i,t} + F\left(X_{r1,t} - X_{r2,t}\right) + K\left(X_{r3,g} - X_{i,t}\right)$

In MoDE-NS the criteria used for selecting individuals vary as follows, depending on the choice of strategy:

- Strategies of type DE/rand: an individual is chosen at random from the population Pt and used in the generation of mutation vector.
- Strategies of type DE/best: in multi-objective analysis there is no concept of global best. Thus, for this type of strategy is adopted the following criteria: two individuals with the same rank are randomly chosen, and the dominated individual is then chosen as $X_{best,t}$.
- Strategies of type DE/current-to-rand: in this strategy the vector used to generate $V_{i,t}$ is the $X_{i,t}$ vector, regardless of their rank in the population Pt, that means that all individuals are used to generate the offspring population.

The values of β can be 1 or 2, for β = 1, we have the combination of two vectors $X_{r1,t} - X_{r2,t}$ and for β = 2, two additional vectors are combined $X_{r3,t} - X_{r4,t}$.

The value of γ indicates binomial (bin) or exponential (exp) crossover. The choice between them is not very important in the optimization process. The exponential crossover is similar to the two point crossover of GA and the binomial similar to uniform crossover. This step if performed after the mutant vector $V_{i,t}$ is generated, generating the so-called test vector $U_{i,t}$ as in equation (3.2).

$$U_{i,t} = u_{j,i,t} = \begin{cases} v_{j,i,t} & if\ rand(0,1) \le Cr\ or\ \ j = jrand \\ x_{j,i,t} & otherwise \end{cases} \tag{3.2}$$

The last step for the generation of offspring vector is to check if the value of O.F. test vector $U_{i,t}$ is better than the original vector $X_{i,t}$, $f(U_{i,t}) \le f(X_{i,t})$. However, in multi-objective analysis the O.F. comparison is not possible, the vector $U_{i,t}$ is accepted to be part of the new generation Qt

However, to bring the selection criteria from DE to MoDE-NS, a parameter that can be set by the user was included. This parameter controls if the algorithm should accept the test vector $U_{i,t}$ only if it dominates$X_{i,t}$,, as in equation (3.3). In practice this mechanism was not efficient, although no negative impact on processing time was noticed. This process is called “Selection Elitism”.

$$X_{i,t} = \begin{cases} U_{i,t} & if \quad U_{i,g} \preccurlyeq X_{i,t} \\ X_{i,t} & otherwise \end{cases}, \qquad where\ X_{i,t} \in \ Qi \tag{3.3}$$

A second mechanism was introduced on DE basic parameters F and CR. User can define them as (i) fixed, (ii) variable with the number of generations between a minimum and maximum values (self-adaptive with linear growth) or (iii) as random within a defined range. In practice the linear variation (ii) presented the best results.

Finally, an additional mechanism of Non-Uniform mutation was used, whose mutation rate decreases with the increasing of the number of generations, according to equation (3.4) (Deb, 2009 and Coello Coello et al., 2010). This mechanism is important to improve the diversity and the results may be slightly better, and it is represented by PMNu (non-uniform mutation probability) in the DSS.

$$\Delta = \ 1 - rand(0,1)^{a}$$

$$a = \left(1 - \frac{t}{Max\ Ger}\right)^{b} \tag{3.4}$$

$$X_{i,t} = x_{j,i,t} = \begin{cases} x_{j,i,t} + \tau\left(b_{j,U} - b_{j,L}\right).\Delta & if \quad rand(0,1) \leq PMNu \\ x_{j,i,t} & otherwise \end{cases}$$

where τ takes 1 or -1 with 50% of changes for each

Where Δ is the factor that is applied to each variable within the upper and lower bounds: $(b_{j,U} - b_{j,L})$. $X_{i,t}$ is offspring population generated from $U_{i,t}$, g the current generation, MaxGer the maximum number of generations, and b is a fixed parameter, whose value adopted was 5, so that for values of g close to the maximum number of generations the factor Δ is very close to zero. The probability of mutation defines the percentage of the population that suffer mutation. It is possible to notice that Δ is very close to zero for values of g close to the maximum number of generations.

To better understand the steps of MoDE-NS to generate the offspring population Qt from the parent population Pt are detailed in flowchart shown in Figure 3.5. Qt * is used in the flowcharts to represent an intermediate population (which can suffer mutation before it becomes final population Qt), but for practical purposes, it is always represents the same variable (data structure) in the algorithm.

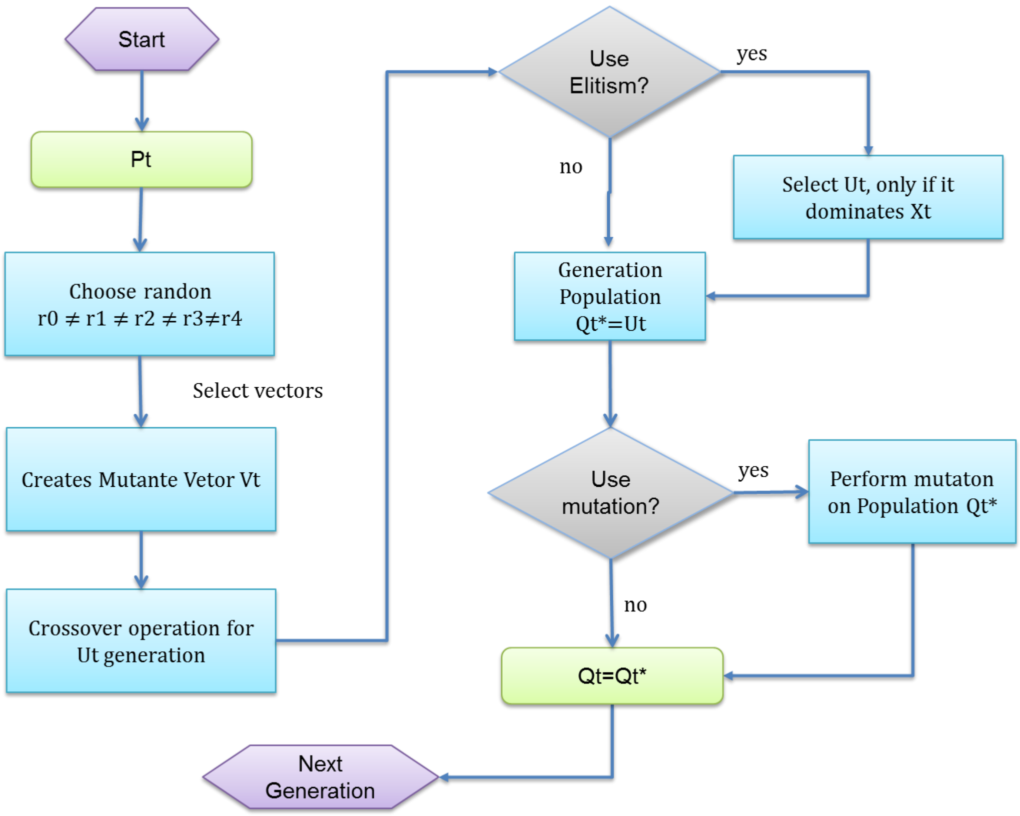


Figure 3.5 – Offspring population generation schema for MoDE-NS

Table 3.2 presents the parameters for MoDE-NS. This table also shows the recommended values, results of experiments with test problems and application to water resources problems. These parameters in the can be configured in the DSS on a specific screen as presented in Figure 3.6.

Table 3.2 – MoDE-NS Parameters

Parameter	**Description**	**Recommended Value**
Strategy	Offspring population strategy Generation	current-to-rand/1/bin
F	Mutation Factor	0.4 – 0.6
CR	Crossover rate	0.8 – 1,0
Parameters variation type	Parameters (F, CR) variation type: {Constant, Linear, Random}	Linear

Parameter	Description	Recommended Value
Use PMNu	Set the use of non-linear mutation: {Yes, No}	Yes
PMNu	Value of PMNu to be used	Min(1/N. Var.; 0.01)
Elitism Selection	Set the use of Elitism for Selection procedure: {Yes, No}	No

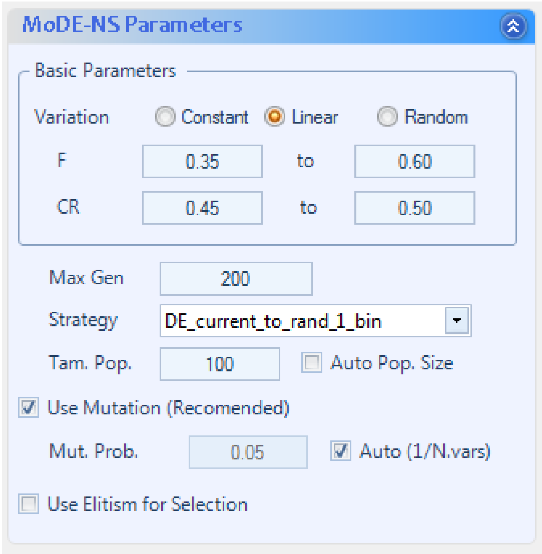


Figure 3.6 – MoDE-NS parameter configuration screen

3.2.2 MoPSO-NS

As in MoDE-NS, MoPSO-NS was developed incorporating the state-of-the-art mechanisms for the generation of the offspring population. The algorithm developed here also uses the same schema of maintaining two populations as in MoDE-NS, that is not very common to algorithms derived from the PSO. Many authors use an external file to store the non-dominated solutions (Coello Coello, 2007).

The PSO algorithm, the update procedure of the velocity and positions of the particles (variables), used in addition to the basic parameters, the information of the best position of each individual and the individual with the best global O.F. value. Naturally as in MoDE-NS, there no exist global optima concept in MoPSO-NS, and therefore there is the need to adopt a criterion for the selection of the individual that plays the rule of best global.

As in MoDE-NS the non-dominated solutions with better rank according to the no-dominated sorting procedures, are stored in the parent population Pt. MoPSO-NS selects two individuals at random from Pt, which are compared with respect to their dominance and the winner is chosen as leader or best global solution. An elitism mechanism was introduced in MoPSO-NS, so that only those solutions with a level 1 are candidates for selection of the individual who has the best overall paper, Pg(t) of equation (3.5). This mechanism is controlled by a parameter that was called "Leader Elitism". If the parameter is not set by user, all individuals in the population Pt is likely to be chosen.

To store the local best individual for each position i of the population a simple list is used, which is updated using the dominance concept. This process works as follows: initially the offspring population is generated and the values of objective functions are updated. Once this process is finished the update the local best (Pbi(t) of equations (3.5)) is performed, by comparing it with the new generated. If the new individual dominates Pbi(t), it takes place of the old one.

$$v_i(t+1) = w * v_i(t) + c_1 * r_1 * [Pb_i(t) - x_i(t)] + c_2 * r_2 * [Pg(t) - x_i(t)] \tag{3.5}$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{3.6}$$

An additional mutation operator was used in the MoPSO-NS to maintain the diversity of the population. This is exactly the same mechanism applied to MoDE-NS as described above and equation (3.4).

Figure 3.7 presents the flowchart of the MoPSO-NS to generate the offspring population.

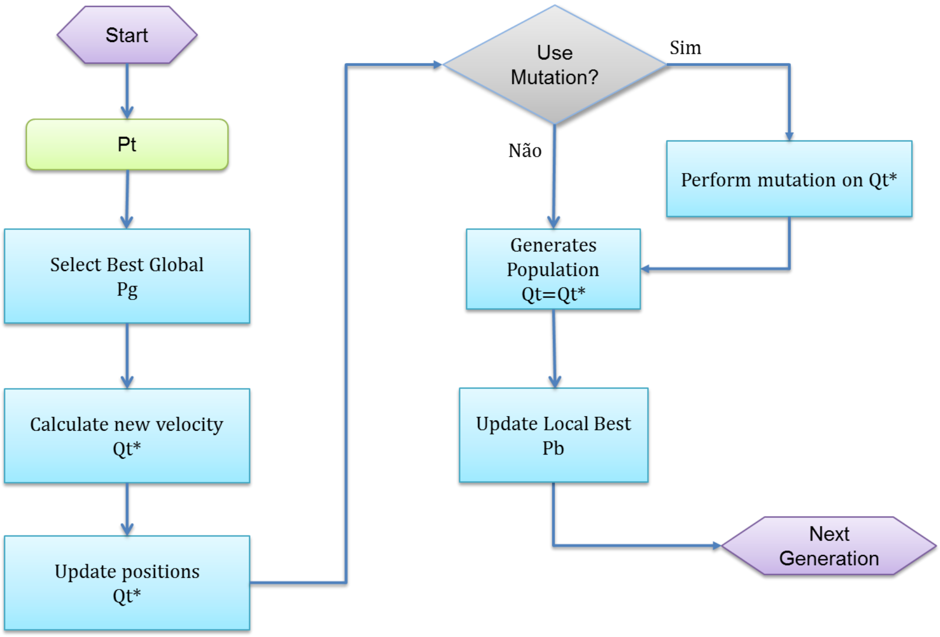


Figure 3.7 – MoPSO-NS Flowchart

Table 3.3 presents the parameters required for the MoPSO-NS. This table also present the recommended values by the authors, results of experiments with test problems and application problems in water resources studies. These parameters in the DSS can be configured in a specific screen, as Figure 3.8.

Table 3.3 – MoPSO-NS parameters

Parameter	**Description**	**Recommended value**
W	Inertia coefficient	0.40 – 0.10
C1	PSO c1 Constant	0.30 – 0.60
C2	PSO c2 Constant	0.50 – 0.80
Parameters Variation type	Parameters (w, c1, c2) variation type: {Constant, Linear, Random}	Random
Use PMNu	Set the use of non-linear mutation: {yes, no}	Yes

Parameter	Description	Recommended value
PMNu	Value of PMNu to be used	Min(1/N. Var.; 0.05)
Leader selection Elitism	Set the use of Leader selection (Best Global Particle): {Yes, No}	Yes

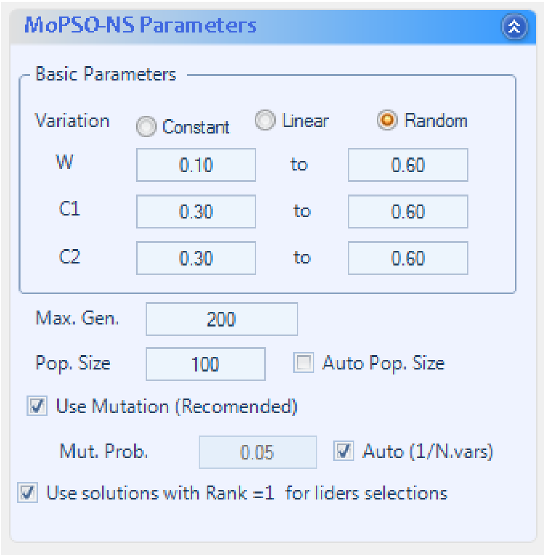


Figure 3.8 –MoPSO-NS dos Parameters configuration

3.2.3 NSGA-II

In NSGA-II, GAs operators are used for generation of the offspring population (Qt) from the parent population Pt. These operators are: selection, crossover and mutation.

For the selection process, two pairs of individuals are randomly chosen and compared for their dominance. The dominant individuals are the winners and used for crossover process. In NSGA-II, SBX crossover is used (Deb, 2009).

After crossover, a polynomial mutation is performed, Deb (2009). This mutation process is similar to the non-uniform mutation process described and used in MoDE-NS and MoPSO-NS, with the difference that the value of the mutation does not decrease with the increase in the number of generations. The flow chart of NSGA-II is shown in Figure 3.9.

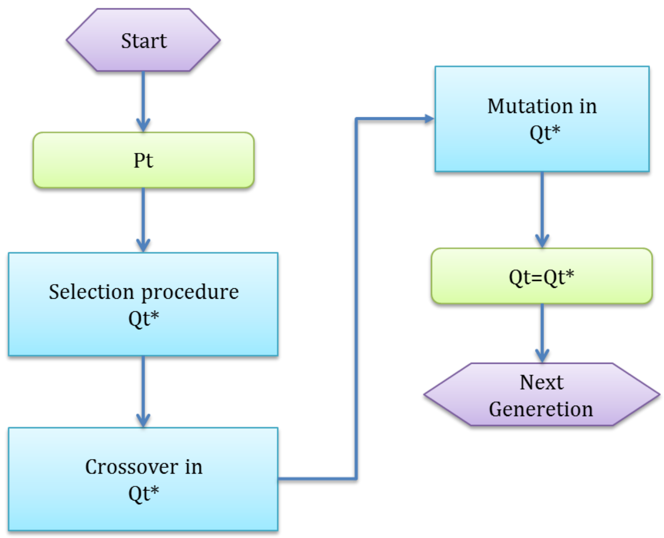


Figure 3.9 –NSGA-II Chart flow

Table 3.4 presents the parameters required for NSGA-II. This Table also presents recommended values by the authors, results of experiments with test problems and application problems in water resources problems. These parameters in the DSS can be configured in a specific screen, as in Figure 3.10.

Table 3.4 –NSGA-II Parameters

Parameter	**Description**	**Recommended Value**
Crossover Prob.	Crossover Probability	0.90
Mutation Prob.	Mutation Probability	Min(1/Num.Var; 0.05)
SBX Mutation Index	SBX Mutation Index	20
SBX Crossover Index	SBX Crossover Index	20

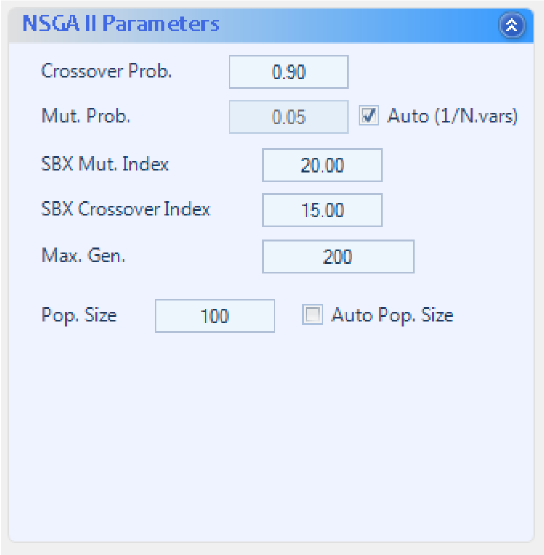


Figure 3.10 –Parameters do NSGA-II configuration Screen

3.3 Constraint handling and dominance

In the development of an AE is common for restrictions to be considered as penalties in the OF, as presented in the previous item. This method was not efficient for the problems analyzed in this report, and for this reason the methodology proposed by Deb (2002) was used. This methodology consists of: (i) the restrictions for each individual of the population is evaluated; (ii) this value of constraints is used as the first criteria for comparing two individuals as their dominance, ie, the individual with lower constrain violation will dominate the other. In case constraint violation is zero, the dominance is checked only by the value of OF.

3.4 Algorithms Complexity

It is common to use the absolute execution time (in milliseconds, seconds, minutes, etc) to measure the efficiency of an algorithm. But time is not an appropriate measure as it depends on the configuration of the computer on which it runs.

Thus, the analysis of the complexity of an algorithm is important to check its relative efficiency. The complexity is determined by the number of operations the algorithm performs for the number of input parameters. The interest is always in the worst case, when the algorithm requires the largest number of iterations (N) to reach the optimal solution.

The notation $O(\,)$ is commonly used in the complexity analysis and indicates the asymptotic behavior of an algorithm (speed at which the number of operations tends to infinity), ie when $N \rightarrow \infty$. Assuming two algorithms:

- Algorithm 1: $N^2 + 5.N$. N operations.
- Algorithm 2: N +1000 operations.

The first algorithm grows with N^2, while the second algorithm with N, so the notation used is $O(N^2)$ and $O(N)$, respectively.

The first MOEAs as the NSGA, NPGA and MOGA had complexity $O(MN^3)$, where M is the number of objectives and N the population size. The second generation that includes the state of the art in MOEAs already has lower complexity, which is the case of the NSGA-II, SPEA 2 and others. The complexity of these, in the worst case is $O(MN^2)$, and the large computational effort is associated with the non-dominated sorting procedure.

Both developed algorithms: MoDE-NS and MoPSO-NS have the same complexity of NSGA-II, ie, $O(MN^2)$. Thus it is expected that the runtime is very similar for the three algorithms, which a comparison is made in the application of test problems. Differences in processing time are related to the way how the offspring population Qt is generated. Thus, as the processing time of algorithms is proportional to the square of population size, it is desirable that it should be adopted as small as possible, remembering that the suggested value in the literature is 10 times the number of variables. This suggestion, however, should not be adopted indiscriminately, since problems with many variables would also a very large population and therefore requiring high processing time. As a practical recommendation, based on results of the test problems and case studies, it is suggested to use a population size of 2.5 to 5 times the number of variables. The higher the number of constraints, the higher the population should be.

3.5 Decision Support System for Multi-Objective Analysis

A Decision Support System was developed for application and validation of algorithms using test problems and water resource problems. The main purpose is to use the system in the management of water resources systems with the possibility of integration with decision support systems such as AcquaNet and ModSim DSS, although it can be applied to any multi-objective analysis problem in a generalized way, in various fields of engineering. Figure 3.11 a flowchart for analyzing a multi-objective problem and the main screen is shown in Figure 3.12. The optimization library consists of the algorithms described above (MoDE-NS, MoPSO-NS and NSGA-II) and is used in the optimization step, Step 5, as Figure 3.11.

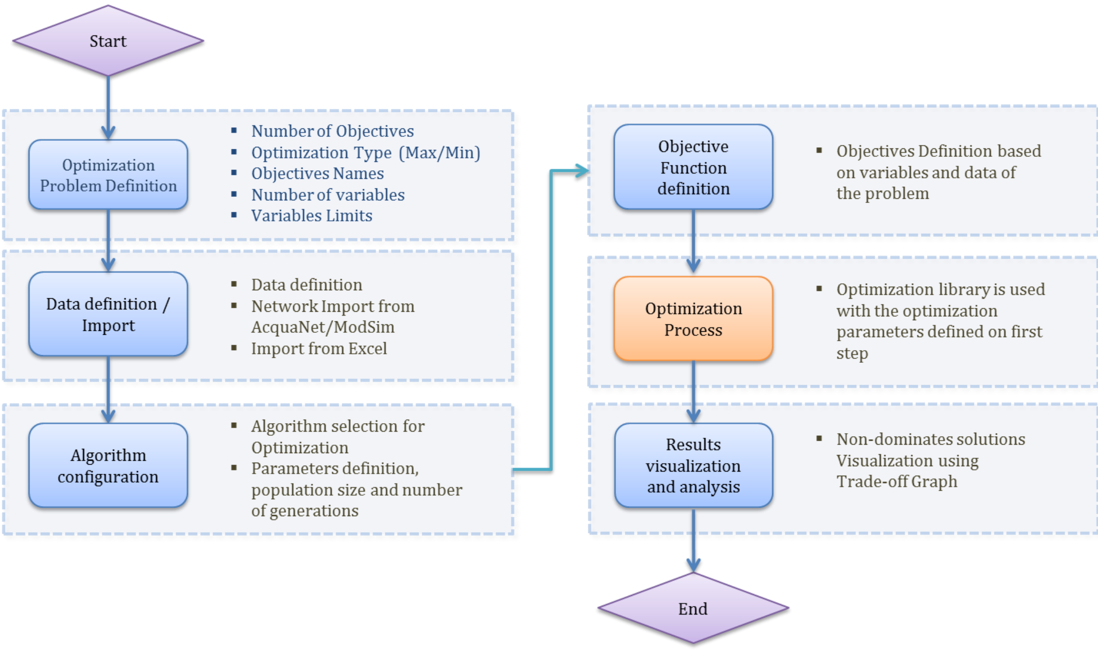


Figure 3.11 – DSS Flowchart

The system consists of two modules: (i) the algorithms library and (ii) the user interface (UI) (Figure 3.12). The algorithms library works independently of the UI, allowing it to be used and applied to other optimization or simulation systems, for example, Smap, AcquaNet DSS, ModSim DSS, and others. In addition to the graphical user interface, the system has a database manager, which allows the user to save the problems analyzed. All information relating to the problems is stored, including data and the parameters of the algorithms.

The Decision Support System and the steps for analyzing a multi-objective problem are described below.

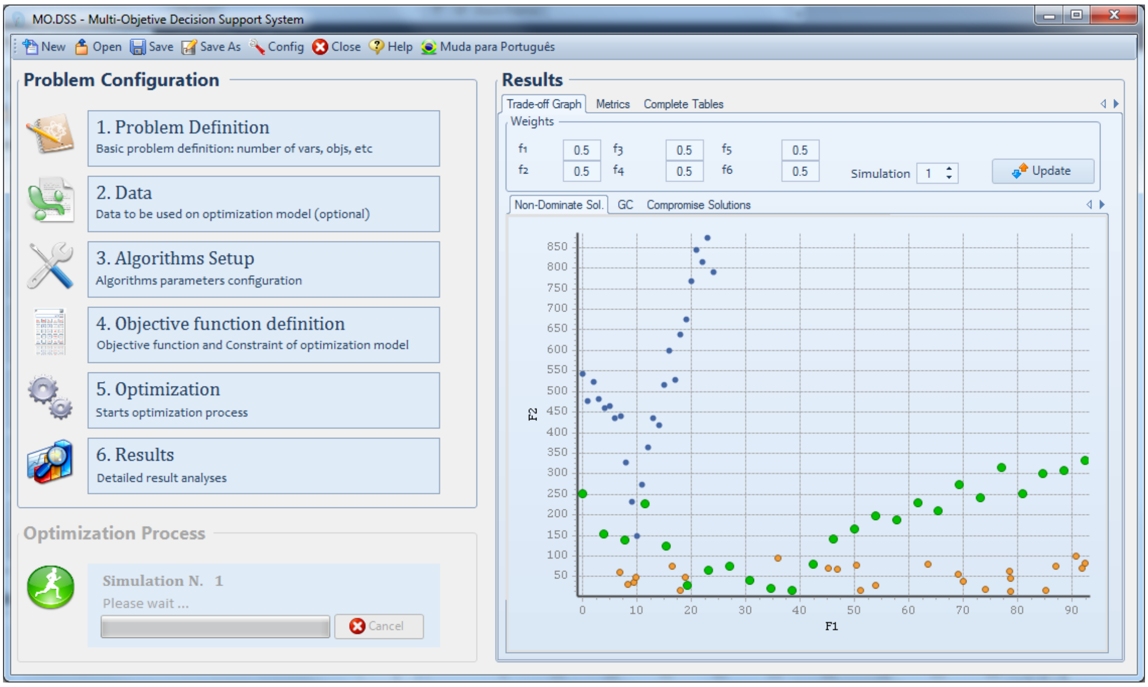


Figure 3.12 – DSS – Main Screen

1 - Problem Definition: the first step is defining the problem, and here should be provided the number of variables and their limits (upper and lower), the number of objectives, their names and the type (maximization or minimization), as Figure 3.13. This screen is also chosen optimization algorithm to be used.

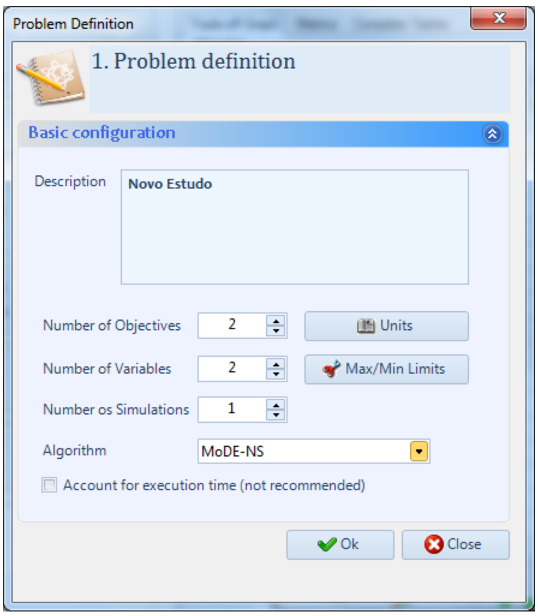


Figure 3.13 – Problem Setup

2 – Data: data can be set directly on the spreadsheets, Figure 3.14, or imported from Excel file. The data is then available for use in the definition of objective function (OF) (Step 4). It is interesting to note that any type of data can be entered, however, only the numerical data can be considered in defining the OF. On the definition of the objective function the access to data is possible by providing the index of the Table, and location of cell: row and column number, using the following convention: Tab*i*[*Row*, *Column*]. For example, in Figure 3.14 the value 808.04 is stored in the variable Tab1[7.7]. This feature gives an interesting aspect to the system, since data of any kind can be stored, including text, observations, etc.

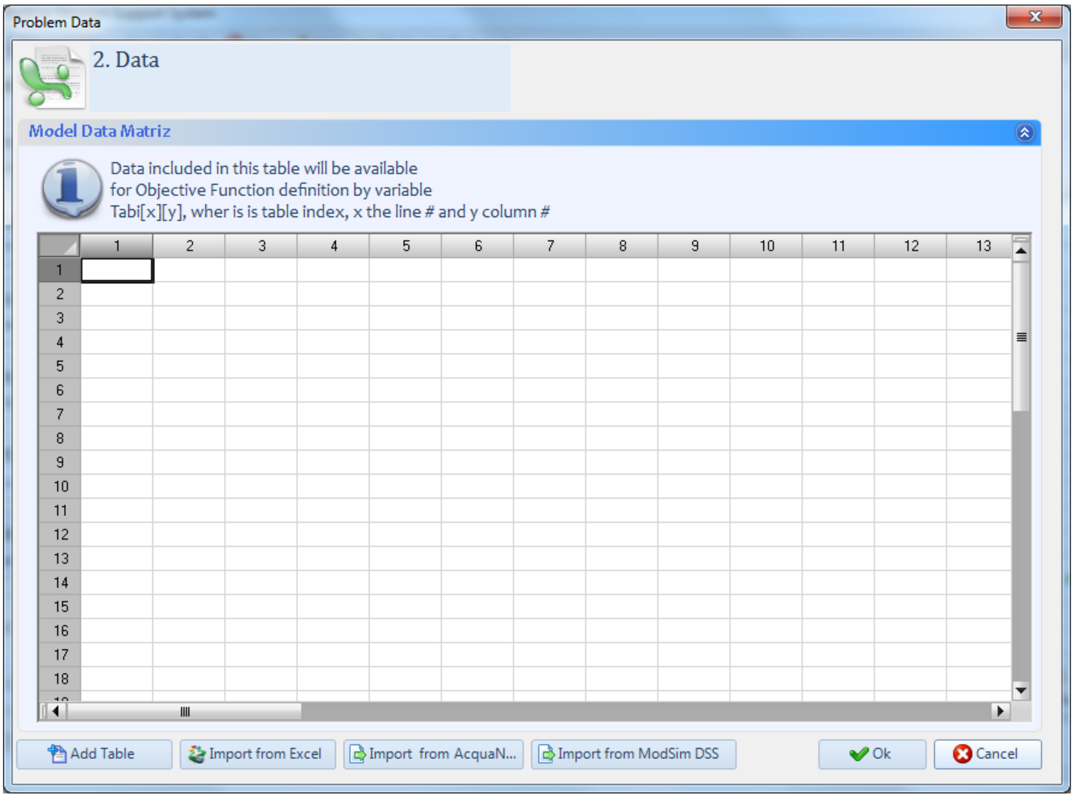


Figure 3.14 – Data setup for multi-objective analysis

3 - Algorithms parameters: in this item is set the parameters of the algorithms can be changed, as well as population size and the maximum number of generations, as in Figure 3.15. The suggested parameters are should provide a good performance of the algorithms, and

usually don't need to be changed.

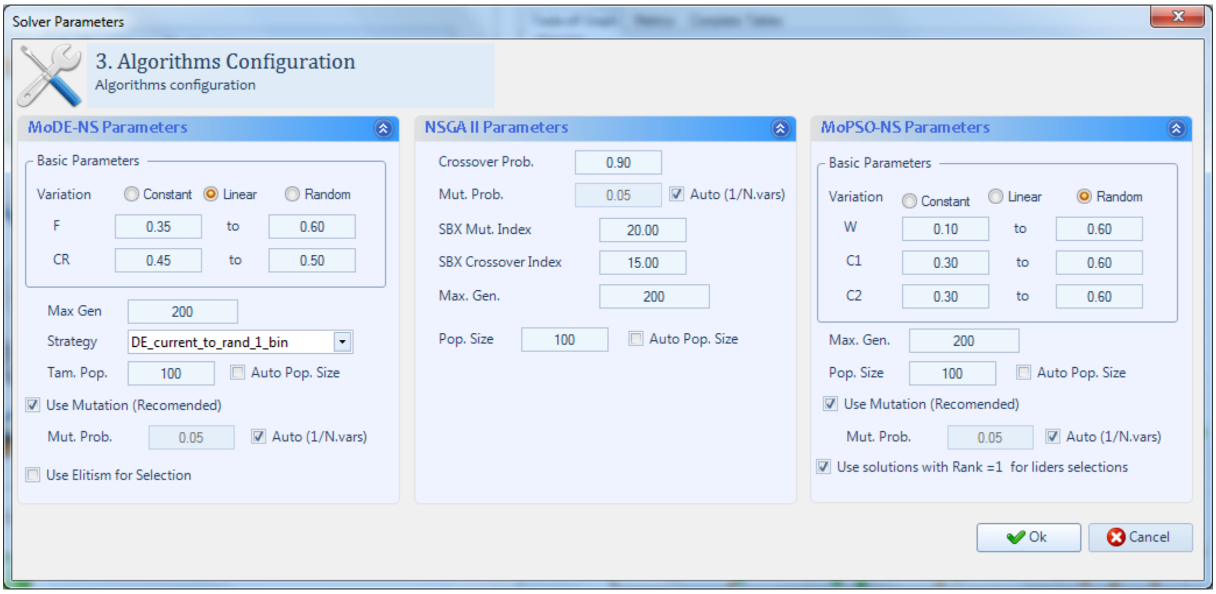


Figure 3.15 – Algorithms configuration

4 - Definition of objective function: this item defines the objective function of the problem. For this, two programming languages are available to the user: VB.Net and C #, as presented in Figure 3.17. The ability for the user to define the function freely gives the system great flexibility for analysis of multi-objective problems. In fact, users with knowledge of programming languages can build extremely complex and complete models with the use of additional classes. Unlike languages such as VBA Script, the equation of the objective functions and classes defined, are compiled (in memory) by the system before starting the optimization process. This ensures that the processing time is much smaller than with scripting languages.

In this step the data defined in Step 3 can be used. Assuming a very simple multi-objective problems as in equation (3.7) and assuming that in the data spreadsheet, was assigned the number 5 to cell with position (row = 1 column = 1). This problem, with two objectives and two variables can be easily written as Figure 3.16 and Figure 3.17, where X is the vector of variables and FO the vector of Objective Functions.

$$\text{Minimizar } f_1(x) = x_1$$

$$\text{Minimizar } f_2(x) = \frac{5 + x_2}{x_1} \tag{3.7}$$

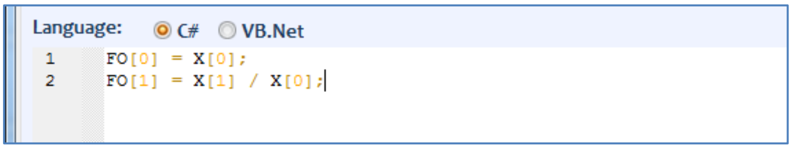


Figure 3.16 – Objective Function for problem with tow objectives

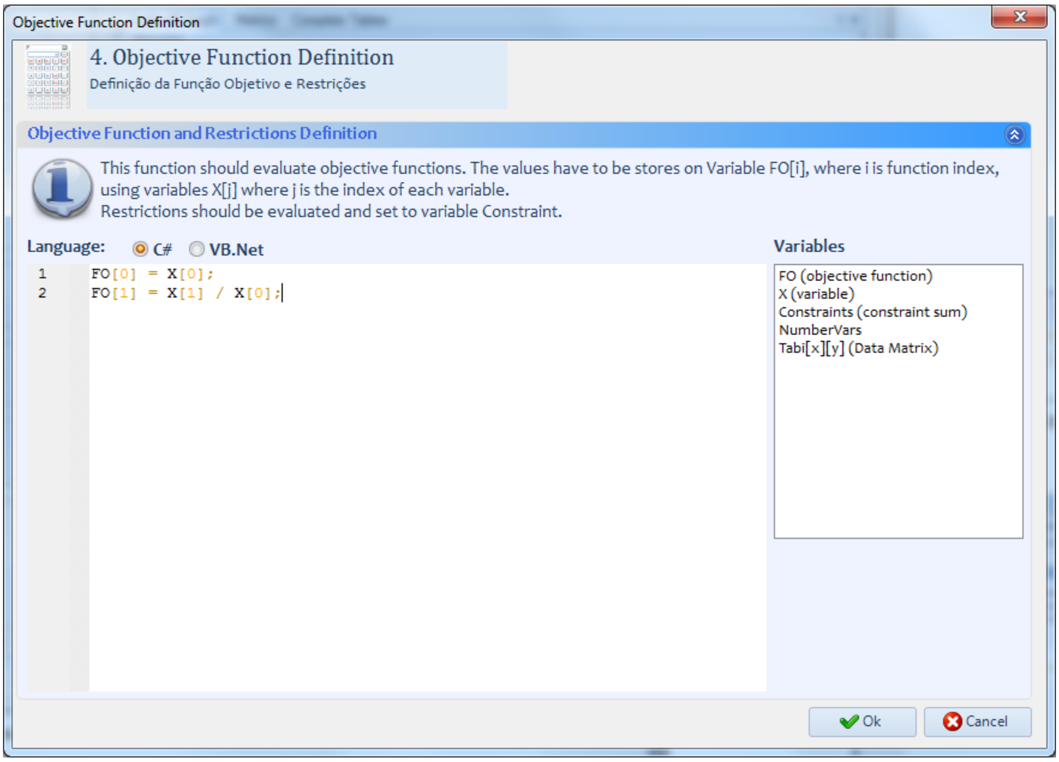


Figure 3.17 – Objective function definition

5 - Optimization: after setting up the problem, defining the number of objectives and variables, definition of objective functions and parameters of the algorithms, the optimization process can be started, and a progress window is display with the number of generations

completed and simulation in progress, as in Figure 3.18.

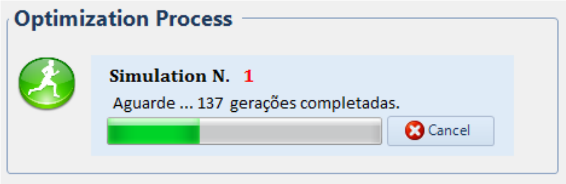


Figure 3.18 – Optimization progress

6 - Visualization and analysis of results: Data visualization can be made directly by analyzing the graph of non-dominated solutions, Figure 3.19, using the Trade-off Graph (TG) and by Tables containing complete results (objective functions and variables associated with each solution). Trade-off Graph (TG) is presented below and is an interesting way to visualize graphically the non-dominated solutions even higher number of objectives.

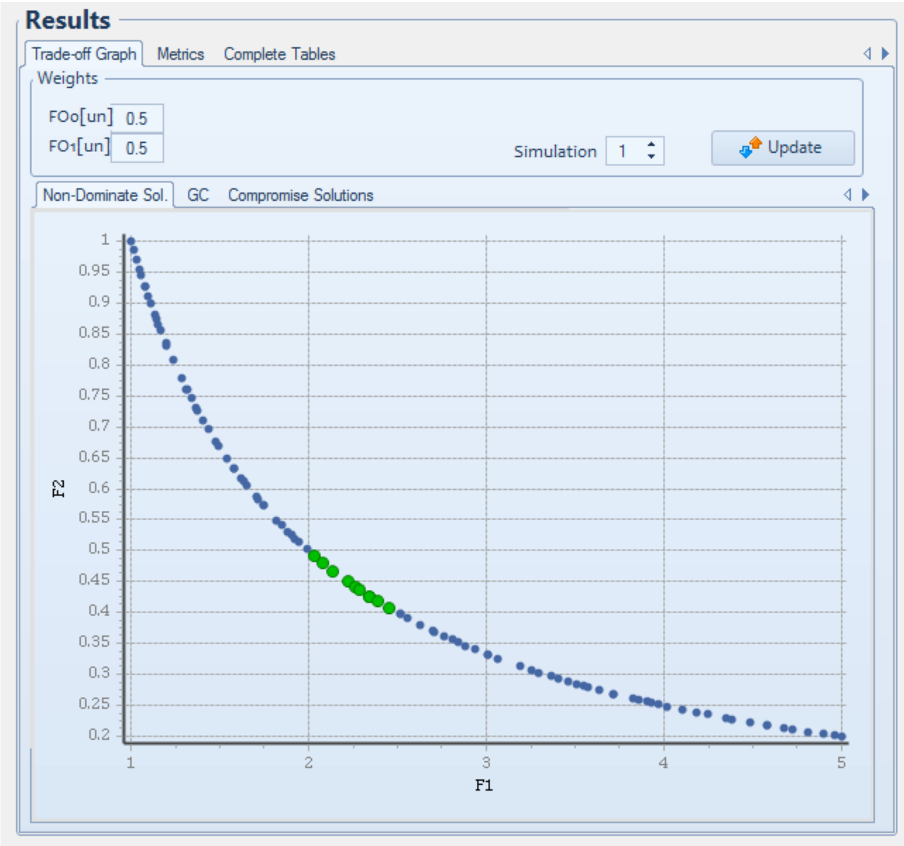


Figure 3.19 –Results visualization

Non-dominated solutions visualization – Trade off Graph - TG

For visualization of the non-dominated set of solutions a graphical tool was created called Trade-off Graph (TG), the is based on the ICC method (Interactive Compromise Coordinate Method) proposed by Baltar (2007). Although the equation presented for our TG is different from the ICC, the final result is very similar, since both methods use normalized values of the objective functions for the projection using Star Coordinate method. Baltar (2007) suggests the equation for a circle of radius = 0.5 and center at (0.5, 0.5). The equation presented below considers a circle of radius = 1 and center at (0, 0) and was based on the Star Coordinate Method (Mañas 1987 apud Deb, 2009).

In the TG, the non-dominated solution set is presented in a two-dimensional graph, regardless of the number of objectives and solutions with highest rank (top 10, based on rank provided by compromise programming method) are highlighted.

Initially all values of the objective functions are normalized in the range 0 to 2 (because the diameter of the circle is 2). The calculation, according to equation (3.8) produces a value $F(i,j)^*$ for each objective of each solution.

$$F(i,j)^* = 2 \times \frac{BestP(i) - P(i,j)}{BestP(i) - Worst(i)} \quad com \quad i = 1, \dots, M \;\; e \;\; j = 1,2, \dots N \qquad (3.8)$$

Where M is the number of objectives, N the size of te population, P the set of non-dominated solutions, BestP and WorstP are, respectively the best and worst values of objective *i*.

For the TG, it is assumed a circle of radius 1 (one), with center at coordinates (0.0, 0.0), and divided into M parts, where M is the number of objectives. For each objective an extreme point on the edge of the circle is calculated which corresponds to the maximum normalized value of that objective. The radial lines that connect the center to the extreme points are separated by an angle θ calculated according to the number of objectives. The angles and the coordinates xF, yF are calculated according to equation (3.9). Taking an example of a problem with three objectives, it is possible to calculate the extreme points as presented in Table 3.5, and the graph of Figure 3.20.

$$\theta = \frac{2.\pi}{M}$$

$$xF_i = \cos\left(\frac{\pi}{2} - \theta \times (i-1)\right) \;\; com \;\; i = 1,2,\ldots,M \qquad (3.9)$$

$$yF_i = \text{sen}\left(\frac{\pi}{2} - \theta \times (i-1)\right) \;\; com \;\; i = 1,2,\ldots,M$$

Where M is the number of objectives

Table 3.5 – Objective Function coordinates for 3 Objectives

Objective	**i**	$\boldsymbol{xF_i}$	$\boldsymbol{yF_i}$
f1	0	0.00	1.00
f2	1	0.87	-0.50
f3	2	-0.87	-0.50

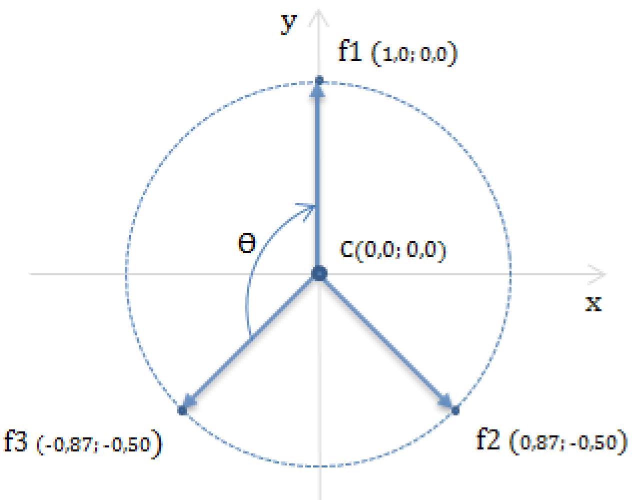


Figure 3.20 – Schema for circle used to project non-dominated solutions

The next step is to determine the position of each point corresponding to the objectives of each solution. Thus for every objective of all solutions a pair of values (x, y) will be obtained and that will generate a polygon. These pairs of values are calculated according to equation (3.10).

$$x(i,j) = -xF_i \times F(i,j)^* \;\; com \;\; i = 1,2,\dots,M \;\; j = 1,2,\dots,N$$
$$y(i,j) = -yF_i \times F(i,j)^* \;\; com \;\; i = 1,2,\dots,M \;\; j = 1,2,\dots,N \tag{3.10}$$

where M is the number of objectives and N is the population size.

From the pairs of values of x, y calculated from equation (3.10), which form a polygon of M vertices, we can calculate the coordinates of the centroid xP, yP as equation (3.11) for each non-dominated solution. A schematic example is shown in Figure 3.21.

$$A(j) = \frac{1}{2}\left|\sum_{i=1}^{M-1}[x(i,j) \times y(i+1,j) - x(i+1,j) \times y(i,j)]\right| \;\; with \;\; i = 1,2,\dots,M \;\; j = 1,2,\dots,N$$
$$xP(j) = \frac{1}{M.A}\sum_{i=1}^{M-1}[x(i,j) + x(i,j+1)][x(i,j) \times y(i,j+1) - x(i,j+1) \times y(i,j)] \;\; with \;\; i = 1,2,\dots,M \;\; j = 1,2,\dots,N \tag{3.11}$$
$$yP(j) = \frac{1}{M.A}\sum_{i=1}^{M-1}[y(i,j) + y(i,j+1)][x(i,j) \times y(i,j+1) - x(i,j+1) \times y(i,j)] \;\; whit \;\; i = 1,2,\dots,M \;\; j = 1,2,\dots,N$$

Where M is the number of objectives and N the population size, A is the area, x and y the coordinates of the points of the polygon, .xP and yP the centroid coordinates.

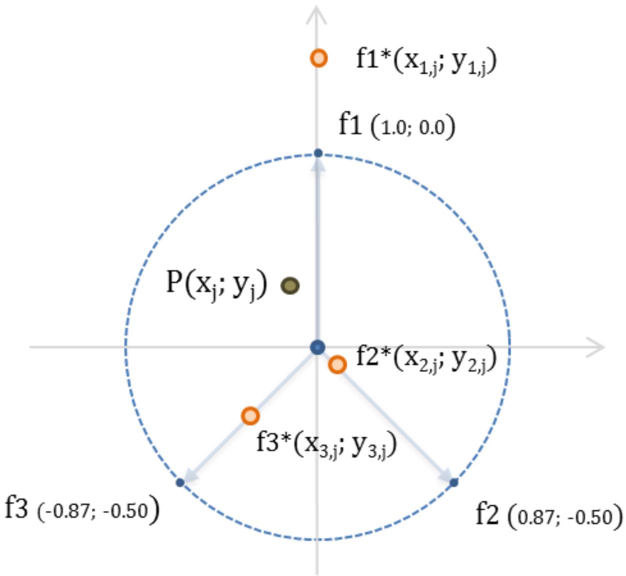


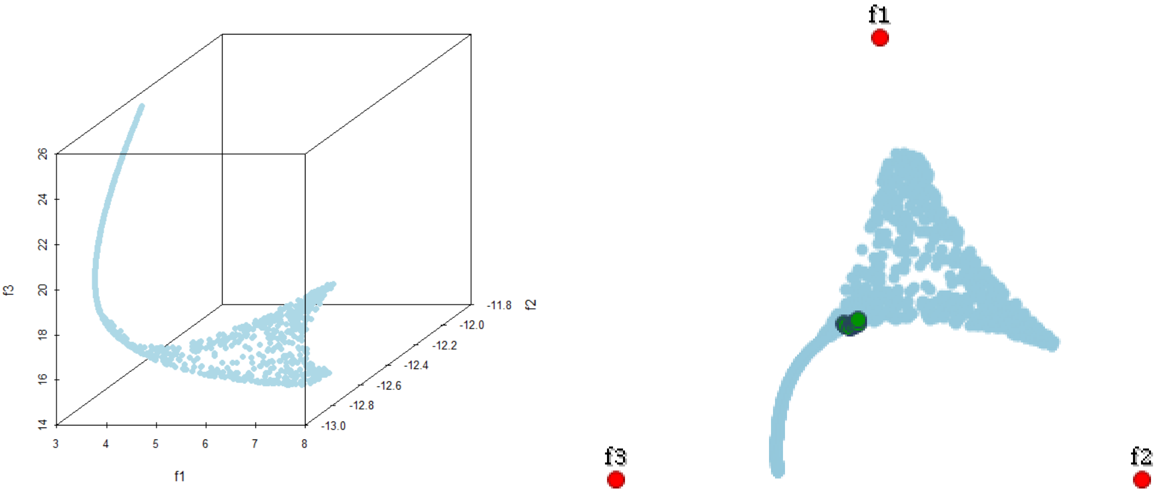
Figure 3.21 – Schema for centroid calculation for a non-dominated solution

An example of the TG is shown in Figure 3.22, where in (a) is presented a 3D-graph with wit the set of non-dominated solutions plotted in 3 dimensions (3 objective problem) and (b) presents the graph according to the proposed methodology.

After calculating the positions of each solution, Compromise Programming is applied to define the best solutions (or most robust) based on weights (α) provided by the user. Initially the value of Lp as equation (2.12) is calculated for each solution using the values of p = 1, 2 and 100, that is, each solution is assigned a value $L_1, L_2\ abd\ L_{100}$ that defines a ranking $R_1, R_2\ and\ R_{100}$ from 1 to N (population size). Finally the average of Ri defines the final ranking of each solution, also from 1 to N. Once classified, the 10 best solutions are highlighted in the TG in another color (green), according to the example of Figure 3.22 and Figure 3.23. In this example the values of the weights used for each objective was 0.5.

$$L_p(j) = \sum_{i=1}^{M} \alpha_i^p \left(\frac{z_i(j) - z_i^{**}}{z_i^* - z_i^{**}} \right)^p, \ \ com \ \ p = 1{,}2\ e\ 100; \ \ j = 1{,}2, \dots, N \ \ e \ \ i = 1{,}2, \dots, M \qquad (3.12)$$

Where M is the number of objective, z_i^* the best value for objective i and z_i^{**} the worst value for objective i, N the population size.



(a) (b)

Figure 3.22 – Example of Trade-off Graph (TG) usage

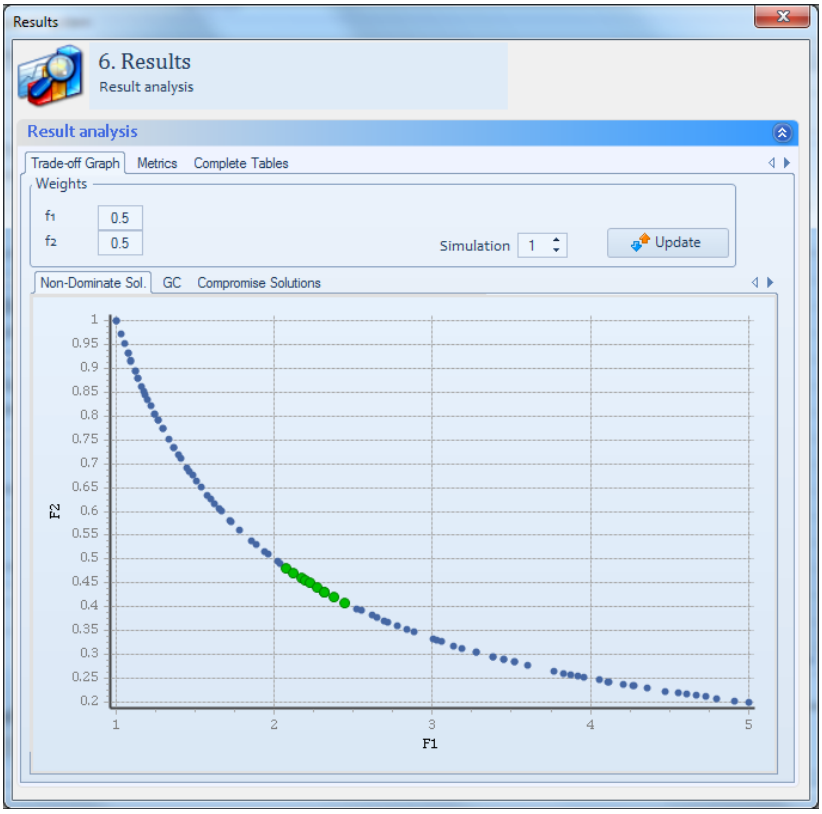


Figure 3.23 – Result screen of the developed DSS

3.6 DSS Integration with AcquaNet and ModSim

As described above, AcquaNet and ModSim DSS are very efficient to solve linear problems that can be represented as networkflow. Therefore, it is natural to consider an integrated analysis between MOEAs and networkflow algorithms. Another interesting feature of both DSSs is that they are powered by and user interface that allows user to easily build a network using graphical elements that represent Reservoirs, Demands, tunnels, channels, rivers.

There are two ways to integrate the proposed DSS to AcquaNet and ModSim:

1. Using the AcquaNet or ModSim as a tool to aid user or Decision Maker (DM) to build the network flow that represents the water resource problems and then importing it in the DSS to define the multi-objective model. AcquaNet and ModSim store the network in a database, which can be then imported into the DSS and all information will available in the datasheets. Each network element is imported as a Table in the datasheets. The user can thus use the information and define the objective functions using the data imported from AcquaNet and ModSim. The Figure 3.24 shows the flowchart of the suggested integration of AcquaNet / ModSim with the DSS described in this report, by importing the network flow.
2. Using AcquaNet/ModSim for the simulation / optimization process: the DSS engine makes use of the result of the network flow algorithm Out-of-Kilter of Relax IV to evaluate of the objective functions of the multi-objective model. An example of this application is presented in the Case study 3, with details of their limitations, advantages and disadvantages.

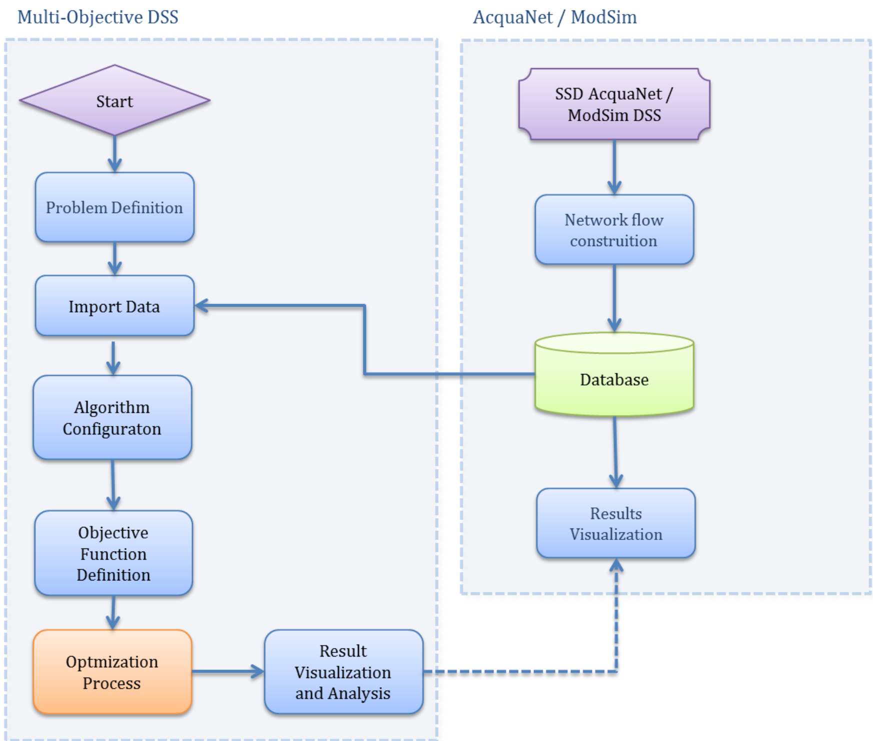


Figure 3.24 – Flowchart of Integration between the DSS and AcquaNet / ModSim DSS

4 Application and Results

The application of the developed algorithms (MoDE-NS and MoPSO-NS) and NSGA-II was made on standard test problems for verification and validation. Five common tests problem were used: Kursawe, Kita, ZDT1, DTLZ5 Viennet and 4, described below. These tests are non-linear problems with non-convex and non-continuous optimal Pareto front.

The algorithms are also applied to two problems related to water resources, one with restriction only on the decision variables (calibration of hydrological rainfall-runoff model, Smap) and an optimization problem of a series of reservoirs that are Cantareira System, responsible for part of the water supply for the city of São Paulo.

The application to Smap was performed with 2 and 5 objectives for performance verification of algorithms, especially with larger number of objectives, and the use the Trade-off Graph.

In Cantareira System three different cases were analysed. Initially, an analysis was done using only MOEAs with two pairs of conflicting objective functions. Then the application was made to the same system, but using the integration MOEAs and networkflow algorithms. Finally in Case 4, an application was made considering only one of the reservoirs and using 3 objectives. The values of the parameters for the algorithms used in all applications of this chapter are presented in Table 4.1. The population size and generation number is indicated in each case analyzed.

Table 4.1 – Parameters used on MOEAs for Test Problems

Solver	**Parameter**	**Value**
MoDE-NS	Strategy	DE/current-to-rand/1/bin
	Cr	0.40 to 0.60
	F	0.80 to 1.00
	PMNu	0.01
	Elitism Selection	No
MoPSO-NS	W	0.40 – 0.10
	c1	0.30 – 0.60
	c2	0.50 – 0.80
	PMNu	0.05
	Leader selection Elitism	Yes
NSGA-II	Mutation Probability	0.90
	Crossover Probability	0.05
	SBX Mutation Index	20
	SBX Crossover Index	20

4.1 Test problems application

For the test problems 10 simulations were performed with population size of 200 individuals and 250 generations. In test Viennet 4 a population of 1,000 individuals and 1,000 generations where used, so that the non-dominated set of solutions have better coverage of the Pareto-optimal front. The results are compared with those available in the literature for verification of non-dominated solutions obtained and the validation of algorithms. Of the five problems, three of them have two objectives (Kursawe, Kita and ZDTL1) and the other two, three objectives (DTLZ5 and Viennet 4).

4.1.1 Test 1 – Kursawe's Test

This problem was proposed by Kursawe (1996) and was used by authors such as Deb et al. (2002), Baltar (2007), Coello Coello et al. (2007), among others.

This problem, presented in equation (4.1)has three decision variables and two objective functions, with the variables in the allowed range of -5 to 5, with no other restrictions. The search space of this problem and is non-convex, and Pareto-optimal front is discontinuous, having three sets of non-dominated solutions.

$$Minimize\ f_1(x_1, x_2, x_3) = \sum_{i=1}^{2}\left(-10exp\left(-0.2\sqrt{x_i^2 + x_{i+1}^2}\right)\right)$$
$$Minimize\ f_2(x_1, x_2, x_3) = \sum_{i=1}^{3}\left(|x_i|^{0.8} + 5sen(x_i^3)\right) \quad (4.1)$$

Where n = 3 subject to:

$$-5 \leq\ x_1, x_2, x_3 \leq 5$$

The test results are presented in Figure 4.1, Figure 4.2 and Figure 4.3, for the algorithms NSGA-II, MoDE-NS and MoPSO, respectively. These graphs present all non-dominated solutions obtained for 10 simulations. The graphical comparison shows that the algorithms NSGA-II and MoDE-NS, was performing better than MoPSO-NS. Performance indicators are presented in Table 4.2 and indicate that the MoDE-NS had better overall performance, although the results are very close.

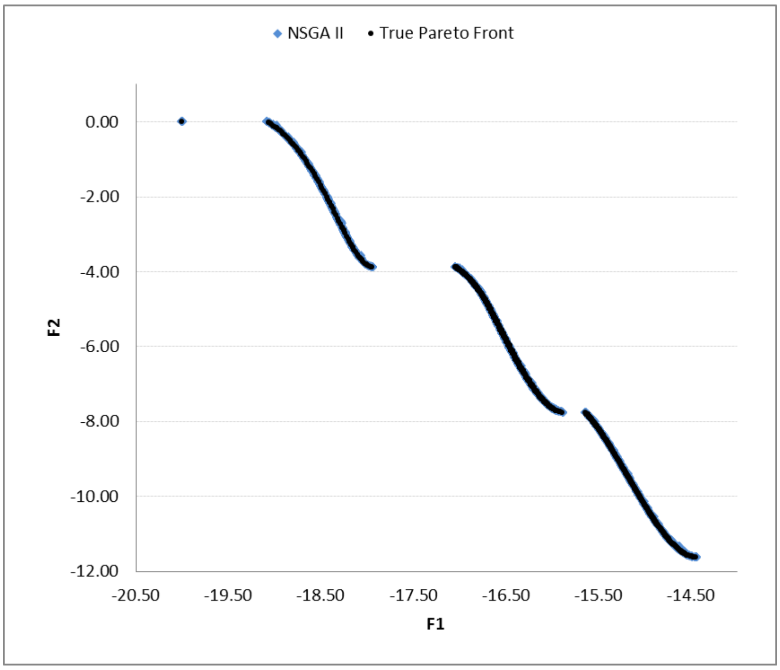


Figure 4.1 – Result for Kursawe's Test – NSGA-II

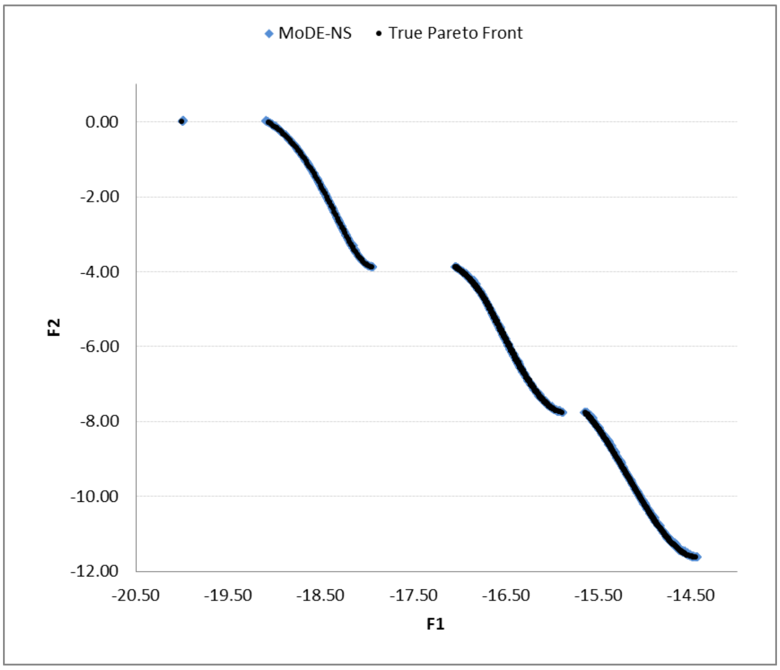


Figure 4.2 – Result for Kursawe's Test – MoDE-NS

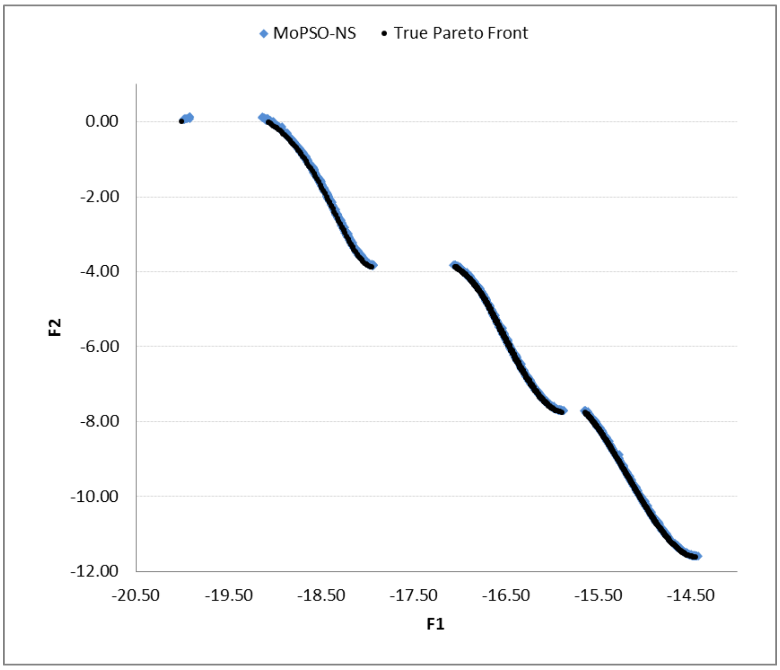


Figure 4.3 – Result for Kursawe's Test – MoPSO-NS

Table 4.2 – Metrics for Kursawe's test problem

Solver	Metric	GD	iGD	ID	DD	SP	Time (ms)
MoDE-NS	Mean	0.00094	0.00087	0.517	0.340	0.094	675

Solver	Metric	GD	iGD	ID	DD	SP	Time (ms)
	Median	0.00094	0.00088	0.519	0.328	0.095	671
	Max	0.00100	0.00090	0.538	0.450	0.106	686
	Min	0.00089	0.00083	0.487	0.280	0.083	655
	DP	0.00004	0.00002	0.014	0.054	0.007	10
	Mean	0.00119	0.00081	0.415	0.559	0.049	596
	Median	0.00121	0.00080	0.420	0.563	0.048	593
MoPSO-NS	Max	0.00130	0.00084	0.435	0.635	0.057	608
	Min	0.00097	0.00078	0.386	0.460	0.042	577
	DP	0.00011	0.00002	0.014	0.047	0.005	10
	Mean	0.00090	0.00091	0.545	0.321	0.096	785
	Median	0.00092	0.00090	0.543	0.318	0.095	780
NSGA-II	Max	0.00098	0.00104	0.563	0.380	0.104	796
	Min	0.00080	0.00085	0.524	0.255	0.089	780
	DP	0.00006	0.00005	0.012	0.039	0.005	8

DP is the Standard deviation, ms is time in milliseconds

4.1.2 Test 2 - Kita's Test

This problem was proposed by Kita et al. (1996) and is shown in equation (4.2), has two decision variables and two objective functions, with the variables in the allowed range of 0 to 30, and three additional restrictions. This test has been used by Coello Coello et al. (2007).

$$Maximize\ f_1(x_1, x_2) = -x_1^2 + x_2$$

$$Maximize\ f_2(x_1, x_2) = \frac{1}{2}x_1 + x_2$$

Subject to:

$$\frac{1}{6}x_1 + x_2 - \frac{13}{2} \leq 0 \tag{4.2}$$

$$\frac{1}{2}x_1 + x_2 - \frac{15}{2} \leq 0$$

$$5x_1 + x_2 - 30 \leq 0$$

$$0 \leq x_1, x_2 \leq 30$$

The test results are presented in the graphs in Figure 4.4, Figure 4.5 and Figure 4.6 for the algorithms MoDE-NS, MoPSO-NS and NSGA-II, respectively. These graphs present all non-dominated solutions obtained for 10 simulations. This problem has three constraints, which are common in the water resources problems. Thus, it is likely that an algorithm with good performance in this test, you will have good results in more complex systems analysis. In this test the MoDE-NS and MoPSO-NS outperformed NSGA-II, as metrics presented in Table 4.3. The MoDE-NS also performed better on two metrics (GD, DD) while the MoPSO-NS was better in the other three.

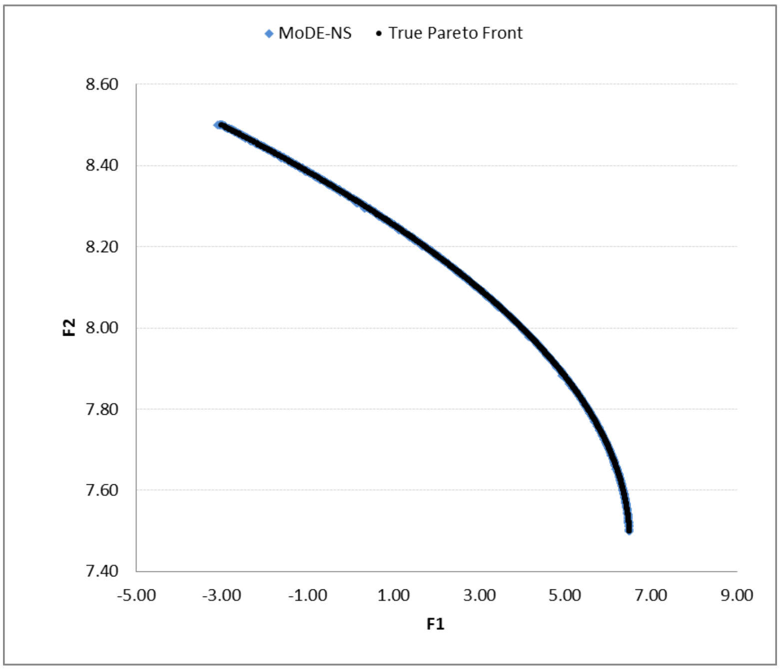


Figure 4.4 – Non-dominated solutions for Kita's test problem – MoDE-NS

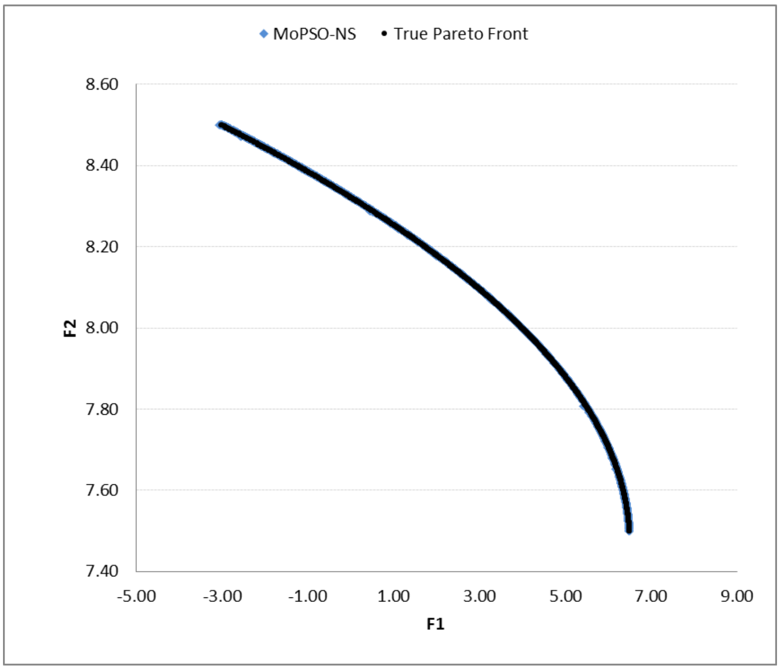


Figure 4.5 – Non-dominated solutions for Kita's test problem – MoPSO-NS

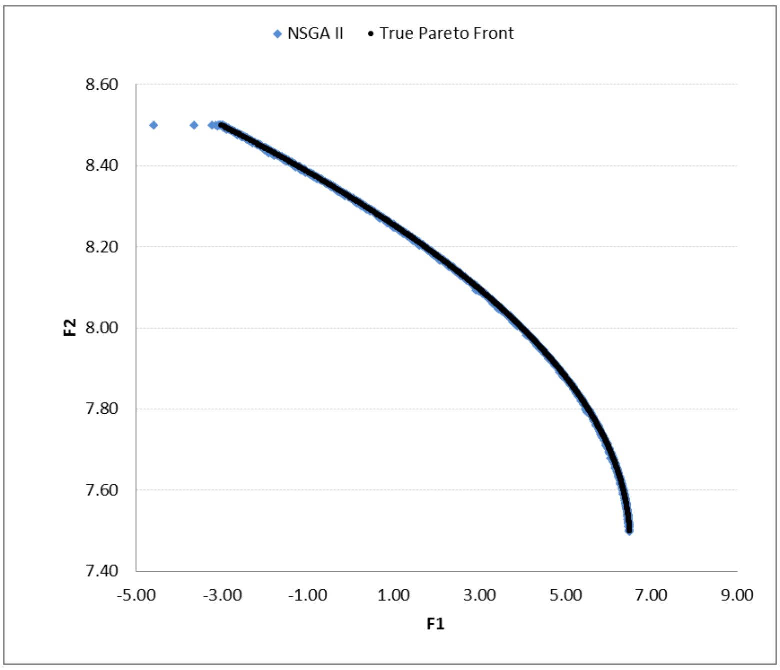


Figure 4.6 – Non-dominated solutions for Kita's test problem – NSGA-II

Table 4.3 – Metrics for Kita's test problems

Solver	Metric	GD	iGD	ID	DD	SP	Time (ms)
MoDE-NS	Average	0.00048	0.00071	0.617	0.167	0.103	672
	Median	0.00047	0.00071	0.613	0.165	0.103	671
	Max.	0.00064	0.00074	0.664	0.260	0.112	686
	Min.	0.00036	0.00068	0.580	0.125	0.093	655
	S.D.	0.00008	0.00002	0.027	0.037	0.005	11
MoPSO-NS	Average	0.00053	0.00061	0.425	0.312	0.053	733
	Median	0.00045	0.00061	0.420	0.310	0.053	733
	Max.	0.00132	0.00067	0.475	0.370	0.058	749
	Min.	0.00040	0.00056	0.389	0.260	0.046	718
	S.D.	0.00028	0.00003	0.023	0.036	0.003	7
NSGA-II	Average	0.00213	0.00079	0.728	0.180	0.101	660
	Median	0.00074	0.00079	0.722	0.180	0.101	655
	Max.	0.00972	0.00084	0.802	0.235	0.106	671
	Min.	0.00045	0.00073	0.671	0.125	0.098	655

Solver	Metric	GD	iGD	ID	DD	SP	Time (ms)
	S.D.	0.00294	0.00003	0.038	0.040	0.003	8

DP is the Standard deviation, ms is time in milliseconds

4.1.3 Test 3 – ZDT1's Test

This problem was proposed by Zitzer et al (2001) and is presented in equation (4.3) has thirty decision variables and two objective functions, with the variables in range from 0 to 1, without further restrictions.

$$Minimize\ f_1(x) = x_1$$

$$Minimize\ f_2(x) = g(x).\left(1 - \sqrt{\frac{f_1(x)}{g(x)}}\right)$$

$$g(x) = 1{,}0 + \frac{9}{n-1}\sum_{i=2}^{n} x_i \qquad (4.3)$$

Subject to:

$$0 \leq\ x_i \leq 1$$

$$n = 30$$

$$i = 1,2, \ldots ,30$$

The test results are presented graphically in Figure 4.7, Figure 4.8 e Figure 4.9, for MoDE-NS, MoPSO-NS and NSGA-II algorithms, respectively. The results show a good fit of the three algorithms, however, as indicators of Table 4.4 the worst performance in this test was for MoDE-NS, and MoPSO-NS and NSGA-II have almost the same performance.

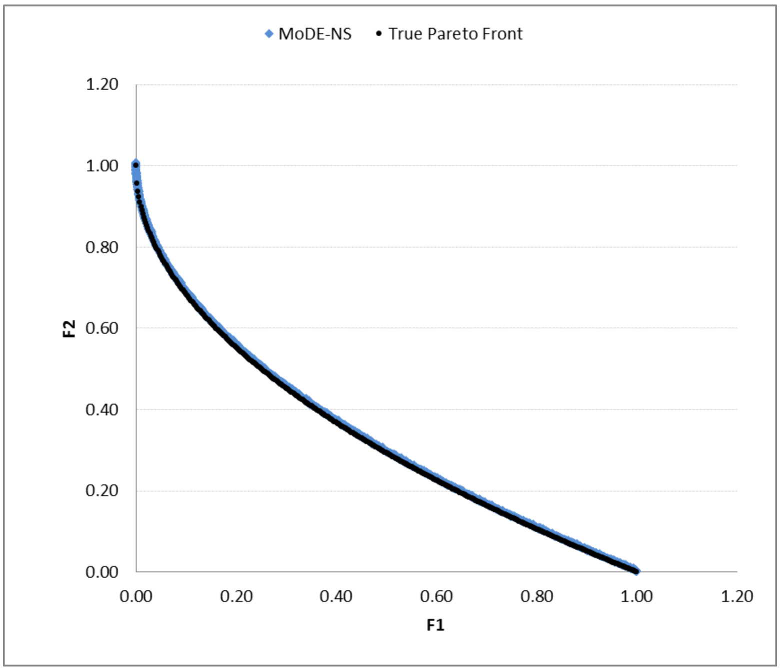


Figure 4.7 – Non-dominated solutions for ZDT1's test problem – MoDE-NS

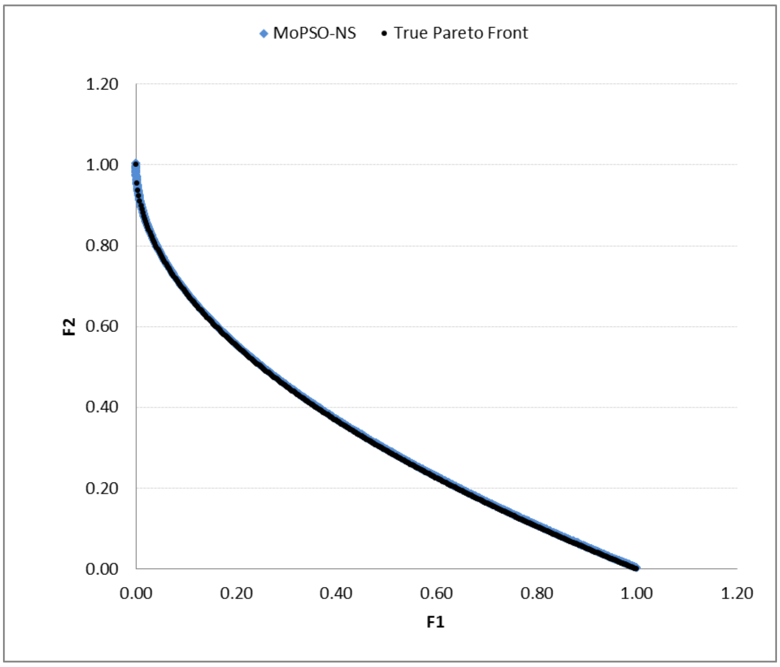


Figure 4.8 – Non-dominated solutions for ZDT1's test problem – MoPSO-NS

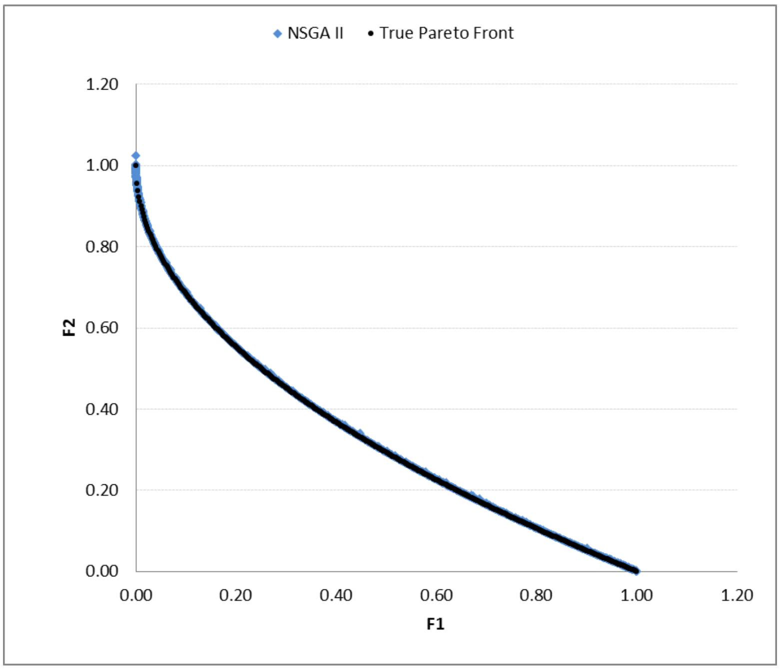


Figure 4.9 – Non-dominated solutions for ZDT1's test problem – NSGA-II

Table 4.4 – Metrics for ZDT1's test problem

Solver	Metric	GD	iGD	ID	DD	SP	Time (ms)
MoDE-NS	Average	0.00027	0.00019	0.302	1.000	0.002	866
	Median	0.00037	0.00023	0.335	1.000	0.002	887
	Max.	0.00024	0.00016	0.261	1.000	0.002	857
	Min.	0.00004	0.00002	0.025	0.000	0.000	10
	S.D.	0.00000	0.00000	0.000	0.000	0.000	0
MoPSO-NS	Average	0.00017	0.00014	0.376	1.000	0.002	1058
	Median	0.00020	0.00016	0.404	1.000	0.003	1145
	Max.	0.00016	0.00013	0.341	1.000	0.002	1029
	Min.	0.00001	0.00001	0.019	0.000	0.000	34
	S.D.	0.00000	0.00000	0.000	0.000	0.000	0
NSGA-II	Average	0.00017	0.00013	0.359	1.000	0.003	972
	Median	0.00023	0.00014	0.390	1.000	0.003	985
	Max.	0.00013	0.00012	0.329	1.000	0.002	963
	Min.	0.00002	0.00000	0.019	0.000	0.000	7

Solver	Metric	GD	iGD	ID	DD	SP	Time (ms)
	S.D.	0.00000	0.00000	0.000	0.000	0.000	0

DP is the Standard deviation, ms is time in milliseconds

4.1.4 Test 4 – DTLZ5

This problem was proposed by Deb et al. (2002) and is presented in equation (4.4), has 12 decision variables and three objective functions, with the variables in the allowed range from 0 to 1, without further restrictions.

$$Minimize\ f_1(x) = cos(\theta_1)cos(\theta_1)(1 + g(x))$$

$$Minimize\ f_2(x) = cos(\theta_1)sin(\theta_2)(1 + g(x))$$

$$Minimize\ f_3(x) = sin(\theta_1)(1 + g(x))$$

$$\theta_1 = \frac{\pi}{2} x_1$$

$$\theta_2 = \frac{\pi}{4 \times \left(1 + g(x)\right)}(1 + 2x_2 + g(x)) \tag{4.4}$$

$$g(x) = \sum_{i=3}^{n} (x_i - 0.5)^2$$

Subject to:

$$0 \leq x_i \leq 1$$

$$n = 12$$

$$i = 1,2, \ldots ,12$$

The results are presented graphically in Figure 4.10, Figure 4.11 e Figure 4.12, for algorithms MoDE-NS, MoPSO-NS and NSGA-II, respectively. Performance metrics are presented in Table 4.5, and it is possible to note that the MoDE-NS performs slightly better than the other two algorithms.

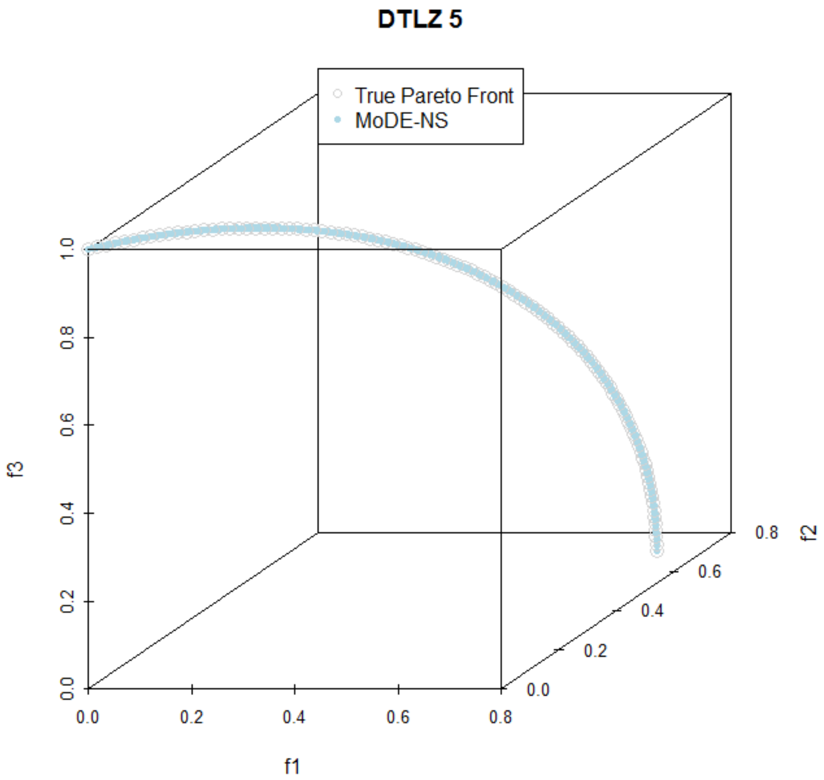


Figure 4.10 – Non-dominated solutions for DTLZ5's test problem – MoDE-NS

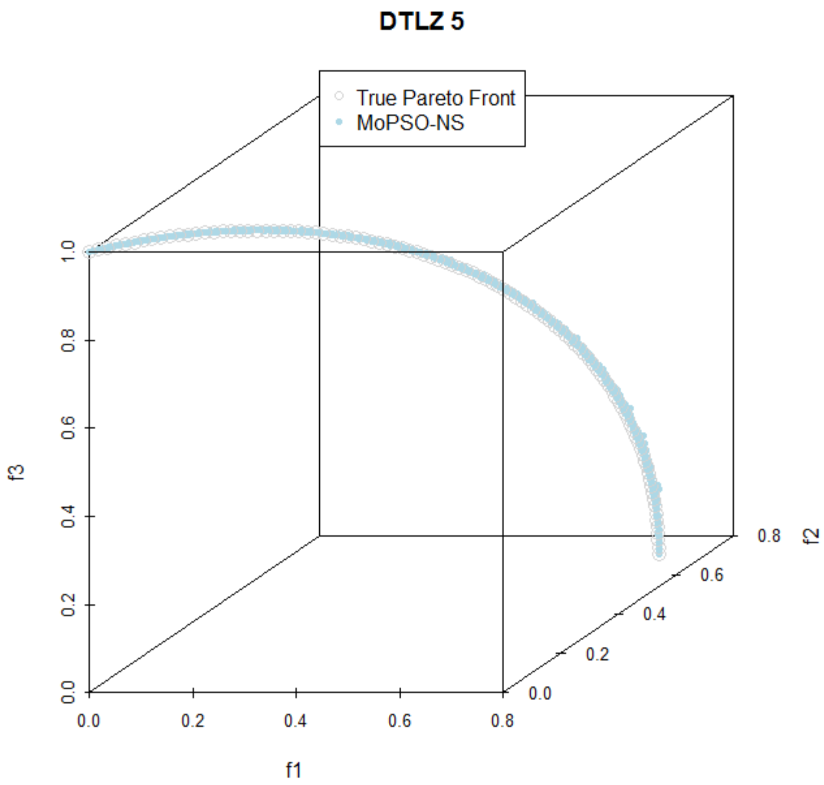


Figure 4.11 – Non-dominated solutions for DTLZ5's test problem – MoPSO-NS

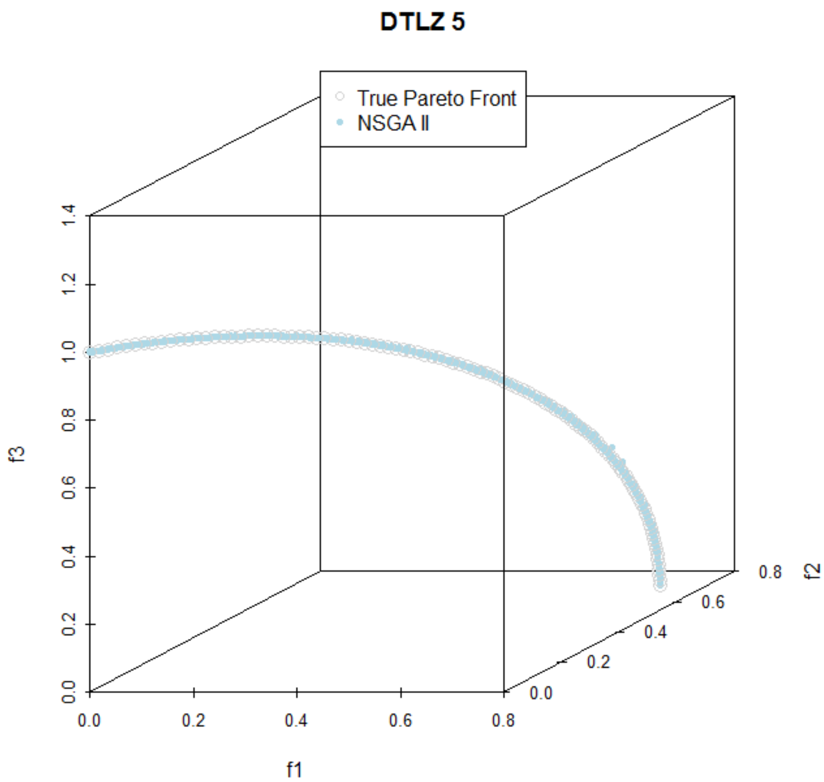


Figure 4.12 – Non-dominated solutions for DTLZ5's test problem – NSGA-II

Table 4.5 – Metrics for DTLZ5's test problem

Solver	Metric	GD	iGD	ID	DD	SP	Time (ms)
MoDE-NS	Average	0.00032	0.00034	--	0.098	0.033	1132
	Median	0.00031	0.00035	--	0.098	0.033	1132
	Max.	0.00033	0.00038	--	0.145	0.036	1148
	Min.	0.00031	0.00030	--	0.045	0.031	1118
	S.D.	0.00001	0.00003	--	0.030	0.002	10
MoPSO-NS	Average	0.00035	0.00038	--	0.188	0.024	1031
	Median	0.00035	0.00038	--	0.188	0.025	1029
	Max.	0.00036	0.00040	--	0.245	0.027	1063
	Min.	0.00032	0.00035	--	0.125	0.021	1006
	S.D.	0.00001	0.00002	--	0.036	0.002	16
NSGA-II	Average	0.00033	0.00035	--	0.042	0.035	1146
	Median	0.00033	0.00034	--	0.040	0.035	1146
	Max.	0.00035	0.00042	--	0.080	0.038	1158
	Min.	0.00031	0.00030	--	0.025	0.033	1129
	S.D.	0.00001	0.00003	--	0.017	0.002	8

DP is the Standard deviation, ms is time in milliseconds

4.1.5 Test 5 – Viennet 4's Test

This problem was proposed by Viennet et al. (1995) and is shown in equation (4.5) has two decision variables and three objective functions, with four constraints.

$$Minimize\ f_1(x_1, x_2) = \frac{(x_1 - 2)^2}{2} + \frac{(x_2 - 1)^2}{13} + 3$$

$$Minimize\ f_2(x_1, x_2) = \frac{(x_1 + x_2 - 3)^2}{175} + \frac{(2x_2 - x_1)^2}{17} - 13$$

$$Minimize\ f_3(x_1, x_2) = \frac{(3x_1 - 2x_2 + 4)^2}{8} + \frac{(x_1 - x_2 + 1)^2}{27} + 15$$

(4.5)

Subject to:

$$-4 \leq x_1, x_2 \leq 4$$

$$x_2 < 4x_1 + 4$$

$$x_1 > -1$$

$$x_2 > x_1 - 2$$

The results are presented graphically in Figure 4.13, Figure 4.14 and Figure 4.15, for algorithms MoDE-NS, MoPSO-NS and NSGA-II, respectively. In these graphs are presented all non-dominated solutions obtained from 10 simulations. The three algorithms have very similar performance as results of metrics shown in Table 4.6. In this test a larger population (1,000) was used in order to better represent the set of non-dominated solutions. The number of generations (1000) used was also higher than on the other test problems.

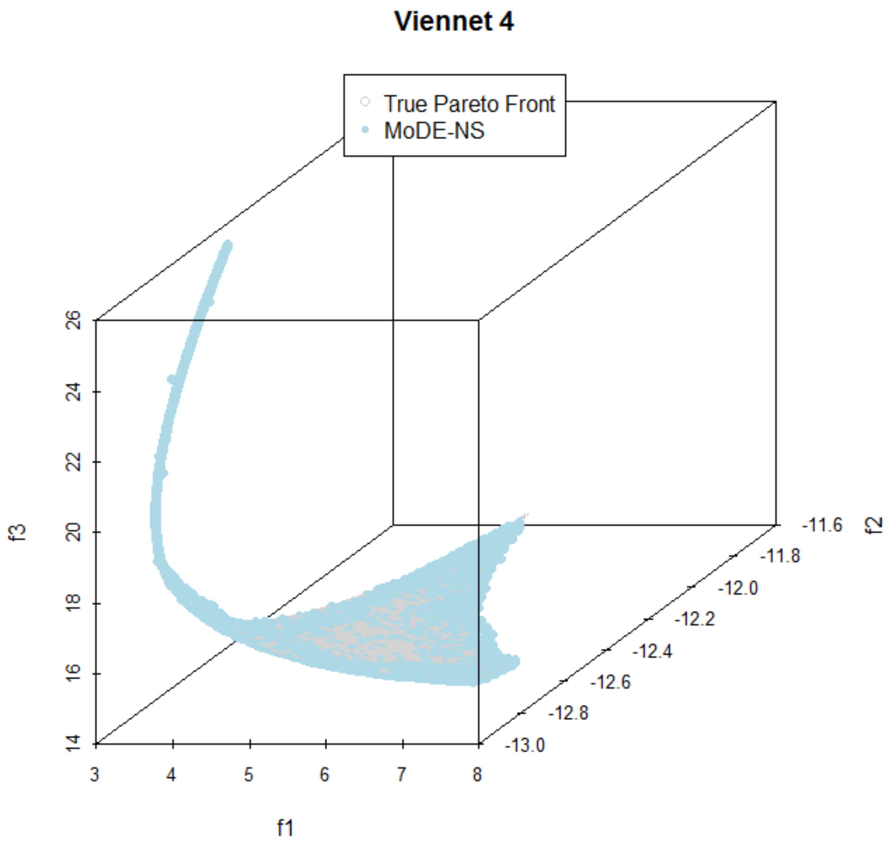


Figure 4.13 – Result do Teste de Viennet 4 – MoDE-NS

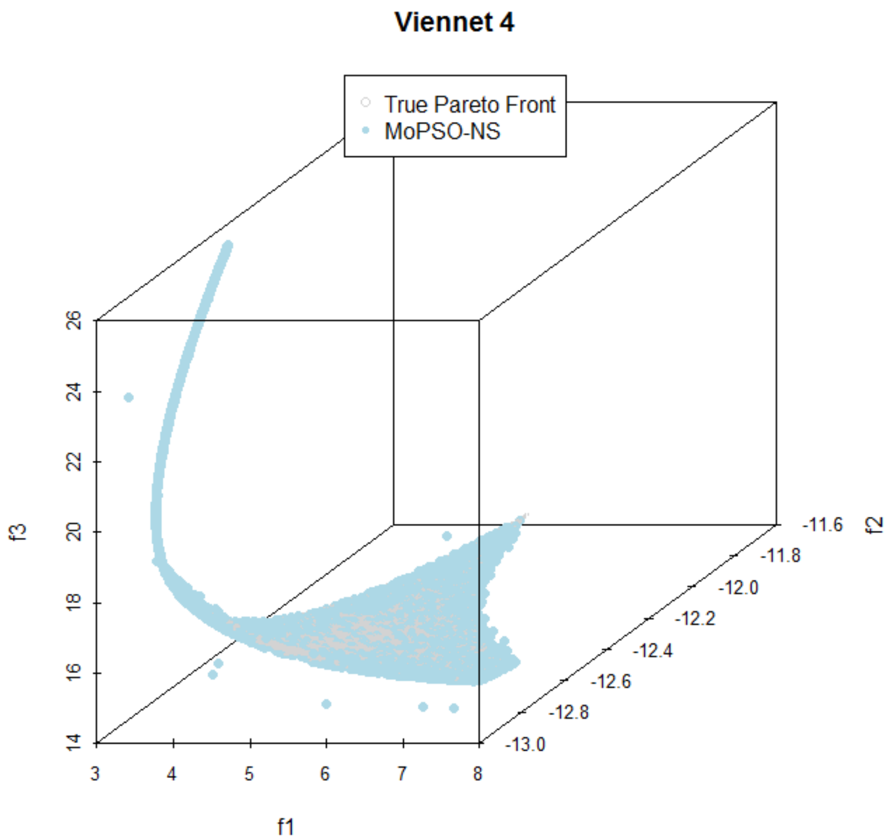


Figure 4.14 – Non-dominated solutions for Viennet 4's Test – MoPSO-NS

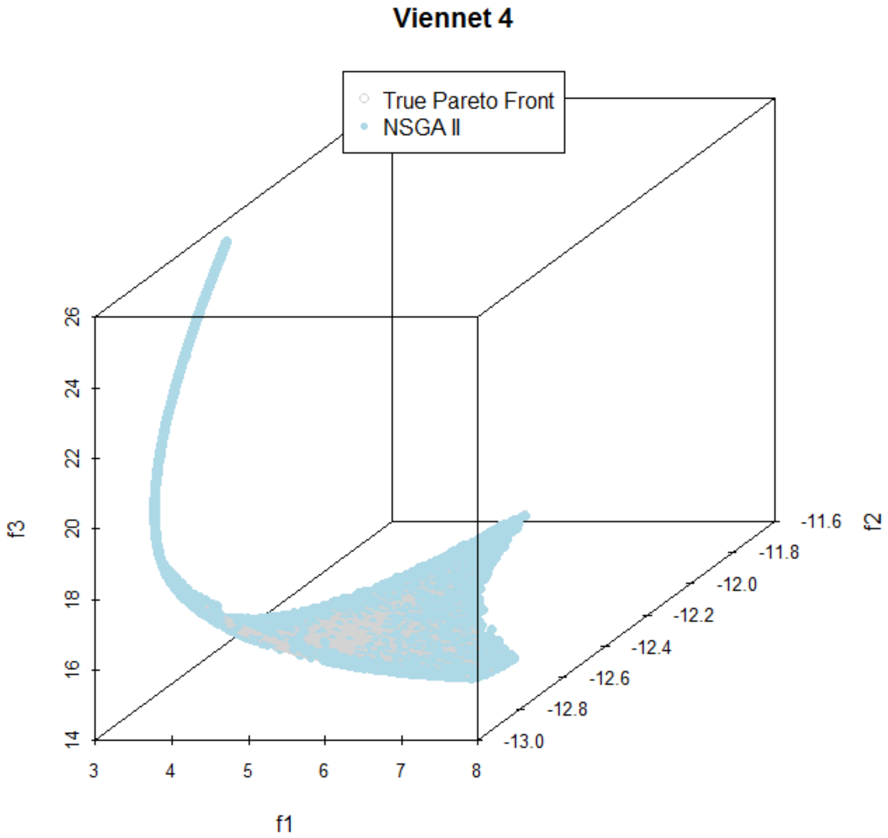


Figure 4.15 – Non-dominated solutions for Viennet 4's Test – NSGA-II

Table 4.6 – Metrics for Viennet 4's test problem

Solver	Metric	GD	iGD	ID	DD	SP	Time (ms)
	Mean	0.88746	0.07685	--	0.000	0.137	20798
	Median	0.88693	0.07685	--	0.000	0.137	20826
MoDE-NS	Max	0.88932	0.07699	--	0.000	0.147	21076
	Min	0.88580	0.07665	--	0.000	0.127	20514
	DP	0.00139	0.00010	--	0.000	0.005	168
	Mean	0.88849	0.07677	--	0.000	0.089	18060
	Median	0.88885	0.07677	--	0.000	0.089	18128
MoPSO-NS	Max	0.89100	0.07688	--	0.000	0.093	18236
	Min	0.88426	0.07668	--	0.000	0.083	17815

Solver	Metric	GD	iGD	ID	DD	SP	Time (ms)
	DP	0.00207	0.00007	--	0.000	0.003	135
NSGA-II	Mean	0.88739	0.07682	--	0.000	0.119	21402
	Median	0.88739	0.07683	--	0.000	0.119	21333
	Max	0.89126	0.07691	--	0.000	0.125	21746
	Min	0.88525	0.07671	--	0.000	0.112	21247
	DP	0.00196	0.00008	--	0.000	0.004	175

DP is the Standard deviation, ms is time in milliseconds.

In this test the Trade-off Graph, presented in Figure 4.16, was used. The graph was obtained from non-dominated solutions with the MoDE-NS, with α = 0.5 (weights) for all objectives. In the graph the highlighted solutions are the best compromise ones, according to weights adopted.

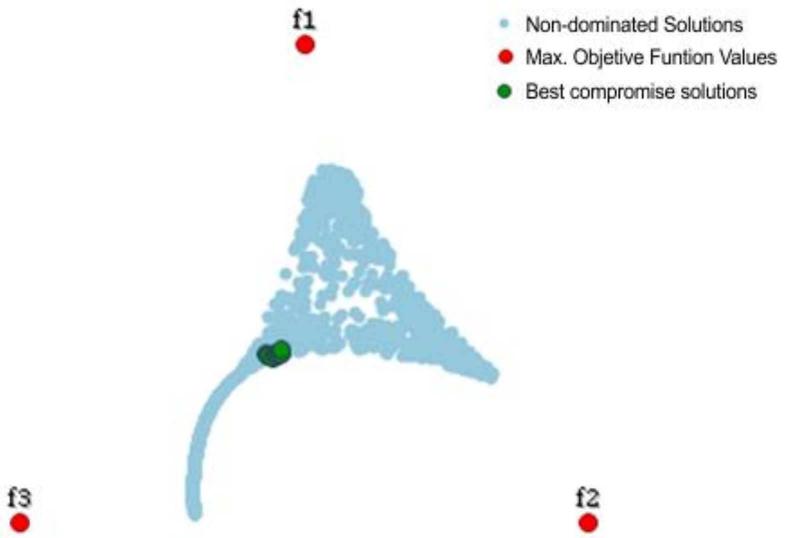


Figure 4.16 – TG graph for Viennet 4 test non-dominated solutions – MoDE-NS

4.1.6 *Test problem results discussion*

The performance of the algorithms developed can be considered satisfactory in the problems tested. However it is noticed that in general, MoDE-NS and MoPSO-NS had a overall better performance compared to NSGA-II on tests 1, 2, 3 and 4. The performance of the NSGA-II was slightly better in test 5. This demonstrates that the structure using two populations (one of parents and another of offspring) and using the parent population to store non-dominated solutions is very efficient when combined with the reproduction schemes of DE and PSO. Baltar (2007), who used an algorithm based on PSO with an external archive to store the non-dominated solutions, found that their algorithm presented, in general, worst results than NSGA-II. The processing time is low in all tests, except for Test 5, whose population is five times greater and the number of generations is four times larger. While the tests 1-4 processing time was less than 1 second, in the test 5 the time increased to about 20 seconds, caused by the increase of the population size. The tests were performed on a computer with an Intel I5.

The best performance of MoDE-NS and MoPSO-NS over NSGA-II is also observed in the application to case studies, especially in real world water resources system with large numbers of variables and constraints, as shown below.

4.2 Case 1: Application to Hydrological Model

The MOEAs are initially applied to a hydrological model for verification of their performance. This problem has fewer decision variables and the only lower and upper boundaries on decision variables. For multi-objective analysis with the model Smap, the library of algorithms was integrated to the interface was developed by Schardong (2009) and is presented in Figure 4.17.

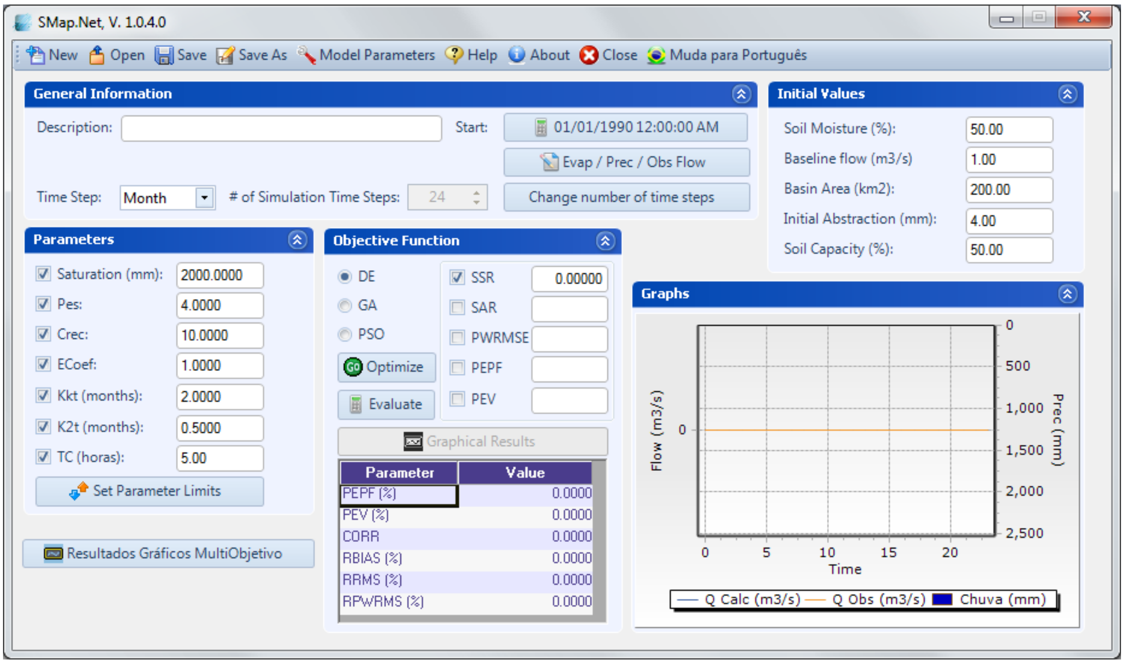


Figure 4.17 – User Interface of Smap.Net Model

SMAP model was originally developed by Lopes et al. (1982) aiming to create a simple model capable of being used and operated in regions with scarce data. The model structure is represented by three reservoirs representing the surface storage, the top layer of soil and underground storage, as in Figure 4.18. The version presented below corresponds to the diary time step version. User should refer to Lopes et al. (1982) and Schardong et al. (2009) for more details.

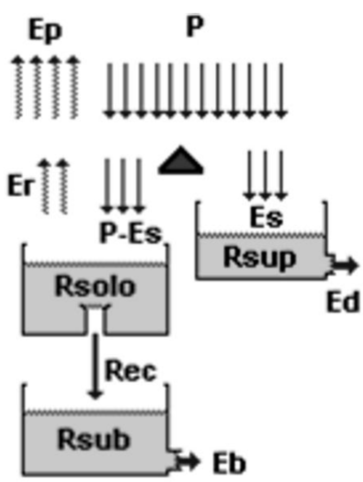


Figure 4.18 – Smap Model Schema - Source: Lopes et al. (1982)

The superficial storage (*Rsup*) can be written as:

$$\frac{dRsup}{dt} = Es - Ed \tag{4.6}$$

$$Es = \begin{cases} 0 & se \;\; P \leq AI \\ \dfrac{(P - AI)^2}{P - AI - S} & se \;\; P > AI \end{cases} \tag{4.7}$$

$$Ed = Rsup(1 - K2), \qquad onde \;\; K2 = 0.5^{1/K2t} \tag{4.8}$$

Where *Es* is the superficial runoff by the method of the Soil Conservation Service (SCS), *Ed* the flow of depletion of the direct runoff, *P* the precipitation, *AI*: initial abstraction for SCS method, S potential abstraction also for the SCS method. *Ed* is a function of the level of the surface reservoir (*Rsup*), *K2t* the coefficient of depletion of the surface reservoir, as defined in Lopes (1982). All units are in mm, except for *dt* and *K2t*, which have a dimension of time (days).

The reservoir of topsoil (*Rsolo*) can be mathematically written as:

$$\frac{dRsolo}{dt} = P - Es - Er - Rec \tag{4.9}$$

$$Es = \begin{cases} Ep & se \;\; IN > Ep \\ IN + (Ep - IN).Ti & se \;\; IN \leq Ep \end{cases} \tag{4.10}$$

$$Rec = \begin{cases} 0 & se \;\; Rsolo \leq RsoloM \\ Crec \times TU \times (Rsolo - RsoloM) & se \;\; Rsolo > RsoloM \end{cases} \tag{4.11}$$

$$TU = \frac{Rsolo}{Str} \tag{4.12}$$

$$RsoloM = Capc \times Rsolo \tag{4.13}$$

$$IN = P - Es \tag{4.14}$$

Where *P* and *Es* are as defined above, *Er* is the evapotranspiration rate, *Rec* is the recharge of the underground reservoir, *Crec* is the recharge coefficient of the underground reservoir, *IN* is defined as the difference between *P* and *Es* and is the portion of precipitation that infiltrates, *TU* is the soil moisture rate, *STR* maximum field capacity of soil and *Capc* maximum soil capacity in percentage (Lopes et al. 1982).

The underground reservoir (*Rsub*) can be mathematically written as:

$$\frac{dRub}{dt} = Rec - Eb \tag{4.15}$$

$$Eb = Rsub\,(1 - Kk) \quad com \quad Kk = 0.5^{(1/Kkt)} \tag{4.16}$$

Where, *Eb* is the base flow and *Kkt* is the depletion coefficient of the underground reservoir as defined in Lopes et al. (1982).

$$Q_M = (Eb + Ed) \times CoefConv \tag{4.17}$$

Where $CoefConv$ is the conversion coefficient from *mm* to m^3/s, and Q_M is the basin outflow in m^3/s.

To evaluate the automatic calibration, five objective functions (OF) were implemented in Smap.Net. These are described below as in Cunderlik and Simonovic, 2004, 2005 and USACE, 2001:

- Sum of Squared Residuals (SSR): This objective functions compares observed and modeled flow at each time interval and uses the squared differences as the measure of fit. Thus it gives greater weight to larger errors and less to small errors, equation (4.18).
- Peak-Weighted Root Mean Square Error (PWRMSE): This function is identical to

the SSR objective function. I also square the differences between flows for all time intervals and weight them. The weight assigned at each time interval is proportional to the magnitude of the modeled flow, thus, flows greater than the mean of the observed flow are assigned a weight greater than 1.00, and those smaller, a weight less than 1.00, equation (4.19).

- Sum of Absolute Errors (SAR): This objective functions compares observed and modeled flow at each time interval and thus, it gives equal weight to larger errors and less to small errors, equation (4.20).
- Percent Error in Peak Flow (PEPF): This objective function only compares the magnitude of computed peak flow and does not account for total volume or timing of the peak, equation (4.21).
- Percent Error in Volume (PEV): This objective function only considers the computed volume and does not account for the magnitude or timing of the peak flow, equation (4.22).

The objective functions SSR, PWRMSE and SAR are implicitly measure of the magnitudes of the peaks, volumes, and times of peak of the modeled and observed hydrographs. The equations for each of the functions described are presented in Table 4.7.

Table 4.7 – Objective Functions Equations used with SMAP

Objective Function	**Equation**	
SSR	$\sum_{t=1}^{N}\left(\frac{Q_o(t)-Q_M(t)}{Q_o(t)}\right)^2$	(4.18)
PWRMSE	$\sqrt{\frac{1}{N}\sum_{t=1}^{N}\left(Q_o(t)-Q_M(t)\right)^2 \times \frac{Q_o(t)-Q_{OA}}{2Q_{oA}}}$	(4.19)

Objective Function	Equation	
SAR	$\sum_{t=1}^{N} \lvert Q_o(t) - Q_M(t) \rvert$	(4.20)
PEPF (%)	$100 \times \left\lvert \frac{Q_o(p) - Q_M(p)}{Q_o(p)} \right\rvert$	(4.21)
PEV (%)	$100 \times \left\lvert \frac{V_o - V_M}{V_o} \right\rvert$	(4.22)

Where N is the number of time steps, $Q_O(t)$ and $Q_M(t)$ are observed and modeled flow as t time interval time, respectively, $Q_O(p)$ and $Q_M(p)$ are observed and modeled flow at the peak, respectively, and V_O and V_M, are observed and modeled volume, respectively and Q_{OA} is the mean observed flow.

For case study a basin in south Brazil, in Parana state was used, as shown in Figure 4.19. This basin has about 2901 km2, and is an important tributary of the Parana River that forms the Itaipu Reservoir, of Itaipu hydroelectric power plant. The climate in the Piquiri basin region is tropical/sub-tropical with an annual average temperature of 15 C and more than 1500 mm precipitation per year. The observed flow series and precipitation for calibration and verification was obtained from Brazilian Hydrologic Information System (HidroWeb – ANA, 2009).

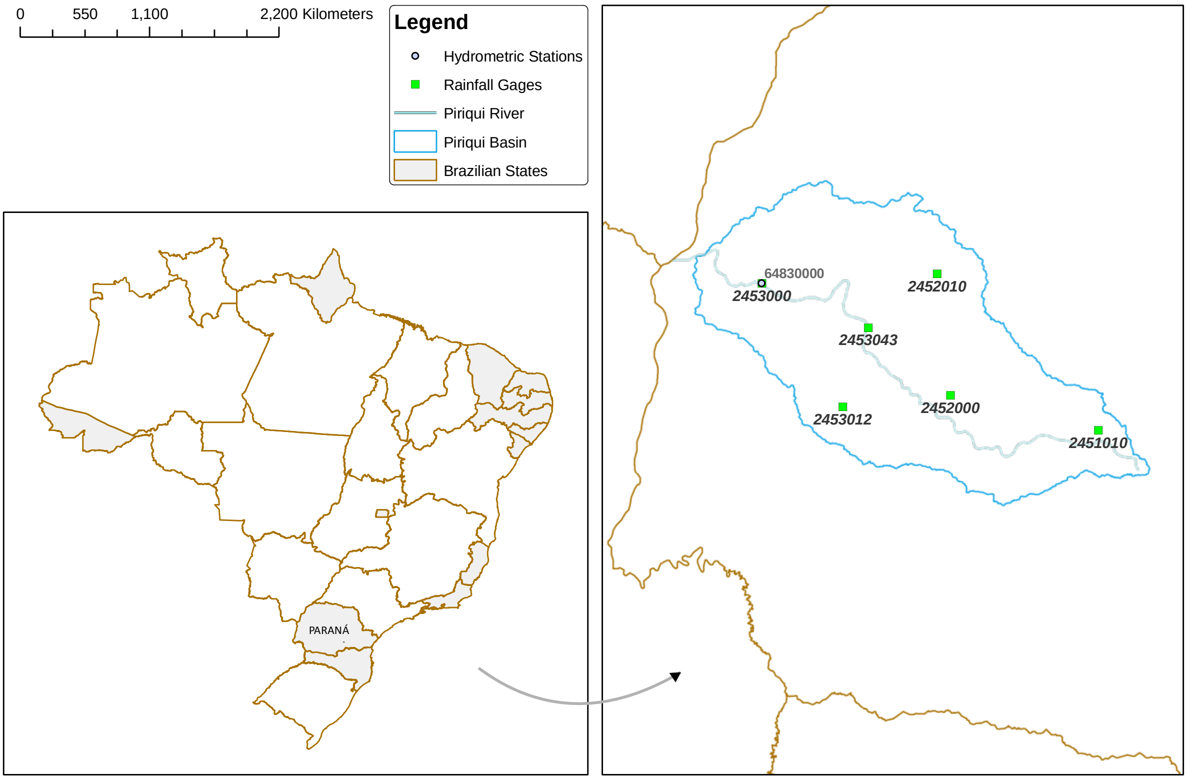


Figure 4.19 –Piriqui River basin location in Parana State - Brazil

Initially the calibration was done considering the objective functions and SSR PWRMSE (using single-objective optimization). For the calibration was used the time period from 08/01/1998 to 30/11/1998. More details can be seen in Schardong et al. (2009).

These values obtained for this calibration are shown in Table 4.8, and the parameters considered were: maximum soil saturation capacity (*STR* in mm), coefficient of groundwater recharge (*CRec*), evapotranspiration coefficient (*ECoef*), surface reservoir depletion coefficient (*Kkt*) and underground reservoir depletion coefficient (*K2t*).

Table 4.8 – Parameter values obtained with Objective Function SSR e PWRMSE – Using single objective optimization

Parameter	Objective SSR	Objective PWRMSE
Valor da Objective Function	3.423	284.061
STR (mm)	124.329	157.072

Parameter	Objective SSR	Objective PWRMSE
Crec	20.000	15.151
ECoef	0.800	0.800
Kkt	46.544	37.623
K2t	2.393	1.924

The graph in Figure 4.20 shows the comparison of observed and modeled flow hydrographs (using single-objective version of Differential Evolution) for both objective functions analyzed. This graph shows a good fit obtained in the calibration process.

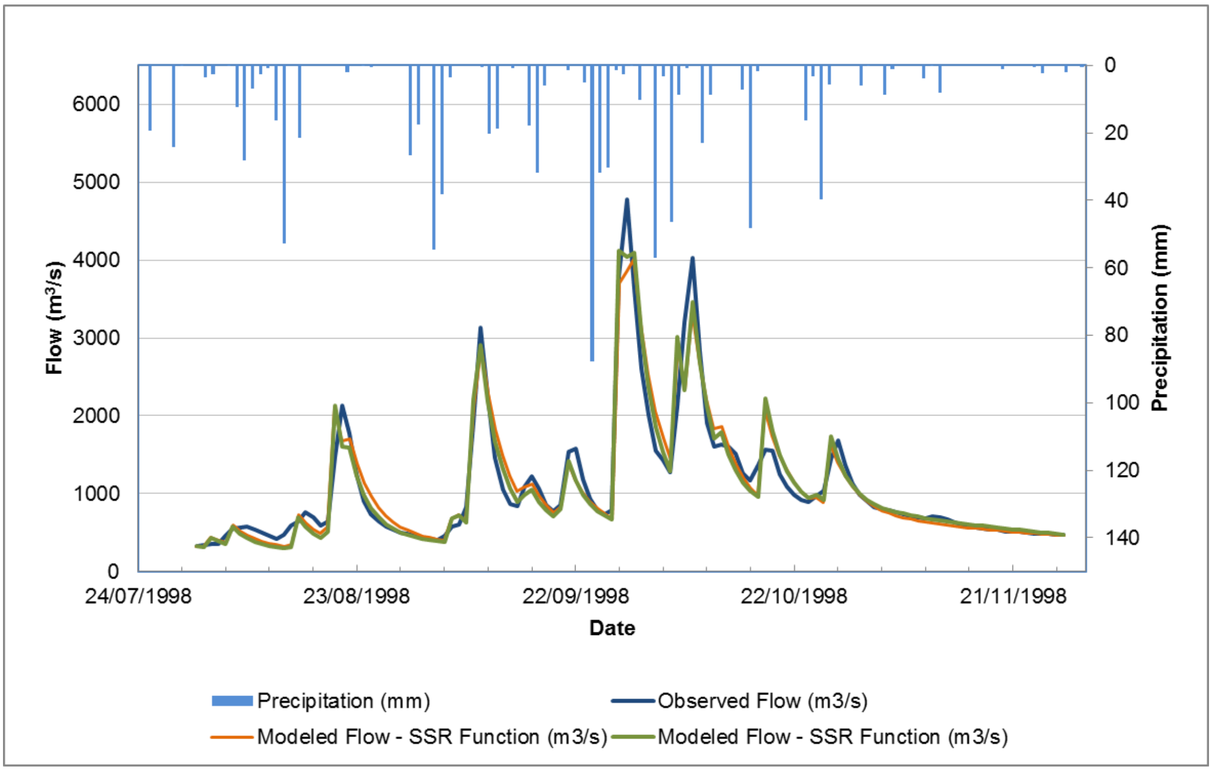


Figure 4.20 – Modeled and calculated flow for Objective Functions SSR and PWRMSE

4.2.1 *Analysis using two objectives*

For the multi-objective analysis were considered two of the objective functions presented in the previous item: SSR and PWRMSE. The three algorithms were applied to the same case study described with 500 generations and a population of 100 individuals. The non-dominated solutions are presented in Figure 4.21. It is possible note that all three algorithms were able to find very similar non-dominated solution.

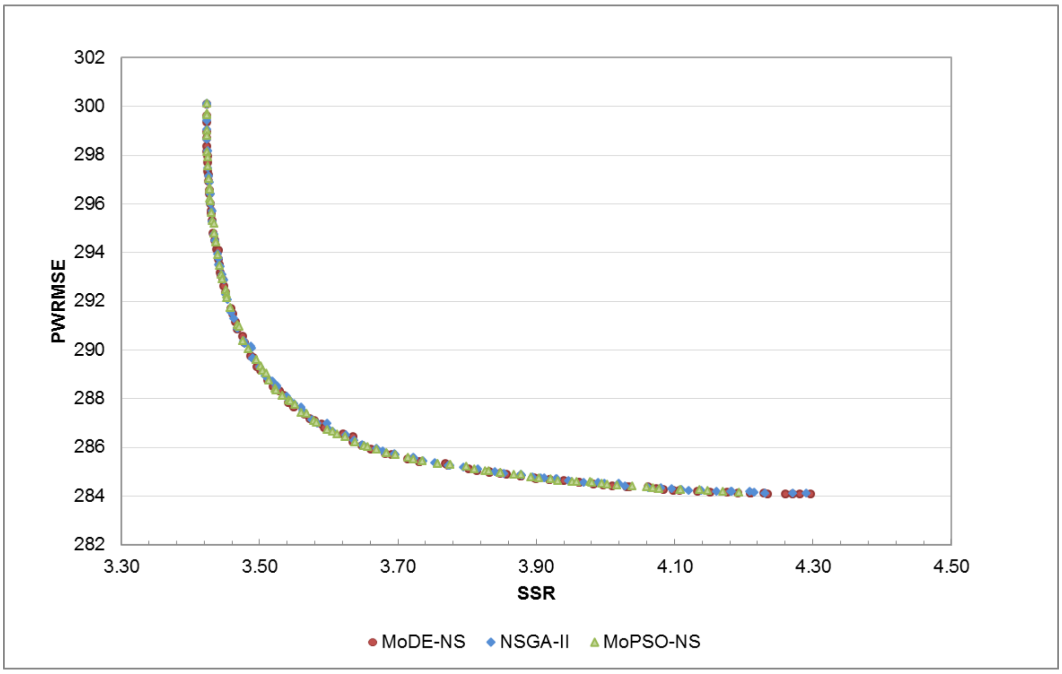


Figure 4.21 – Non-dominated solutions for two Objectives: SSR e PWRMSE

Applying the TG to the results with α = 0.5 (weights) for both objective functions it is possible obtain the “best” compromise solutions for each algorithm, as well as the optimal values of the parameters, as shown in Table 4.9. The results are nearly identical, and the graph presented in Figure 4.22 presents the modeled flows using the parameters in Table 4.9 from MoDE-NS algorithm. The hydrograph of the other two algorithms are not presented here, as is not possible distinguish them visually due to the overlap. Table 4.10 shows some performance metrics of calibration (USACE, 2001). CORR metric measures the correlation between observed and modeled flow, and its ideal value is 1 (one), when there is perfect correlation. This table is presented mainly for comparison with the next item, where five objectives will be considered. The metrics include PEPF and PEV that are also used as objective functions in the next section.

Table 4.9 – Parameter values obtained for one compromise solution with $\alpha = 0.5$

		MoDE-NS	**MoPSO-NS**	**NSGA-II**
Function Objective	SSR	3.57	3.58	3.56
	PWRMSE	287.15	287.08	287.40
Parameters	Soil saturation (mm)	134.59	135.04	134.42
	Crec	20.00	19.99	20.00
	Ecoef	0.80	0.80	0.80
	Kkt	42.45	42.82	42.54
	K2t	2.06	2.06	2.08

Table 4.10 – Calibration quality measure from previous solution analysed

Measure	**MoDE-NS**	**MoPSO-NS**	**NSGA-II**
PEPF (%)	13.954	14.119	14.200
PEV (%)	0.678	0.539	0.673
CORR	0.960	0.960	0.960
RBIAS (%)	0.125	0.032	0.182
RRMS (%)	15.808	15.766	15.791
RPWRMS (%)	16.279	16.236	16.259

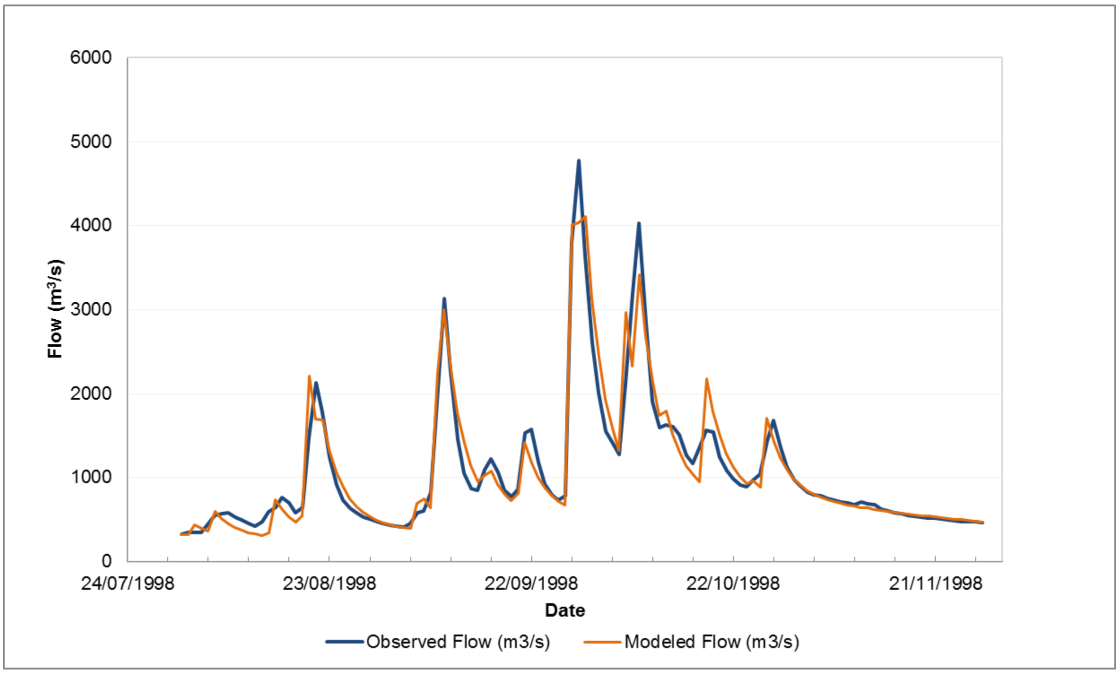


Figure 4.22 – Modeled and calculated flow for compromise solution selected, with α = 0.5 for all objectives (MoDE-NS)

4.2.2 Calibration analysis with five Objectives

In this section, all five objective functions are considered for the multi-objective analysis. The optimization is performed using the same number of individuals (100) and 500 generations. The result of parameters and the values of objective functions, considering α = 0.5 (weights) for all objectives, are presented in Table 4.11. The best compromise solution presents model parameters values very similar for all three algorithms. Table 4.12 presents the same metrics calculated in the previous item. The comparative analysis indicates that the CORR metric did not change. The PEPF metric, which measures the difference between modeled and observed peak flow, had a small reduction in all three algorithms. This indicates that the peaks of the modeled flows in this item will be slightly larger than those calculated in the previous item, which can be seen in Figure 4.23. In this case the modeled flow that results from the calibration with the three algorithms is almost coincident and therefore only one of them (from MoDE-NS algorithm) is presented in Figure 4.23.

Table 4.11 – Parameter values obtained for one compromise solution with α = 0.5, from analysis with 5 objectives

		MoDE-NS	MoPSO-NS	NSGA-II
Objective Function	SSR	3.79	3.89	3.84
	PWRMSE	287.70	289.55	289.36
	SAR	17111.34	17144.06	17115.38
	PEPF	11.55	10.16	10.87
	PEV	0.87	0.45	1.32
Parameters	Soil saturation	140.21	141.68	140.37
	Crec	20.00	20.00	18.85
	Ecoef	0.80	0.80	0.80
	Kkt	39.58	40.56	37.54
	K2t	1.88	1.82	1.87

Table 4.12 – Calibration quality measure from previous solution analysed

Measure	MoDE-NS	MoPSO-NS	NSGA-II
PEPF (%)	11.55	10.16	10.86
PEV (%)	0.87	0.45	1.34
CORR	0.96	0.96	0.96
RBIAS (%)	0.03	-0.44	0.12
RRMSE (%)	16.05	16.16	16.23
RPWRMSE (%)	16.52	16.63	16.71

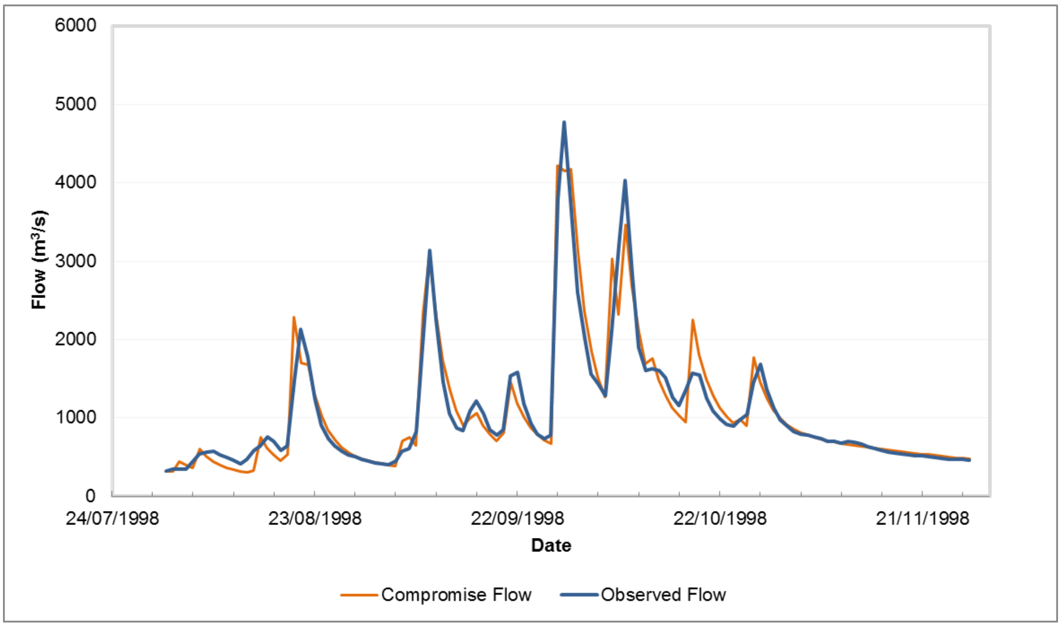


Figure 4.23 – Modeled and calculated flow for compromise solution selected with com α = 0.5 for all objectives (MoDE-NS)

In Figure 4.24, Figure 4.25 and Figure 4.26 present the TG for MoDE-NS, MoPSO-NS and NSGA-II, respectively. In the three graphs, α = 0.5 was used for all objectives. The distribution of solutions for the three algorithms is similar; however the number of non-dominated solutions is limited to the number of individuals, which in this case is 100. With a larger population would be expected that the solutions space would be more populated. It is important to note however, that the best compromise solution using two and five objectives provide very similar flows.

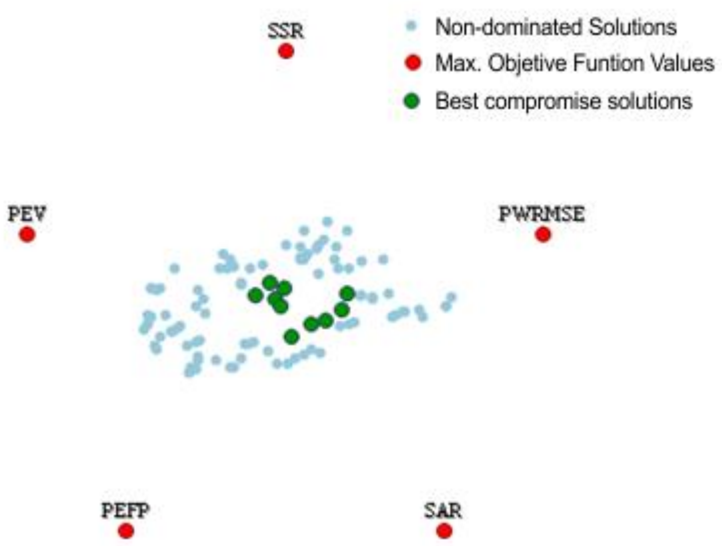


Figure 4.24 – Trade-off Graph for MoDE-NS with five Objective, and α = 0.5 (for all objectives)

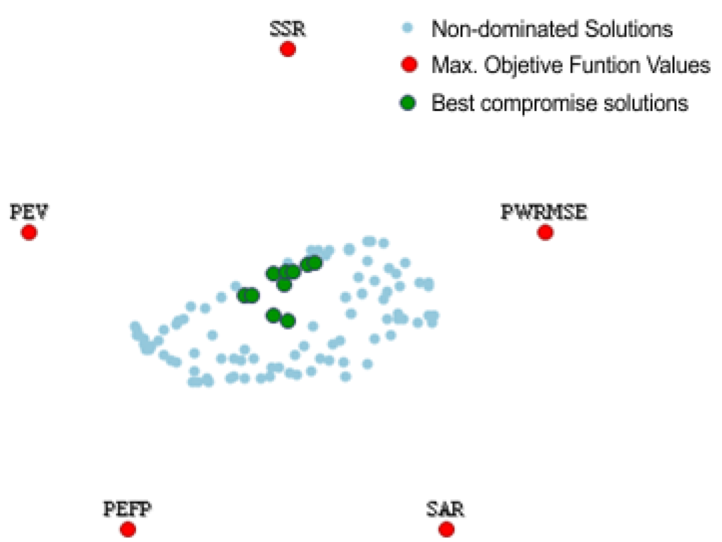


Figure 4.25 – Trade-off Graph for MoPSO-NS with five Objective, and α = 0.5 (for all objectives)

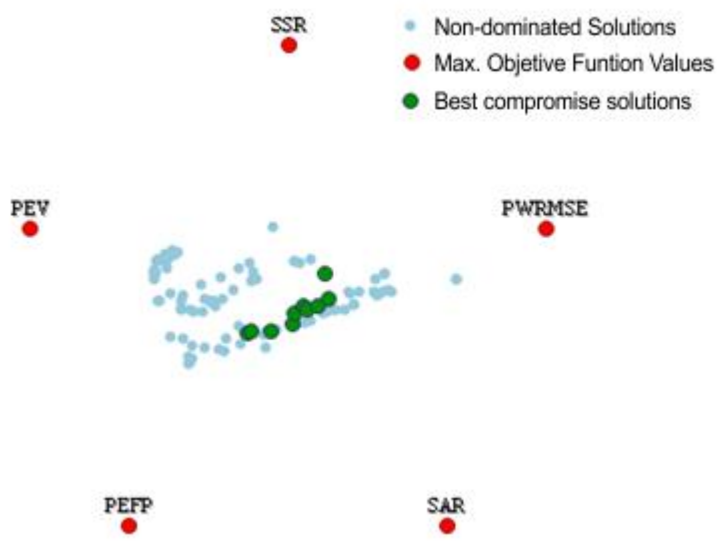


Figure 4.26 – Trade-off Graph for NSGA-II with five Objective, and α = 0.5 (for all objectives)

Calibration of hydrological model parameters Smap is performed without difficulty using the MOEAs. Calibration was performed considering two and five objectives, both with satisfactory results. The processing time for each simulation was about 20 seconds for the analysis with two and around 30 seconds to five objectives in a computer with an Intel I5 processor.

4.3 Case 2: Multi-objective analysis of a water resource system

To test the performance of the algorithms, they were applied to a complex water supply system with a series of reservoirs connected by tunnels and canals. Cantareira system is one of the main water supplies for the São Paulo metropolitan region – SPMR, providing about 33 m^3/s.

4.3.1 Metropolitan Region of São Paulo Water Supply System

The Metropolitan Region of São Paulo - SPMR is supplied by 6 (six) major systems: Cantareira, Guarapiranga, Alto Tiete, Rio Grande, Rio Claro and Cotia. The most import system is Cantareira, which supplies about 33 m^3/s for the SPMR.

The Cantareira system derives water of Piracicaba, Capivari and Jundiaí (PCJ) basin for the Upper Tiete River basin by a series of reservoirs, channels and tunnels. The Cantareira system consists of five main reservoirs, four of them: Jaguari Jacareí, Cachoeira and Atibainha are located in the PCJ basin (Figure 4.27) and the Paiva Castro reservoir belongs to the basin of the Upper Tiete River. Operational data, demand values (both for the Piracicaba River and for the SPMR), minimum and maximum flows (flow restriction) downstream of the reservoirs, has been compiled from, ANA (2004), DAEE (2009) and Sabesp (2009a and 2009b) and are presented in Table 4.13 and Table 4.14. The series of natural flows used was obtained from Sabesp (2009b).

Table 4.13 – Operational data of Cantareira System

Reservoir	**Operation levels (m)**		**Volumes (x10e6 m3)**		
	Min	Max	Min	Max	Live
Jaguari-Jacarei	820.80	844.00	239.45	1047.49	808.04
Cachoeira	811.72	821.78	46.92	115.71	68.79
Atibainha	781.88	786.86	199.20	295.52	96.32
Paiva Castro	743.80	745.61	25.52	33.05	7.53

Table 4.14 – Cantareira system minimum and maximum downstream restrictions flows

Reservoir	**Min Downstream Flow (m^3/s)**	*** Max Downstream Flow (m^3/s)**	**Basin Area (km^2)**	**Historical Avg. Flow (m^3/s)**

Reservoir	Min Downstream Flow (m^3/s)	* Max Downstream Flow (m^3/s)	Basin Area (km^2)	Historical Avg. Flow (m^3/s)
Jaguari-Jacarei	1.00	50.00	1230.00	25.20
Cachoeira	1.00	7.00	392.00	8.50
Atibainha	1.00	3.00	312.00	6.00
Paiva Castro	1.00	50.00	639.00	4.60
			Total	44.30

* may vary depending on downstream conditions, DAEE (2009).

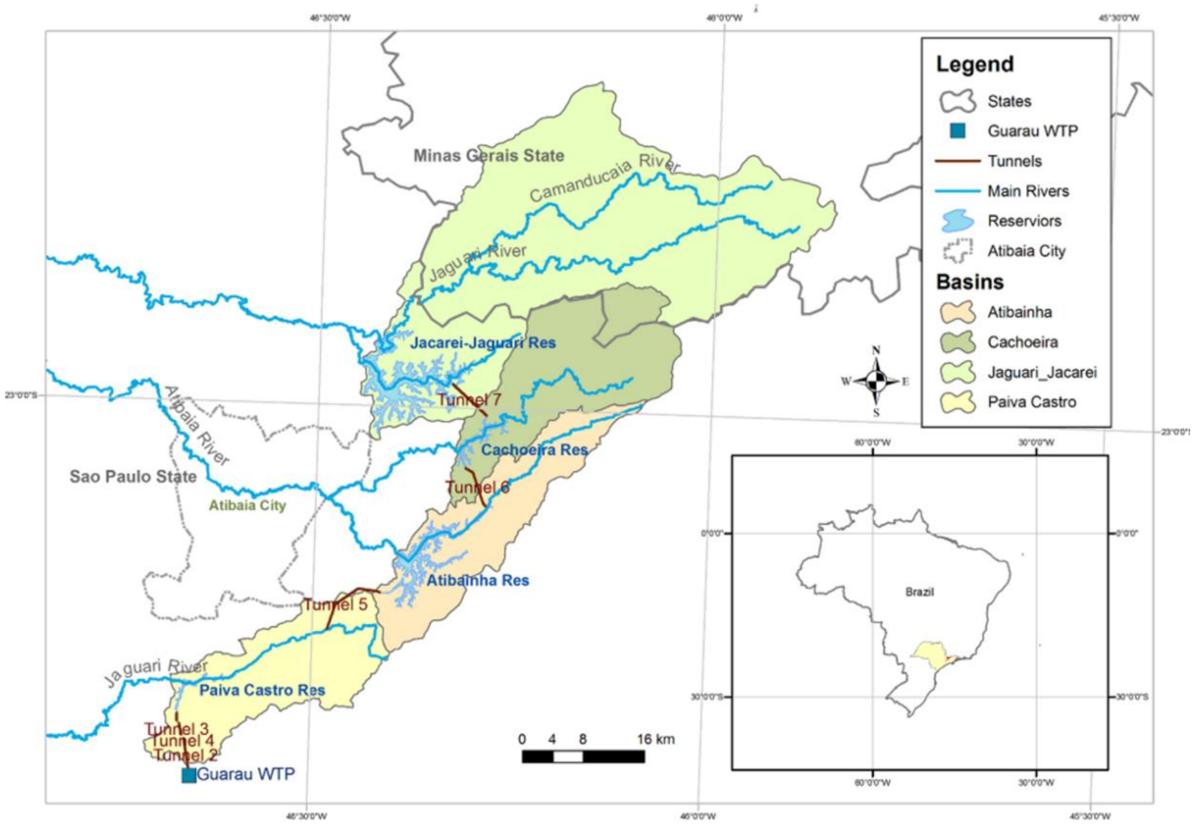


Figure 4.27 – Cantareira System location and main components. Source: Sabesp (2009a)

The first reservoirs of the system, as in Figure 4.27, are Jaguari and Jacarei that are connected by a channel and can be treated as a single reservoir in the model. These are connected to the Cachoeira reservoir by Tunnel 7. Tunnel 6 connects the Cachoeira reservoir to Atibainha reservoir which is connected by Tunnel 5 to the Paiva Castro reservoir, where the transfer occurs between the PCJ and Upper Tiete basins. The three tunnels transfer water by gravity. Paiva Castro reservoir water is conducted by gravity to the pumping station Santa Ines (EESI). After being elevated by a set of pumps the water passes through a small reservoir for regulation in case of emergency (e.g. power outage) and finally led to the water treatment plant (WTA) Guarau. For its small storage capacity (<1 hm^3), the Aguas Claras Reservoir will not be considered here, to be held in the monthly period.

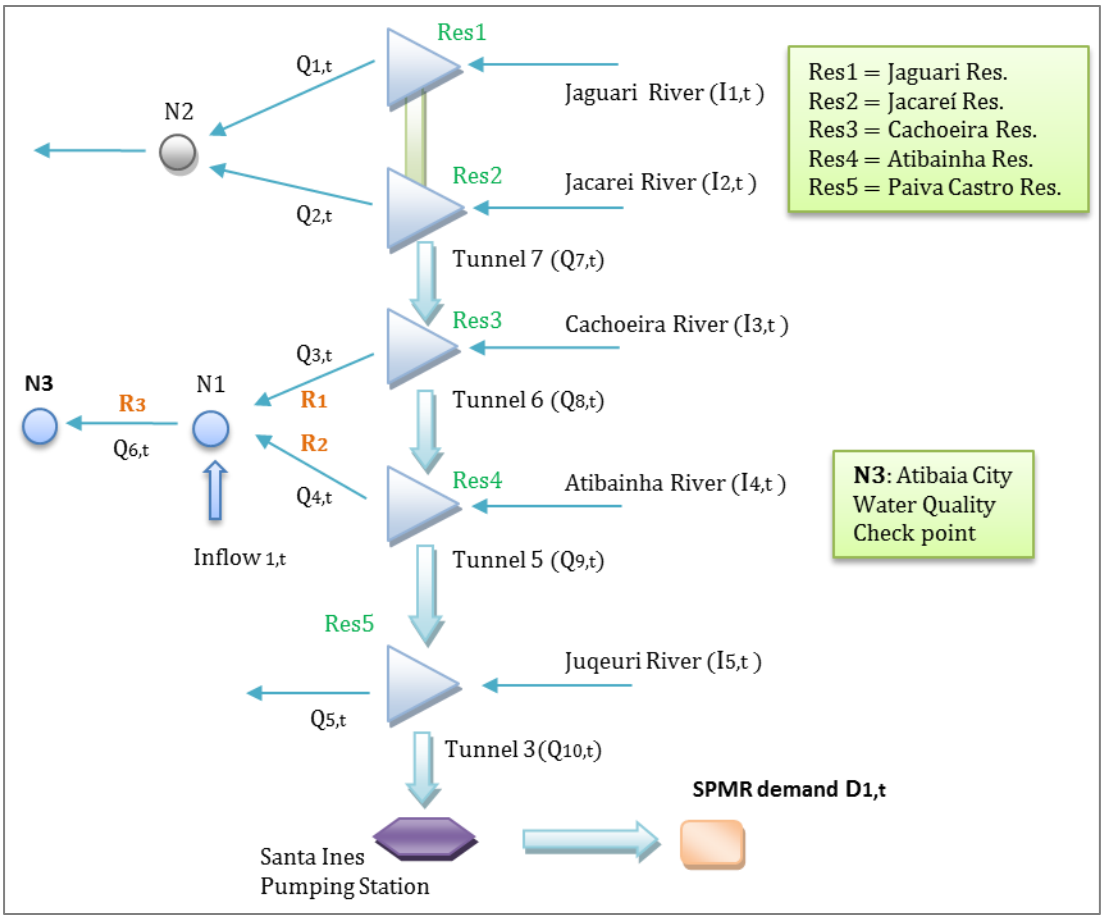


Figure 4.28 – Schematic for Cantareira System Water Supply

The analysis was done three objectives: f_1 – Minimization of demands deficits for SPMR and PCJ basin; f_2: Minimization of water quality standards deviation of Atibaia river at the control point N3 (as in Figure 4.28) considering the BOD (Biochemical Oxygen Demand) and f_3 - Minimization of pumping cost for Santa Ines station. Figure 4.28 presents a schematic of the Cantareira system with indication of the variables used in the simulations. The objective functions are described in detail below, as well as balance constraints and limits of the system variables.

- Objective Function f_1: Minimizing Deficit meet demand: The objective function that calculates the demand deficits is presented in equation (4.23).

$$\min f_1 = \sum_{t=1}^{m} \left(D_{1,t} - Q_{10,t}\right)^2 + \left(D_{2,t} - [Q_{1,t} + Q_{2,t} + Q_{3,t} + Q_{4,t}]\right)^2 \qquad (4.23)$$

Where $D_{1,t}$ is the demand for SPMR, $Q_{10,t}$ is the flow at station Santa Ines pumping station, $D_{2,t}$ the demand for PCJ basin, $Q_{1,t}$, $Q_{2,t}$, $Q_{3,t}$ and $Q_{4,t}$ are the downstream flow for Jaguari/Jacarei, Cachoeira and Atibainha reservoir, respectively and is adopted as 5 m^3/s.

- Objective Function f_2: Minimization of water quality deviation from standards at Atibaia city at the control point N3: the objective function for the deviation of the BOD value in relation to the class of the Atibaia river standards is calculated as shown in equation (4.24).

$$\min f_2 = \sum_{t=1}^{m} \left(BDOlim - f_{DBO}(Q_{3,t}, Q_{4,t}\ e\ Q_{6,t})\right)^2 \qquad (4.24)$$

Where f_{DBO} is the value of BOD concentration in mg/L at node N3 (at Atibaia city) and BOD is the maximum limit of the class.

This stretch of the Atibaia River is classified as Class 2 (according Brazilian regulation). For this class, the maximum allowed of BOD is 5.0 mg/L. For evaluation of the BOD, equation (4.25) is used (Chapra, 1997).

$$L = L_0 e^{-\frac{K_r}{U}x}$$

Where:

- $K_r = K_d + K_s$
- L = BOD concentration (mg/L)
- L_0 = BOD upstream concentration (mg/L)
- K_d = oxygenation constant (d^{-1})
- K_s = sedimentation constant (d^{-1})
- K_r = BOD removal rate (d^{-1})
- x = reach length (m)
- U = reach velocity (m/s)

(4.25)

Where the velocity (U), and the water level (H) are calculated as in equation (4.27).

$$U = aQ^b$$

$$H = cQ^d$$

Where:

(4.26)

- H = water level depth (m)
- U = reach velocity (m/s)
- x = reach length (m)
- Q = reach flow (m^3/s)
- *a, b, c, d*: coefficients

The BOD decay is calculated in reaches R_1, R_2 and R_3 (Figure 4.27) for each time interval and the mass balance is calculated at N1 and N3. The coefficients used in the equations for water quality model are presented in Table 4.15, and were obtained from COBRAPE (2010). Other data such as flow and concentration of loads were also obtained from COBRAPE (2010), and are presented in Table 4.16.

Table 4.15. Water quality model coefficients

Reach	Length (km)	a	b	C	d	Kd	Ks
R1	35.27	0.35	0.28	0.41	0.44	1.05	0.25

Reach	Length (km)	a	b	C	d	Kd	Ks
R2	26.30	0.34	0.27	0.41	0.45	1.05	0.25
R3	24.40	0.83	0.27	0.28	0.49	1.05	0.25

Table 4.16. Withdraw and loads for each reach

Reach	**Withdraws (m^3/s)**	**Loads**	
		Flow (m^3/s)	BOD (mg/L)
R1	0.13	0.07	129.20
R2	0.08	0.05	155.82
R3	0.90	0.30	149.77

- Objective Function f_3 – Minimization of the cost of pumping: The objective function of pumping cost of the EESI is presented in equation (4.23), whose estimative was made based on the head and on the average price of electricity, since it was not possible to obtain the data directly from Sabesp.

$$\min f_3 = \sum_{t=1}^{m} f_{bomb}(Q_{10,t}) \tag{4.27}$$

Where $Q_{10,t}$ is the flow transferred through the pumping station Santa Ines at each time period *t*. the value of f_{bomb} is calculating, assuming that:

- Pumping hours $pHrs = 720/33\ Q_{10,t}$, where 730 is the number of average hours per month, and 33 is the maximum monthly flow in m^3/s
- Estimated energy cost (pC): R$[1] 0.17/kWh
- Total head: $H \approx 120\ m$
- Power: $P(hp) = (\gamma \times H \times Q_{10,t})/76\eta$

[1] 1 U$ = 1,80 R$ (Exchange rate by Nov-2011)

- Yield and efficiency of the pump $\eta \approx 0.81$
- Water Specific weight $\gamma = 1000\ kg/m^3$
- Energy consumed: $E(kWh) = 0.7457 \times P$
- The pumping cost can be written as function of $Q_{10,t}$, as in equation (4.28).

$$f_{bomb}(Q_{10,t}) = pC \times pHrs \times E$$
$$f_{bomb}(Q_{10,t}) = pC \times pHrs \times 0.7457 \frac{\gamma \times H \times Q_{10,t}}{76\eta} \tag{4.28}$$

The problem constraints are:

- Reservoir mass balance:
  - $S_{k,t} = S_{k,t-1} + I_{k,t} - E_{k,t} - Q_{k,t} + \sum Q_{j,t} \quad \forall i;\ \forall t;\ \forall k$
- Minimum and maximum storage constrain:
  - $SMin_{k,t} \leq S_{k,t} \leq SMax_{k,t} \quad \forall i;\ \forall t$
- Maximum downstream flows:
  - $Q_{k,t} \geq L_{k,t} \quad \forall i;\ \forall t, k = 1, \dots, 5$
- Maximum tunnels transfers:
  - $0 \leq Q_{j,t} \leq UT_{j,t} \quad \forall i;\ \forall t, j = 7, \dots, 10$
- Non-negative flows:
  - $Q_{k,t}, Q_{j,t} \geq 0 \quad \forall i;\ \forall t$

The problem variables are:

- Reservoir downstream flows:
  - $Q_{1,t},\ Q_{2,t},\ Q_{3,t},\ Q_{4,t}$
- Flows transferred through tunnels
  - $Q_{7,t}, Q_{8,t}, Q_{9,t}$
- Flow provided for SPMR:
  - $Q_{10,t}$

Where k is the reservoir index, t the time period, $S_{k,t}$ the volume of reservoir k at time period t, $I_{k,t}$ the inflow, $Q_{k,t}$ the downstream flow, $Q_{j,t}$ are the flows transferred throw the tunnels, and $E_{k,t}$ the evaporation. $L_{k,t}$ is the downstream minimum flow, $UT_{j,t}$ is the maximum flow at tunnel j at time period t, $SMin_{k,t}$ *and* $SMax_{i,t}$ are the minimum and maximum volumes of reservoirs *k* at time period *t*.

Using schematic shown in Figure 4.28, the mass balance equation can be written as:

Jaguari-Jacarei (1 e 2) reservoir:

- $S_{1,t} = S_{1,t-1} + I_{1,t} + I_{2,t} - E_{1,t} - Q_{1,t} - Q_{2,t} - Q_{7,t}$
- $Q_{1,t} \geq 0.5\ \frac{m^3}{s}$
- $Q_{2,t} \geq 0.5\ \frac{m^3}{s}$
- $0 \leq Q_{7,t} \leq 31\ \frac{m^3}{s}$
- $239.45 \leq S_{1,t} \leq 1047.49$ hm^3

Cachoeira (3) reservoir:

- $S_{3,t} = S_{3,t-1} + I_{3,t} - E_{3,t} - Q_{3,t} + Q_{7,t} - Q_{8,t}$
- $Q_{3,t} \geq 0.5\ \frac{m^3}{s}$
- $0 \leq Q_{8,t} \leq 31\ \frac{m^3}{s}$
- $46.92 \leq S_{3,t} \leq 115.71$ hm^3

Atibainha (4) reservoir:

- $S_{4,t} = S_{4,t-1} + I_{4,t} - E_{4,t} - Q_{4,t} + Q_{8,t} - Q_{9,t}$
- $Q_{4,t} \geq 2.0\ \frac{m^3}{s}$
- $0 \leq Q_{9,t} \leq 35\ \frac{m^3}{s}$
- $199.20 \leq S_{4,t} \leq 295.52$ hm^3

Paiva Castro (5) reservoir:

- $S_{5,t} = S_{5,t-1} + I_{5,t} - E_{5,t} - Q_{5,t} + Q_{9,t} - QT_{10,t}$
- $Q_{5,t} \geq 0.5\ \frac{m^3}{s}$
- $0 \leq Q_{10,t} \leq 33\ \frac{m^3}{s}$
- $25.52 \leq S_{5,t} \leq 33.05$ hm^3

For node N1

- $Q_{6,t} = Q_{3,t} + Q_{4,t} + NI_{1,t},\ wher\ \ NI_{1,t}\ \ is\ the\ downstream\ flow$

To simplify the analysis, evaporation was not considered, but could easily be incorporated with the use of polynomial equations or exponential adjustments that would allow calculating the area and the volume evaporated. In evolutionary algorithms there is no restriction on the type of functions to be used. This problem has 198 variables and 96 constraints for the monthly mass balance in the reservoirs.

4.3.2 Multi-objective analysis

The multi-objective analysis was performed considering a period of 24 months, corresponding to two years of data for natural inflows in the period 1952 to 1954, which average flow is 33.95 m^3/s that is below the historical average of approximately 44.3 m^3/s. For this analysis, we two scenarios were considered for comparing the objective functions defined above:

- Scenario1 1: f_1 vs f_3
- Scenario 2: f_1 vs f_2

For the analysis the demand for SPMR as set to 33 m^3/ s. The population size was set to 500 individuals with a maximum of 50,000 generations, and for each scenario were performed a total of 10 full simulations.

The objective function f_1 and f_3 are calculated with the flow supplied to the SPMR, $Q_{10,t}$, and thus it is possible to calculate the Pareto optimal front, with which it is possible to compare and evaluate the results of the algorithms applied. To calculate the optimal Pareto front, the hypothesis adopted is that there are no demand deficits for the SPMR. Even if this hypothesis is not true, there will be no negative effect on checking the algorithms performance, since the analysis is done by comparing the relative performance metrics. Therefore, the value of $Q_{10,t}$ is changed from 0 to 33 m^3/s in 0.1 m^3/s steps, and the values of f_1 e f_3 calculated, resulting in a set of 330 points that are part of Pareto optimal front. This set of solutions is shown in Figure 4.29.

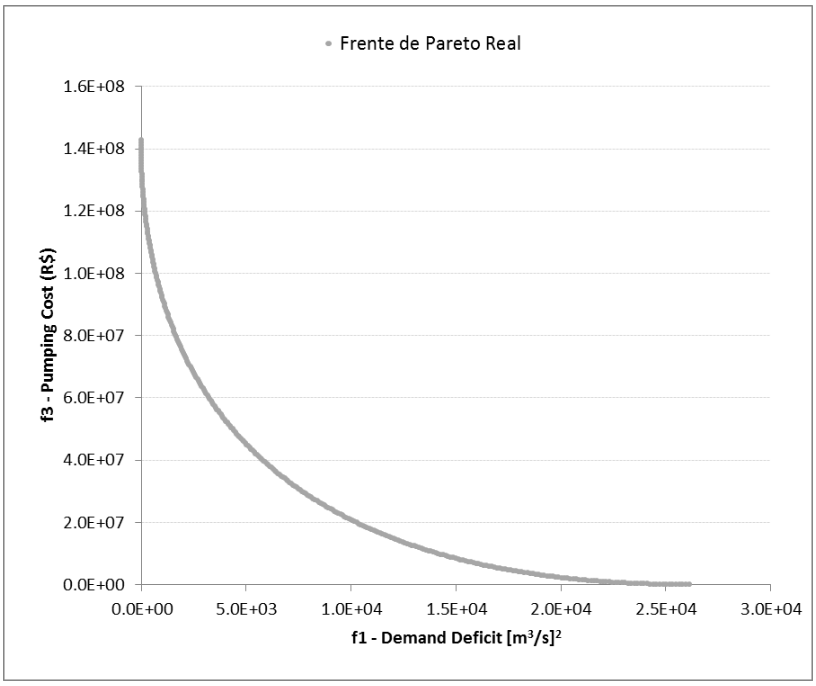


Figure 4.29 – True Pareto front for scenario f_1 *vs* f_3 – for 24 months simulation period

Scenario 1: f_1 vs f_3

Table 4.17 shows the performance metrics for the simulation considering f_1 vs f_3, and the graph of non-dominated set of solutions is shown in Figure 4.30, Figure 4.31 and Figure 4.32, for MoDE-NS, MoPSO-NS and NSGA-NS-II, respectively. It is possible to note, that the NSGA-II had more trouble finding the extreme value corresponding to lower values of f_1. Using the TG with α (f_1) = 1 and α (f_3) = 0 it is possible to determine these extreme solutions (with the minimum deficit for demands) for each of the algorithms, as shown Table 4.18. Figure 4.33 shows the comparison of the best non-dominated set of solutions of each algorithm from 10 simulations. MoPSO-NS obtained the best results for achieving better coverage of the Pareto optimal front, especially for lower values of demand deficits, where the NSGA-II has extreme difficulties to determine solutions. The results for MoDE-NS are very close to MoPSO-NS's, and the difference is for lower value of f1 when the demand deficit is close to zero.

A simulation was done in AcquaNet (which uses a network flow algorithm), with the same data and network configuration. The maximum deficit calculated with this result, using equation f_1 was approximately 6.25 $[m^3/s]^2$, that is equivalent to 2.50 m^3/s of total deficit (during 24 months), indicating that the solution with lowest f_1 value of the MoPSO-NS is very close to the optimal solution found by AcquaNet. This solution of MoPSO-NS has the value 2.33 $[m^3/s]^2$ equivalent to 4.61 m^3/s of total deficit (during 24 months). In MoDE-NS solution with the lowest value of f_1 was 83.94 $[m^3/s]^2$.

In multi-objective analysis each non-dominated solution contains the set of optimal values for each decision variable of the analyzed problem. Thus, when selecting a solution, a complete set of decision variables of the problem is obtained. As an example the Figure 4.34 shows the monthly flow values provided for the SPMR for the solutions listed in Table 4.18 and the solution found by AcquaNet. From the graph presented in Figure 4.34 it is possible to note that the deficits by MOEAs are distributed over the months while in AcquaNet they are concentrated in the last month of simulation. This behavior is associated with the nature of the difference between the objectives functions used, which is linear in AcquaNet and quadratic in our problem that uses MOEAs. On the other hand, if the aim is zero cost of pumping, the demand deficits for the SPMR will be maximum. The PCJ River Basin demands are always met.

Table 4.17 – Algorithms metrics for scenario f_1 vs f_3

Algorithm	**Metric**	**GD**	**iGD**	**ID**	**GRD**	**SP**
MoDE-NS	Mean	0.000065	0.00146	0.393	0.135	0.418
	Median	0.000065	0.00150	0.390	0.137	0.392
	Max	0.000066	0.00213	0.428	0.178	0.578
	Min	0.000063	0.00082	0.373	0.098	0.323

Algorithm	Metric	GD	iGD	ID	GRD	SP
	DP	0.000001	0.00042	0.018	0.028	0.094
MoPSO-NS	Mean	0.000065	0.00088	0.408	0.074	0.327
	Median	0.000065	0.00075	0.401	0.076	0.323
	Max	0.000067	0.00235	0.451	0.098	0.361
	Min	0.000063	0.00008	0.369	0.054	0.323
	DP	0.000001	0.00071	0.025	0.015	0.012
NSGA-II	Mean	0.000078	0.00701	0.536	0.651	0.359
	Median	0.000078	0.00678	0.536	0.648	0.323
	Max	0.000081	0.00808	0.560	0.688	0.684
	Min	0.000076	0.00597	0.513	0.612	0.323
	DP	0.000002	0.00068	0.013	0.023	0.114

Where DP é the standard deviation

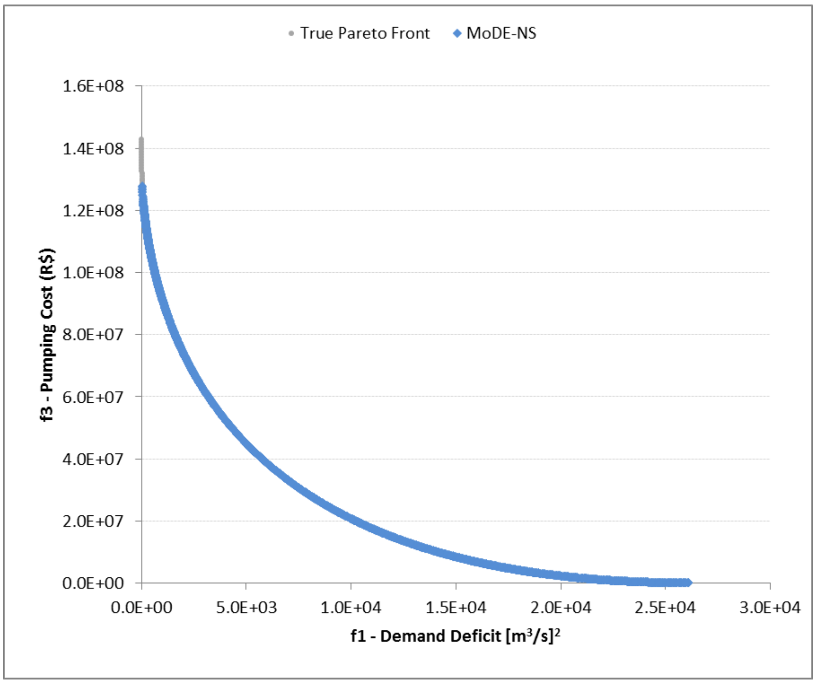


Figure 4.30 – Non-dominated solutions for scenario f_1 $_{vs}$ f_3 – MoDE-NS

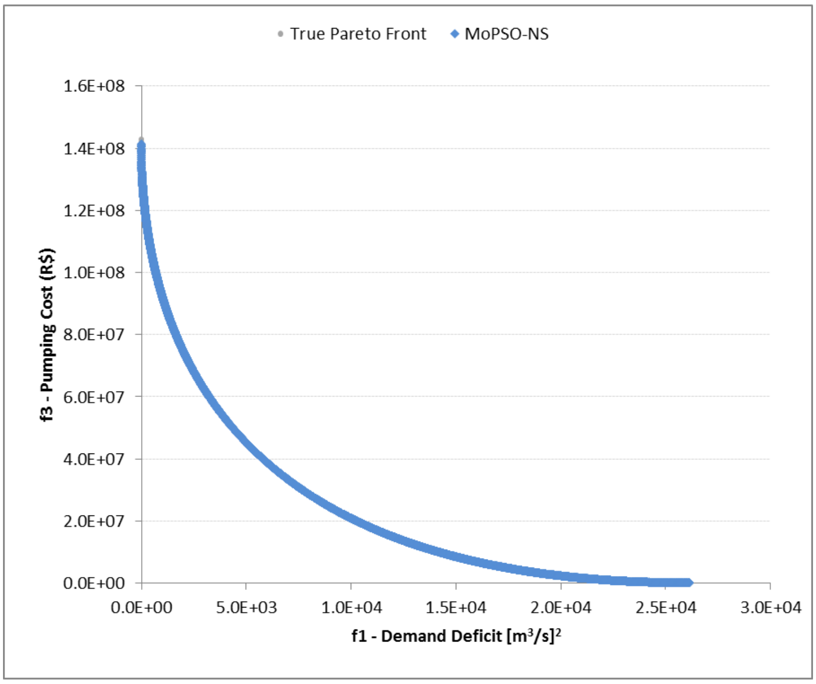


Figure 4.31 – Non-dominated solutions for scenario f_1 *vs* f_3 – MoPSO-NS

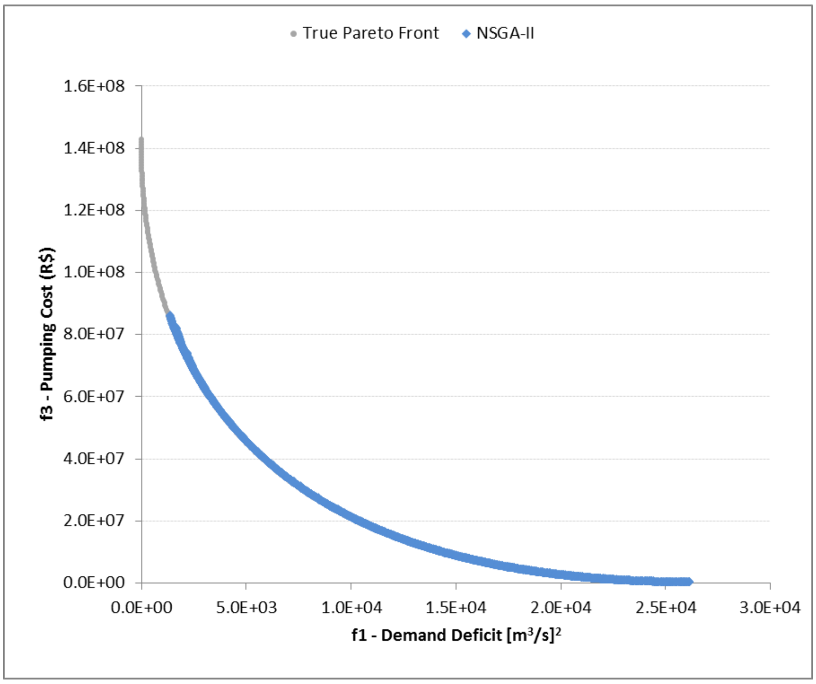


Figure 4.32 – Non-dominated solutions for scenario f_1 *vs* f_3 – NSGA-II

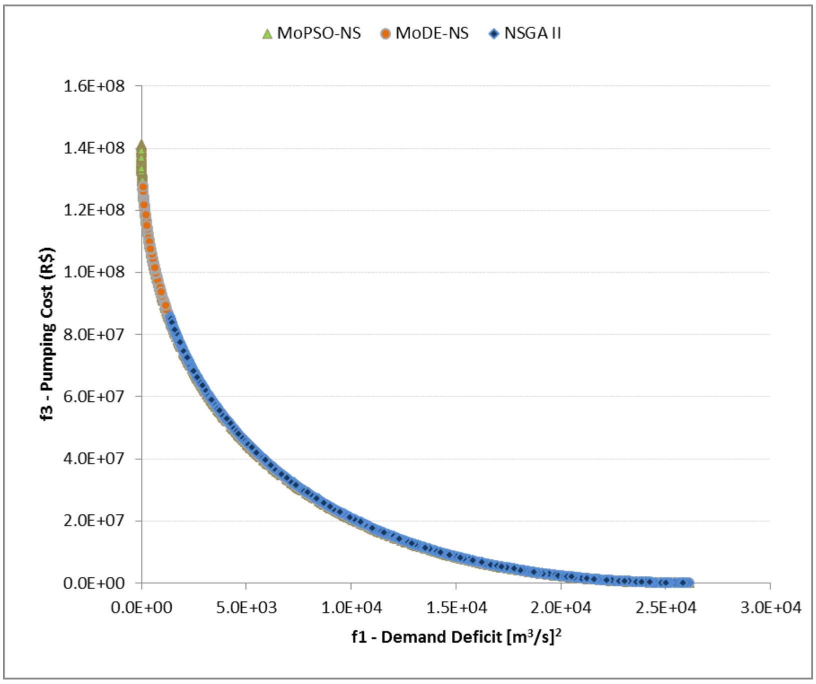


Figure 4.33 – Non-dominated solutions for scenario f_1 *vs* f_3 for three algorithms

Table 4.18 – Compromise solution for $\alpha(f_1) = 1$ e $\alpha(f_3) = 0$

Algorithm	f_1 $[m^3/s]^2$	f_3 (R$)
MoDE-NS	83.94	1.28E+08
MoPSO-NS	2.33	1.41E+08
NSGA-II	1365.36	8.58E+07

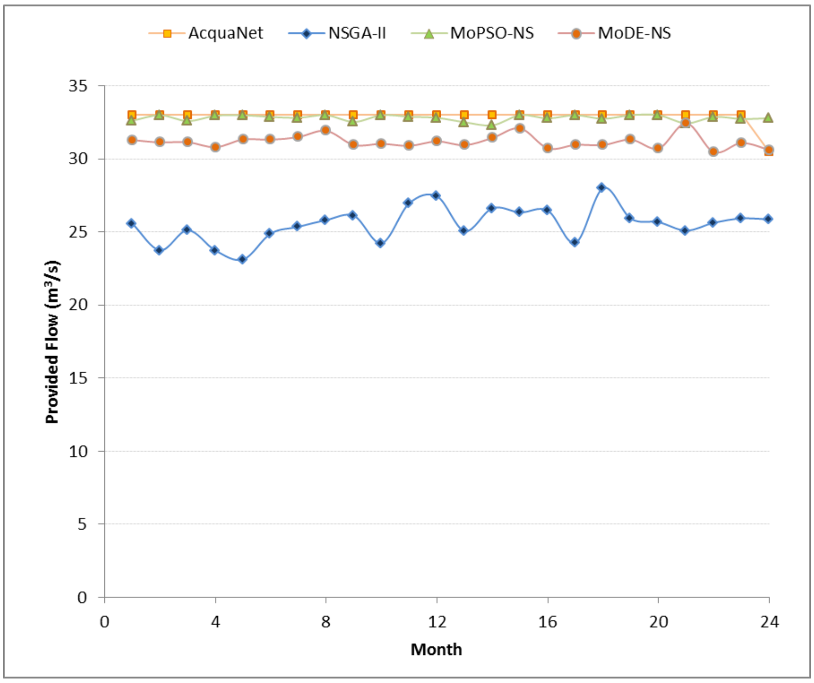


Figure 4.34 – Monthly flow provided to SPMR

Scenario 2: f_1 vs f_2

In this scenario is verified the relationship between the deficit of demand for the SPMR and the impact of water quality in the river Atibaia downstream of the reservoirs Cachoeira and Atibainha. The quality model used measures the deviation of the BOD value in relation to Class 2 at the point of comparison at check point N3 (Atibaia City), as Figure 4.28.

The graph in Figure 4.35 presents the comparison of non-dominated set of solutions of each algorithms for the simulation considering f_1 vs f_2. Analyzing the non-dominated solution sets in Figure 4.35, and the shape of the trade-off curve, allowing a small deficit on the demands supply could significantly improve the value of the objective function f_2, which measures the deviation of the BOD regarding Class 2. The gradient of the Pareto-optimal front defined by the non-dominated solutions is high for small deficits in the demands. But as the deficits of the demands increase, the rate of decrease of the function f_2 drops significantly. The analysis was performed considering a constant value and to the BOD of the downstream flow to the reservoirs Cachoeira and Atibainha (5 mg/L). This scenario was set up to highlight the trade-off between demands deficit and the impact on water quality downstream. In next item two simulations are presented with lower values of BOD (3 and 1 mg/L) for downstream flow to the Cachoeira and Atibaia reservoirs. It is possible to verify that the Pareto-optimal fronts maintain the same shape as the curves presented on this simulation, as shown in Figure 4.35. The same should happen when employing a more detailed quality model such as Qual2e.

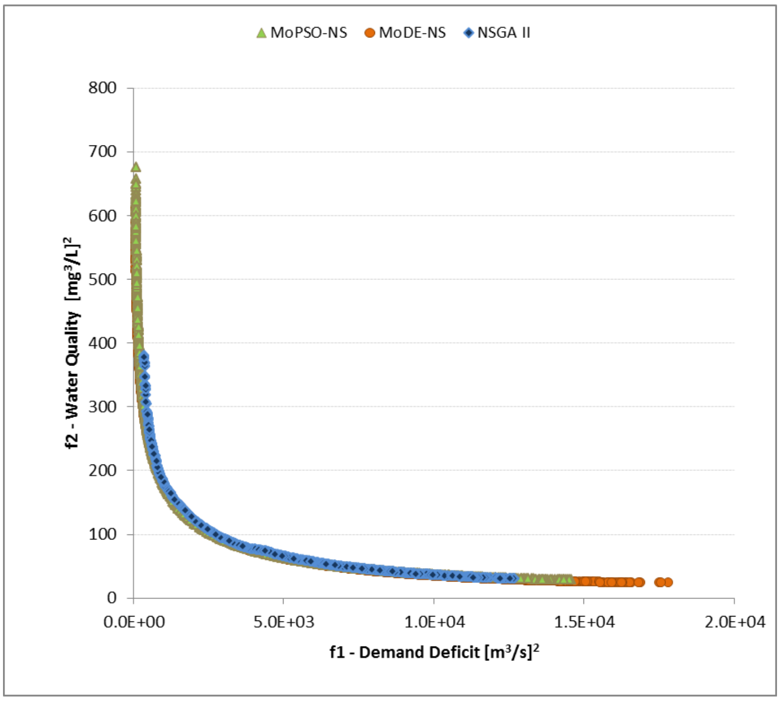


Figure 4.35 – Non-dominated solutions for scenario f_1 vs f_2 for three algorithms

Table 4.19 shows the values of best compromise solutions using $\alpha(f_1) = 0.5$ and $\alpha(f_2) = 0.5$ which is obtained with a demand deficit for SPMR of about 2000 $[m^3/s]^2$ which is equivalent to approximately 1.8 m^3/s average monthly deficit. The graph of BOD to this solution at point N3 is shown in the graph of Figure 4.36, whose value is around 7 mg/L in all months of the simulation.

Considering now the solution with best value of BOD, in which the demand deficit of the SPMR is maximum (demand very close to zero, as in Table 4.20), the value of BOD is about 6 mg/L in all months, as comparison in Figure 4.36.

Table 4.19 – Compromise solution for α(f_1) = 0.5 e α(f_2) = 0.5

Algorithm	**f_1 – [m³/s]²**	**f_2 - [mg/L]²**
MoDE-NS	2314.81	106.67
MoPSO-NS	1917.01	121.60
NSGA-II	1998.19	123.66

Table 4.20 – Compromise solution for α(f_1) = 0 e α(f_2) = 1

Algorithm	**f_1 – [m³/s]²**	**f_2 - [mg/L]²**
MoDE-NS	17791.01	24.45
MoPSO-NS	16632.53	24.59
NSGA-II	14464.51	30.04

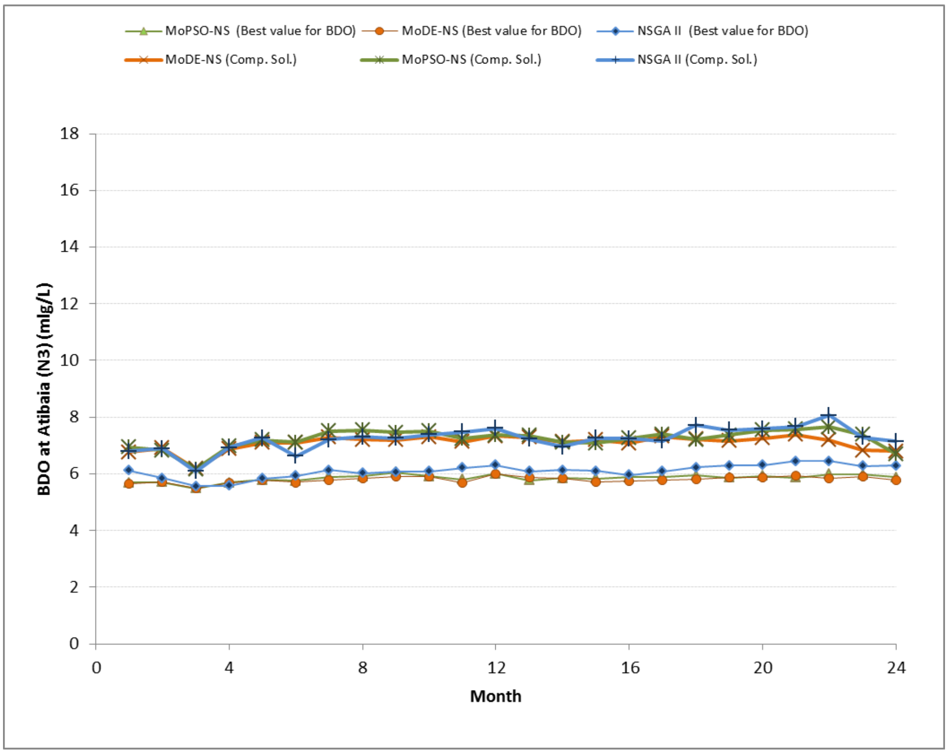


Figure 4.36 – 1. Value of BOD at N3 for Best Compromise solution with α(f_1) = 0.5 and α(f_2) = 0.5 (Comp. Sol.); 2. Solutions with best value of BOD - α(f_1) = 0 e α(f_2) = 1 (Best value for BOD)

In these two scenarios, the number of simulations required to obtain the non-dominated set of solutions is quite high (50,000). Still, given the set of solutions is not ideally distributed in Scenario 2 and MOEAs fail to determine extreme solutions. In Scenario 1 the distribution of solutions is better, but again the MOEs fail to determine extreme non-dominated solutions, especially the NSGA-II.

The results show that MOAEs have difficulties with complex and problems with high number of variables and constraint, and require a very large number of generations to achieve the non-dominated set of solutions. To overcome this limitation an integrated analysis with the AcquaNet is presented in the next section.

This analysis highlights another important feature of the multi-objective analysis, which is the ability to capture the trade-offs between two conflicting objectives, that is perhaps the most value information, regarding than choosing a single solution of the set of non-dominated solutions determined by the algorithm

4.4 Case 3: Integrated analysis: MOEAs and Networkflow models

In this item multi-objective analysis was performed using the AcquaNet as a tool for simulation and optimization associated with the MOEAs. Thus the multi-objective model has only three variables: the demand for the SPMR, downstream flow of the Cachoeira and Atibainha reservoir. The objective functions are the same as in Scenario 2: minimization of demand deficits (f_1) and the environment in Atibaia river N3 (f_2). In this structure no constraint are required for the MOEAs, except for the upper and lower limits on variables.

In this methodology, each at each generation, AcquaNet is executed and solves the flow network N times, where N is the population size. In this process, the values of the demands are defined by MOEAs and used in Out-of-Kilter to solve the network flow, resulting in the optimal flow on each arc / link. These flows are then used by the DSS to perform the evaluation of the objective function and in generation of new population. This process is repeated until the maximum number of generations reach. For better understanding of the process, a flowchart is shown in Figure 4.37.

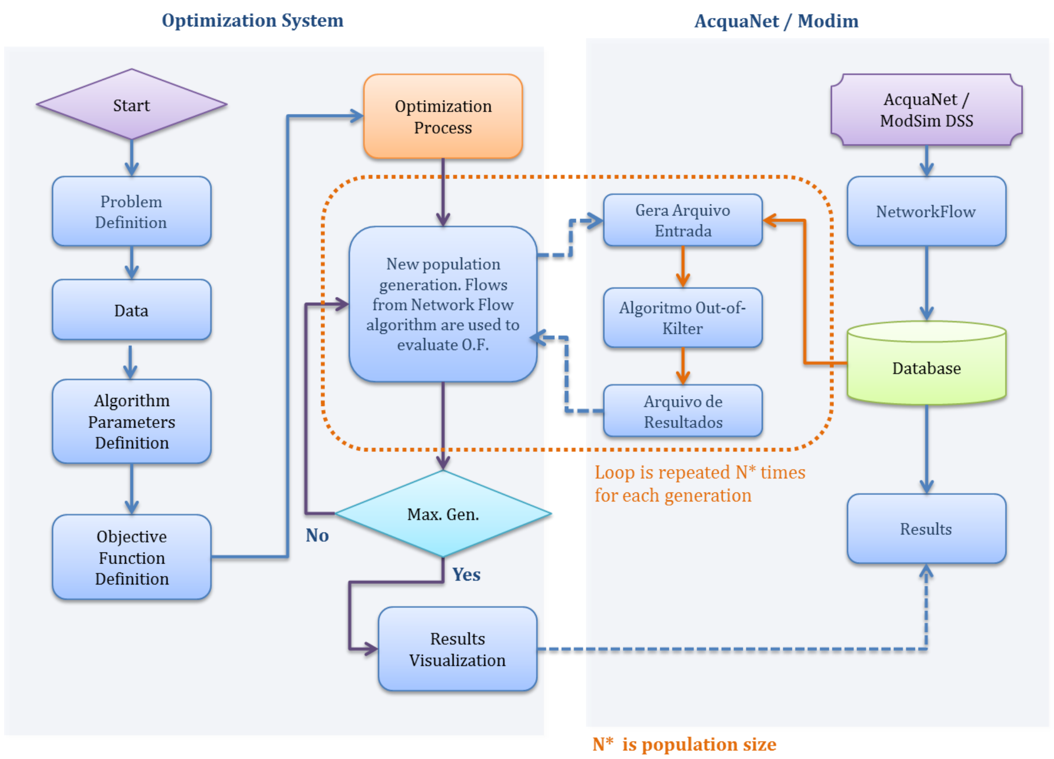


Figure 4.37 – Flowchart for M.O. analysis using MOEAs + AcquaNet DSS

The network flow was built in AcquaNet (Figure 4.38) base on schematic of Figure 4.28 and the same operational data of Case two is Case 2.

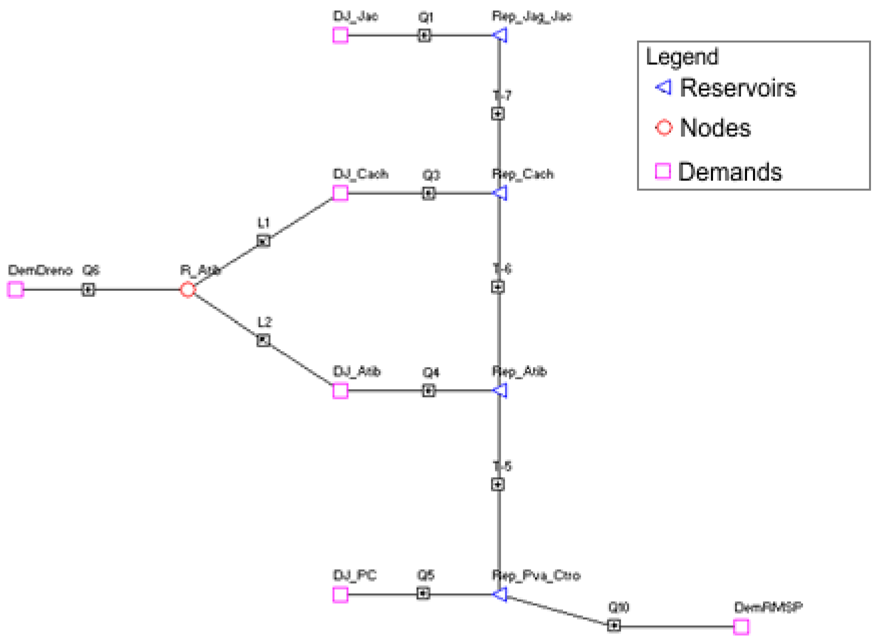


Figure 4.38 –AcquaNet DSS networkflow representation

The three decision variables are represented by demand nodes: "DemRMSP", "DJ_Cach", "DJ_Atib" as the flow network of Figure 4.38. The value of demand is assumed constant throughout the simulated period. Demand node "DemDreno" is kept high so that the entire surplus in the system is directed to the arcs R1 (Q3) and R2 (Q4). AcquaNet uses a priority scheme, and they are set so that the following conditions are met:

1. The minimum flows downstream of the reservoirs have the highest priority (value 1) and are represented in the networkflow by "DJ_Jac", "DJ_Atib", "DJ_Cach" and "DJ_PC";
2. The target volume of the reservoirs have priority (value 90) but slightly higher than the demand "drain", in a way that the excess in system is stored in reservoirs;
3. The priority for the demand "DemSP" must be higher than the priority for minimum downstream flows of the reservoirs, and therefore is set to value 5.
4. Demands "DJ_Cach" and "DJ_Atib" are also variables in the model, and are responsible for controlling the flow on links R1 (Q3) and R2 (Q4) and directly impact the objective function relating to water quality, and its upper limits are

defined as 25 and 15 m^3/s respectively.

5. The demand node "Dreno" has the lowest priority (value 99) and is used to receive excess water in the system. This node is required by the characteristics of Out-of-Kilter algorithm.

It is important to note that the scale of priorities in AcquaNet is 1 to 99, with 1 being the highest priority and 99 the lowest. The limits of variation of demand for SPMR are defined from 0 to 33 m^3/s. The limits of the demands "DJ_Cach" and "DJ_Atib" are defined between 0.5 to 25 m^3/s and 0.5 to 15 m^3/s, respectively

As the number of variables is small, a smaller population can be used ,as well as few generations. This problem has no restrictions, only lower and upper limits on the variables. The population size was set to 100 individuals and 250 generations. In this configuration, the problem is solved 100 times for each generation by Out-of-Kilter resulting in out 25,000 simulations of the network flow. In this analysis the same settings where used as Case 2 for comparison porpoise. MoDE-NS used to perform integrated multi-objective analysis and results are identified as AcquaNet-MoDE-NS.

Initially the MO analysis was performed to compare results with Case 2. The non-dominated sets of solutions generated in Case 2 using MoDE-NS and in this case are shown in Figure 4.39. This graph show that the Pareto-optimal fronts are practically coincident and extreme solutions are achieved more efficiently using the integrated analysis (AcquaNet MOEA, MoDE-NS in this case). Table 4.21 presents the solutions with best (lower) value for BOD for both methods. The objective function value using the AcquaNet+MoDE-NS is “better” than those from Case 2, using only the MoDE-NS. This difference can be explained by the fact that the MOEAs are not able to determine the extreme solutions where the highest demand deficits and consequently BOD has lower values. The Out-of-Kilter algorithm, AcquaNet’s networkflow algorithm, has no “knowledge” about how the values of the flows on links R1 and R2 affect the objective function of the multi-objective model.

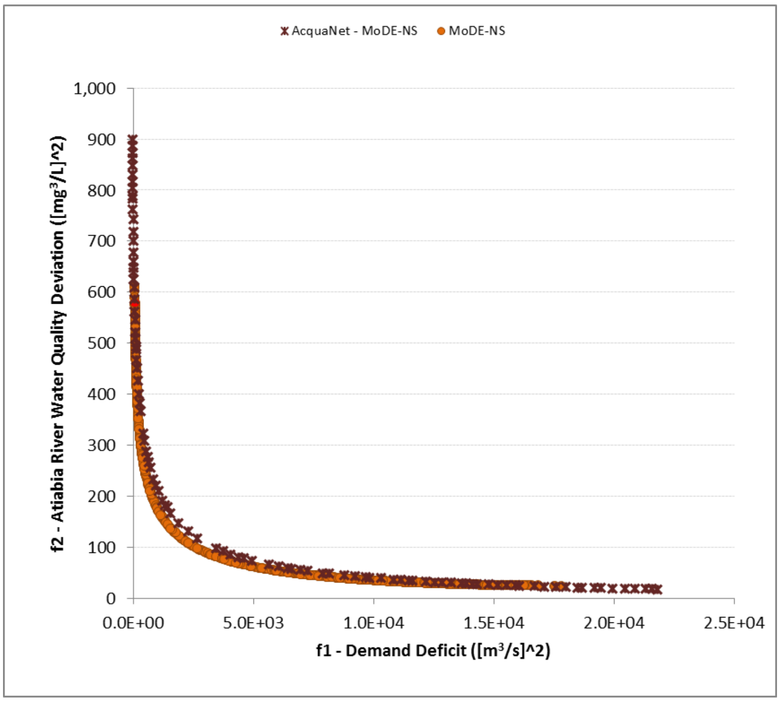


Figure 4.39 – Non-dominated solutions for scenario f_1 vs f_2, MoDE-NS and AcquaNet+MoDE-NS

Table 4.21 – Solutions with best values for f_2 for both MoDE-NS and AcquaNet+MoDE-NS

Algorithm	**f_1 – $[m^3/s]^2$**	**f_2 - $[mg/L]^2$**
MoDE-NS	17791.01	24.45
AcquaNet+MoDE-NS	21585.63	17.55

As discussed in the previous section, the simulation considering BOD = 5 mg / L for downstream flow of the Cachoeira and Atibainha reservoirs, can be too unfavorable. Thus, two other simulations were performed considering BOD = 3 mg / L and BOD = 1 mg / L for downstream o flow of the reservoirs Atibainha and Cachoeira, whose non-dominated solution sets are shown in Figure 4.40. Even considered BOD = 1 mg/L for downstream flow of the reservoirs, it is necessary to allow a deficit of 2,500 $[m^3/s]^2$ (equivalent to a monthly deficit of 2.08 m3 / s) so that the BOD value at node N3 is within the limit of Class 2 (i.e. $f_2 = 0$). For the simulation with BOD = 3 mg/L equivalent to average monthly deficit of 4.16 m3/s would be necessary in order to have $f_2 = 0$. This analysis is simplified, and the water flow scenario considered was a period with flows below historical average. Only a small stretch of the river was considered Atibaia. The aim was to explore the ability of the MOEA and integration with AcquaNet for determining the non-dominated solutions.

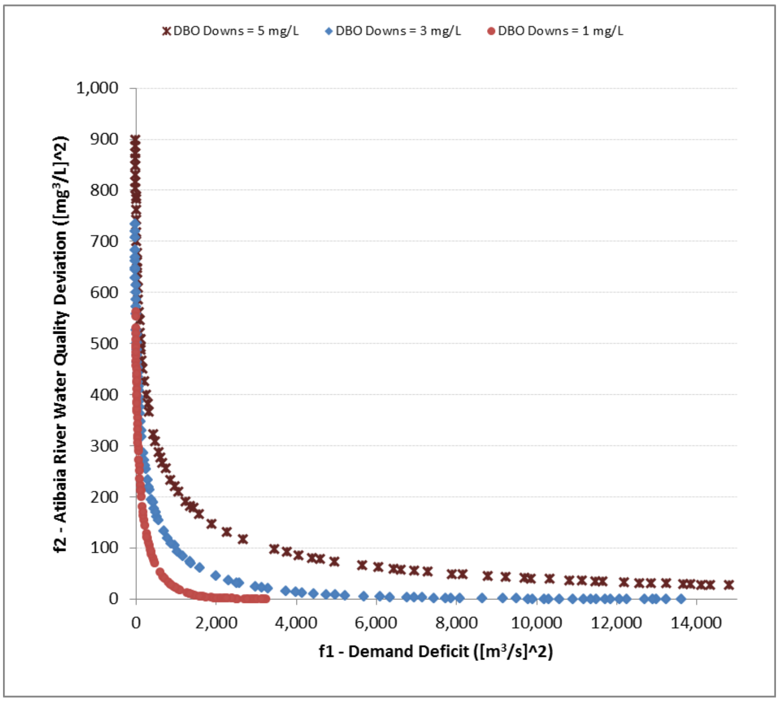


Figure 4.40 – Non-dominated solutions for three different BOD scenarios for downstream flow {1, 3, 5} mg/L

4.5 Case 4: Simplified analysis with three objectives

In this session a simplified problem was considered in which the only one reservoir (Jaguari/Jacarei Res.) was included in the Multi-objective with three objectives: 1 - transfer flow of 28 m^3/s by Tunnel 7 (for SPMR demand), 2 - withdraw of 10 m^3/s to other demands (irrigation, downstream flow, etc.) and 3 – maximization of flood space for flood control (which as limited to a maximum of 400 hm^3, approximately half the live volume of

Jaguari/Jacarei Reservoir. These three objectives are conflicting and presented in the equation .

$$\min f_1 = \sum_{t=1}^{n}\left(28 - Q_{1,t}\right)^2$$
$$\min f_2 = \sum_{t=1}^{n}\left(10 - Q_{2,t}\right)^2 \tag{4.29}$$
$$\min f_3 = \sum_{t=1}^{n}(400 - S_t)^2$$

Where $Q_{1,t}$ is the flow transferred through tunnel 7, $Q_{2,t}$ is the flow for other demands (downstream, irrigation, and others) and S_t is the reservoir volume at the of the time period *t* (month).

For this scenario one year simulation period with the natural flows of the Dec/1952 Jan/1952 was used. The initial volume of the reservoir was set to 20% of total storage. In this analysis the graphical results are presented only for MoDE-NS, because the result is identical for the other two algorithms. In this analysis is considered only one reservoir and a 12-month simulation period, and therefore the population size and number of generations required is less than in Case 2. For the optimization, a population of size 1,000 and 1,000 generations was used. In this configuration the problem has 36 variables and 12 constraints (one for each month reservoir mass balance). The population size of 100 individuals would be sufficient in this scenario, however, the value of 1,000 was used for a better visualization of the trade-offs between objectives. This is a limitation of the structure of the MOEA used in this report, i.e., the maximum number of non-dominated solutions found is limited by the size of the population. In problems where the Pareto surface has to be very well represented, it is necessary to increase the size of the population to obtain a higher density of points (solutions).

The graph in Figure 4.41 presents the non-dominated set of solutions obtained using MoDE-NS. The results are presented as a three dimensional surface that can be viewed on the

Trade-off Graph (Figure 4.42 along with the best compromise solutions, highlighted in green, with $\alpha = 0.5$ for the three objectives. The values of the compromise solutions obtained for $\alpha = 0.5$ are presented in Figure 4.42 for the three algorithms. The solutions obtained are similar, indicating that none of the algorithms had difficulty in obtaining non-dominated set of solutions for this problem.

This analysis demonstrates that the integration of MOEAs with networkflow models like AcquaNet and ModSim has great potential, since the MOEAs had difficulties in finding non-dominates solutions for problems with high number of variables and constraints.

The variables and objective functions are defined in the Multi-objective model and the MOEAs are responsible for the process of finding non-dominated solutions, and the mass balance in reservoirs, as well as the optimal flow in the system is determined by the network flow algorithm for each evaluation of the objective function. The great advantage of this methodology is the possibility of multi-objective analysis of very complex systems with extensive simulation horizons.

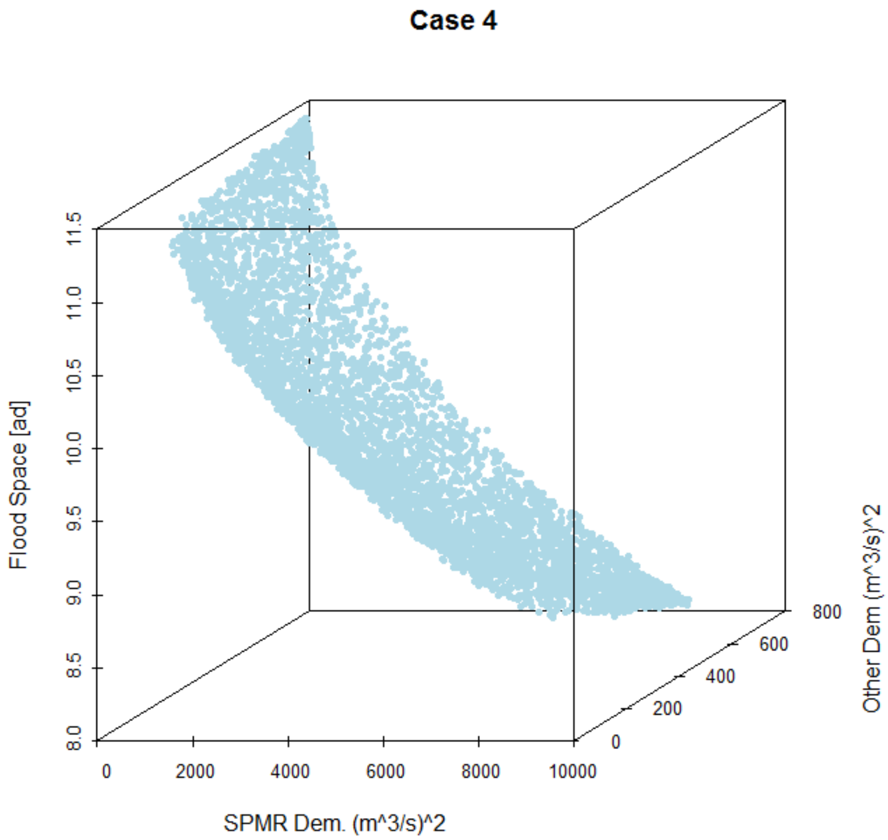


Figure 4.41 – 3D graph for non-dominates solutions for three objectives using MoDE-NS

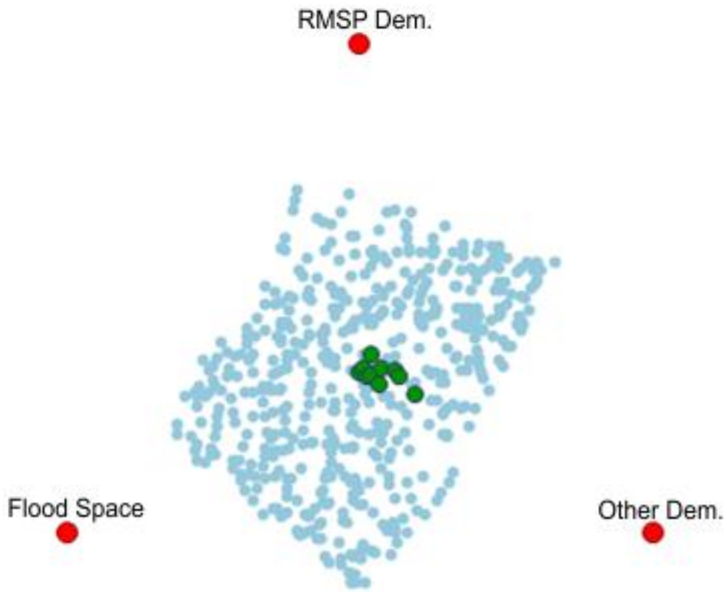


Figure 4.42 – Trade-off graph for three objectives using MoDE-NS with α = 0,5

Table 4.22 – Compromise solutions for three objectives and algorithms with α = 0,5

	MoDE-NS	MoPSO-NS	NSGA-II
Dem. RMSP $[m^3/s]^2$	3744,58	3750,11	3765,20
Dem. Outros $[m^3/s]^2$	426,21	521,22	534,21
Vol. Esp. [ad]	9,63	9,57	9,42

In this section three objective function where considered in the multi-objective analysis and a small number of periods (12 months) was used. Also only one reservoir (Jaguari/Jacarei reservoir) was used. In this configuration, MOEAs are able to easily find the non-dominated solutions set.

The aim of the application of MOEAs in such a simplified configuration was to show how a higher number of variables and constrains heavy affects the performance of such algorithms, especially regarding the population size and number of generations needed to accomplish multi-objective analysis goals. Case 2 has 5 times more variables and 8 times more constraints than this analysis. While in Case 4, 1,000 generations where needed, in Case 2, the number of generations needed was at least 50,000.

4.6 Result analysis

In this section the Decision Support System and the MOEAs were applied to standard test problems, to hydrological model and to a water supply system (Cantareira system, Sao Paulo – Brazil). The algorithms presented good results in all applications. The results obtained with the algorithms MoDE-NS and MoPSO-NS are, in general, better than the NSGA-II. This can be seen by comparing the performance indicators and visual analysis of non-dominated solutions.

In the application to test problems and hydrological model Smap (Case 1) the three algorithms produced very satisfactory results, i.e., non-dominated solutions are found without much difficulty.

The algorithms had more difficult to find non-dominated solutions for Case 2, in which a large number of generations were necessary for a satisfactory coverage of the Pareto front. In this simulation all variables in the simulated system were calculated using the MOEAs. NSGA-II presented the worst performance among the three algorithms, whose non-dominated set of solutions was not properly distributed in the Pareto-optimal front, by comparing the goals of minimizing the demand deficit of f_1 and minimizing the cost of pumping f_3.

In Case 4 a simplified configuration was adopted by considering one of the reservoirs Cantareira system with a smaller number of months of simulation and three objectives. Case 2 has 198 variables and 96, while Case 4, 36 variables and 12 constraints, a 5:1 ratio in number of variables, and 8:1 in the number of restrictions. Case 2 a large number of generations (50,000) and a population of size 500 where necessary for satisfactory results. In Case 4 a population size of 1,000 and only 1,000 generations were required to obtain the non-dominated solutions.

In Case 3 a multi-objective analysis was performed using the AcquaNet (Networkflow model) as a tool for simulation and optimization associated with the MOEAs. As the networkflow model was responsible for calculating the optimal flows and mass balance, a fewer number of generations and a smaller population size was required. The set of non-dominated solutions obtained by this method are very well distributed and very close to the set obtained with the MOEAs applied in Case 2.

The processing time, using a computer with an Intel I5, were:

Case 2: for each simulation took about 15 minutes to complete the 50,000 generations, for both scenarios.

Case 3: In this case the processing time required for each simulation was about 3 seconds, with a total of 10 minutes. For each evaluation of the Objective Function (OF) a complete simulation of the networkflow is necessary. The version used in this report is the DOS version whose data input and output are handled via text files. Most of the simulation time is used for processes of writing and reading the text files. The new version in development does not use files in text format and the processing time is expected be drastically reduced for this type of analysis.

5 Conclusions and recommendations

This report presents an application of evolutionary algorithms in multi-objective analysis for water resource management, as well as their integration into decision support systems such as AcquaNet and ModSim. Three multi-objective algorithms derived from Differential Evolution (MoDE-NS), Particle Swarm Optimization (MoPSO-NS) and Genetic Algorithm (NSGA-II) are used, which were implemented in the form of a decision support system that is applied to standard test problems, calibration of hydrological model and a complex water resources system.

The DSS developed represents an extremely flexible tool that allows multi-objective analysis of generic problems with the possibility of integration with DSS such as AcquaNet ModSim, either by importing data networkflow from them or by direct integration, taking advantage of the network flow algorithms in solving large problems.

Results of application to standard test problems and case studies show that the MoPSO-NS and MoDE-NS presented slightly better results than NSGA-II. The non-uniform mutation operator used in the algorithms is important for better results, especially for application in more complex water resources systems, like the one used on our case study. The elitism operator is only important for the MoPSO-NS.

The computation structure (population structure and generation of offspring) used in MoDE-NS and MoPSO-NS is derived from the NSGA-II. This structure has proved very efficient in both algorithms, especially for MoPOS-NS, which results in some of the problems outperformed MoDE-NS and NSGA-II. In the MoDE-NS, it is important to note that the only strategy capable of producing good results, when applied to more complex problems, is DE/current-to-rand.

The MoDE-NS and MOPSO-NS, along with the NSGA-II represent the state of the art available in MOEA, and its application is recommended for multi-objective analysis of water resources systems, since the problem does not have a large number of variables and / or restrictions. Case Study 2 showed that more efficient evolutionary algorithms are still needed.

The limitation of the MOEAs in handling problems with large number of variables and constraints can be overcome with the integration to extremely network flow algorithms efficient as the Out-of-Kilter of AcquaNet or Relax IV of ModSim, as demonstrated in Case Study 3. This integration is also shows the great potential of MOEAs, since user can set any kind of objective function from the variables determined by the optimal network flow algorithm.

MOEAs are undoubtedly an important and interesting alternative to the so call “classical” methods, but it is important to consider their limitations, mainly on the number of variables and constraints. Thus, smaller problems can be easily solved by the use of MOEA presented on this report. The Trade-off Graph (TG) is a very useful and interesting tool for visualization and analysis of non-dominated solutions, especially when the number of objectives is higher than three. The TG combines a coordinate system method and compromise programming to highlight the best or most robust solutions based on weights defined by the decision maker.

The flexible nature of multi-objective evolutionary algorithms discussed in this report and DSS developed represent important tools for managers and decision makers for management in complex systems considering multiple uses and explore potential solutions to conflicts arising from water use.

6 References

ADEYEMO, J. A. Reservoir Operation using Multi-objective Evolutionary Algorithms-A Review. Asian Journal of Scientific Research, v. 4 (1). p.16-27. 2011

ABBASS, H.A.; SARKER, R. The Pareto Differential Evolution Algorithm. International Journal on Artificial Intelligence Tools 11(4), 531–552. 2002.

ANA. AGÊNCIA NACIONAL DE ÁGUAS. Nota Técnica nº 353/SOC - Restrições de usos de recursos hídricos em função das regras de operação do Sistema Cantareira. Brasília: ANA, 2004.

BALTAR, A. M. 2007 Use of Multi-Objective Particle Swarm Optimization in Water Resource Management. 2007. 263 p. Tese (Doutorado) - Department of Civil and Environmental Engineering - Colorado State University, Fort Collins, Colorado

BENITEZ, J.E.A., EVERSON, R.M.; e FIELDSEND, J. E. A MOPSO Algorithm Based Exclusively on Pareto Dominance Concepts. Proc. of Evolutionary Multi-Criterion Optimization Third International Conf., BerlinÇ Springer-Verlag, 459-473. 2005

BRASIL. Lei nº 9.433, de 8 de janeiro de 1997. "Institui a Política Nacional de Recursos Hídricos, cria o Sistema Nacional de Gerenciamento de Recursos Hídricos, regulamenta o inciso XIX do art. 21 da Constituição Federal, e altera o art. 1º da Lei nº 8.001, de 13 de março de 1990, que modificou a Lei nº 7.990, de 28 de dezembro de 1989. DOU. Brasília, 09.01.1997.

CHAKRABORTY, U. K. (ed.) Advances in Differential Evolution, Berlin: Springer-

Verlag, 2008. 340 p. (Studies in Computational Intelligence, Vol. 143).

CHAPRA, S. C. Surface Water Quality Modeling. USA: McGraw-Hill. 844p. 1997.

CHIU S-Y et al. Cross-searching strategy for multi-objective particle swarm optimization. Evolutionary Computation, 2007. IEEE Congress on , vol., no., pp.3135-3141, 25-28 Sept. 2007

COBRAPE - Companhia Brasileira de Projetos e Empreendimento. Plano das Bacias Hidrográficas dos Rios Piracicaba, Capivari e Jundiaí 2010 a 2020. São Paulo. 2010.

COELLO COELLO C. A.; Lamont G. B.; Veldhuizen D. A. V., Evolutionary Algorithms for Solving Multi-Objective Problems, 2st ed. Berlin: Springer-Verlag.. 810 p. 2007

COELLO COELLO, C.A.., e LECHUGA, M. S. MOPSO: A Proposal for Multiple Objective Particle Swarm Optimization. Proc. Congress on Evolutionary Computation, IEEE, Piscataway, NJ, Vol. 2, 1051-105. 2002

COELLO COELLO, C. A., PULIDO, G. T., LECHUGA, M.S. "Handling Multiple Objectives with Particle Swarm Optimization." IEEE Transactions on Evolutionary Computation, IEEE, Piscataway, NJ, 8 (3) 256-279. 2004.

COELLO COELLO C. A.; DHAENENS, C.; JOURDAN, L. (Ed.) Advances in Multi-Objective Nature Inspired Computing, 1st ed. Berlin: Springer-Verlag, 200 p. (Studies in Computational Intelligence, Vol. 272). 2010

CUNDERLIK, J. M. e SIMONOVIC, S. P. Calibration, verification, and sensitivity analysis of the HEC-HMS hydrological model. Water Resources Research Report No. 048, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada. 2004

CUNDERLIK, J. M. e SIMONOVIC, S. P. Hydrological extremes in a southwestern Ontario river basin under future climate conditions. Hydrological Sciences Journal. 50 (4), 631-654. 2005

DAEE, Secretaria De Estado De Saneamento E Energia Departamento De Águas E Energia Elétrica Diretoria De Procedimentos De Outorga E Fiscalização. Nota Técnica DAEE/DPO – Sistema Cantareira - Recomendações para manutenção de volumes de espera nos reservatórios dos aproveitamentos do Sistema Cantareira, 2009.

DEB, K. E AGRAWAL, R. B. Simulated binary crossover for continuous search space. Complex Systems, 9115-148. 1995

DEB, K. An introduction to genetic algorithms. Sadhana. 24 (4), 293-315. 1999.

DEB K., AGRAWAL S., PRATAP A. E MEYARIVAN T. A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimisation: NSGA-II, Proceedings of the 6th International Conference on Parallel Problem Solving from Nature, p.849-858, September 18-20, 2000.

DEB K., AGRAWAL S., PRATAP A. E MEYARIVAN T, A fast and elitist multiobjective genetic algorithm: NSGA-II. Evolutionary Computation, vol. 6, no. 2, pp. 182–197, April 2002.

DEB, K. e BEYER H.-G. Self-Adaptive Genetic Algorithms with Simulated Binary Crossover. Evolutionary Computation. 9 (2) - Massachusetts Institute of Technology. 2001

DEB, K. Introduction to Evolutionary Multiobjective. In: Multiobjective Optimization: Interactive and Evolutionary Approaches. BRANKE, J. et al. (ed.). Berlin: Springer-Verlag, 2008. 416 p.

DEB, K. Multi-Objective Optimization Using Evolucionary Algorithms, United Kingdom: John Wiley & Sons. 2009. 515p.

FONSECA, C.; FLEMING, P. Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In: Forrest, S., ed. Proceedings of the Fifth International Conference on Genetic Algorithms, University of Illinois at Urbana Champaign, San Mateo, California: Morgan Kauffman Publishers, p. 416–423. 1993

GOLDBERG, D. E. Genetic Algorithms in Search, Optimization and Machine Learning, New York : Addision-Wesley, New York. 1989. 412 p;

HOLlAND, J. H. Adaptation in Natural and Artificial Systems. Ann Arbor: University of Michigan Press. 1975

JUNG, B. S. e KARNEY, B. W. Hydraulic Optimization of Transient Protection Devices Using GA and PSO Approaches. J. Water Resour. Plng. and Mgmt., ASCE. Reston, VA, 132 (1) 44-52. 2006

. KENNEDY, J. and EBERHART, R. Particle Swarm Optimization. Proc. 4th IEEE Int. Conf. on Neural Networks, IEEE, Piscataway, NJ, 1942-1948. 1995

KIM T., HEO J-H E JEONG C-S. Multireservoir system optimization in the Han River basin using multi-objective genetic algorithms. Hydrological Processes, Vol. 20, No. 9, pp. 2057-2075. 2006.

KUKKONEN, S., LAMPINEN, J. GDE3: The third Evolution Step of Generalized Differential Evolution. In: 2005 IEEE Congress on Evolutionary Computation (CEC 2005), Edinburgh, Scotland, September 2005, vol. 1, pp. 443–450. IEEE Service Center (2005)

KWAN, C., YANG, F., CHANG, C. A Differential Evolution Variant of NSGA-II for Real World Multiobjective Optimization, ACAL 2007, LNAI 4828, pp. 345–356, 2007. Springer-Verlag Berlin Heidelberg, 2007

HORN J., NAFPLIOTIS N. e GOLDBERG D. E., A niched Pareto genetic algorithms for multiobjetcive optimization,” in Proc. 1st IEEE Cong. Evolutionary Computation pp. 82-87, 1994.

LABADIE, J. W. MODSIM: Decision support system for river basin management. Documentation and user manual, Dept. of Civil Engineering, Colorado State University., Fort Collins, 2010. 130 p.

LABADIE, J. W. Optimal Operation of Multireservoir Systems: State-of-the-Art Review. Journal of Water Resources Planning and Management, v. 130, n. 2, p. 93-111, 2004.

LECHUGA, M. S. e ROWE, J. E. Particle Swarm Optimization and Fitness Sharing to Solve Multi -Objective Optimization Problems. Proc. of IEEE Congress on Evolutionary Computation CEC'05, IEEE, Piscataway, NJ, 1204-1211. 2005.

LI, X. A Non-dominated Sorting Particle Swarm Optimizer for Multiobjective Optimization. Proc. of the Genetic and Evolutionary Computation Conf. GECCO. Springer-Verlag, 37-48. 2003.

LOPES, J. E. G., BRAGA, B. P. F. e CONEJO, J. G. L. SMAP: A simplified hydrologic model. In: Applied Modeling in Catchment Hydrology Water Resources Publications, Littleton, Co. p. 167–176. 1982

LOUCKS, D.P.; BEEK, E. van, Water resources systems planning and management. An introduction to methods, models and applications, Studies and Reports in Hydrology. Paris: UNESCO Publishing, 2005.

MELLO JUNIOR, A. G.; FAGA, M. T. W. Uso de motores de combustão interna a gás natural em substituição aos motores elétricos nas instalações de centrais de bombeamento de água. V congresso nacional de engenharia mecânica. 25 a 28 de agosto de 2008, Salvador, Brasil. 2008.

NICKLOW et al. State of the Art for Genetic Algorithms and Beyond in Water Resources Planning and Management. ASCE Task Committee on Evolutionary Computation in Environmental and Water Resources Engineering, J. Water Resour. Plng. and Mgmt. 136. 412. 2010.

PORTO, R. L; AZEVEDO, L. G. T. Sistema de Suporte a Decisão Aplicado a Problemas de Recursos Hídricos. In: Técnicas quantitativas para o gerenciamento de recursos hídricos. Org. por Porto, R. L., ABRH, ed. UFRGS, Porto Alegre - RS, 1997. pp. 43 – 95p.

PORTO, R. L. L. et al. Sistema de suporte a decisão para análise de sistemas de recursos hídricos. In: Métodos Numéricos em Recursos Hídricos 6. Org. por da Silva, R. C. V., ABRH, ed. UFRGS, Porto Alegre - RS, 2003. pp. 93 – 240p.

PORTO, R. L. L. AcquaNet: Arquitetura, Estratégias e Ferramentas. In: VXI Simpósio Brasileiro de Recursos Hídricos, João Pessoa PB, 2005.

PRASAD T. D. e PARK, S. Multiobjective Genetic Algorithms for Design of Water Distribution Networks JOURNAL OF WATER RESOURCES PLANNING AND MANAGEMENT ASCE / JANUARY/FEBRUARY / 73, 2004

PRICE, K. V., STORN, R. M. e LAMPINEN, J. A. Differential evolution: A practical approach to global optimization. Berlin: Springer. 2005.

PRICE, K. V. e STORN, R. M. Differential evolution - A simple evolution strategy for fast optimization. Dr. Dobb's Journal. 22, 18-24. 1997

PULIDO, G.T. (2005). On the Use of Self-Adaptation and Elitism for Multiobjective Particle Swarm Optimization. 2005. 259 p. (Tese Doutorado). Electrical Engineering Department Centro de Investigación y Estudios Avanzados del Instituto Politécnico Nacional, Mexico City, Mexico

RAQUEL, C. R. e NAVAL P.C. An Effective use of Crowding Distance in Multiobjective Particle Swarm Optimization. Proceedings of Genetic and Evolutionary Computation Conference (GECCO 2005), Washington, D.C., June 25-29, 2005.

REDDY J. e KUMAR N. Multiobjective Differential Evolution With Application To Reservoir System Optimization. Journal Of Computing In Civil Engineering, Asce, 2007.

REGULWAR D. G., CHOUDHARI, S. A. E RAJ P. A. Differential Evolution Algorithm with Application to Optimal Operation of Multipurpose Reservoir. J. Water Resource and Protection, 2, 560-568. 2010.

REIS, L. F. R., WALTERS, G. A., SAVIC, D. e CHAUDHRY, F. H. Multi-Reservoir

Operation Planning Using Hybrid Genetic Algorithm and Linear Programming (GA-LP): An Alternative Stochastic Approach. Water Resources Management (2005) 19: 831–848.

REIS, L. F. R., BESSLER, F. T., WALTERS, G. A. e SAVIC3REIS, D. Water Supply Reservoir Operation by Combined Genetic Algorithm – Linear Programming (GA-LP). Water Resources Management (2006) 20: 227–255.

ROBIC, T., FILIPIC, B. DEMO: Differential Evolution for Multiobjective Optimization. In: Coello Coello, C.A., Hernandez Aguirre, A., Zitzler, E. (eds.) EMO 2005.LNCS, vol. 3410, pp. 520–533. Springer, Heidelberg. 2005

SABESP – Companhia de Saneamento Básico do Estado de São Paulo. Estudos De Cheias Para Avaliação Dos Volumes De Espera Dos Aproveitamentos Do Sistema Cantareira, Relatório Final, 2009.

_______, Reconstituição De Vazões Naturais Do Sistema Cantareira, Relatório Final, 2009.

SCHARDONG, A. et al. Comparação De Algoritmos Evolucionários Na Calibração De Modelo Chuva-Vazão - Smap. In: XVIII Simpósio Brasileiro de Recursos Hídricos, 2009, Campo Grande - MS. XVIII Simpósio Brasileiro de Recursos Hídricos. PORTO ALEGRE : ABRH, 2009.

SHAIK J. S. e YEASIN M. Visualization of High Dimensional Data using an Automated 3D Star Co-ordinate System. 2006 International Joint Conference on Neural Networks. Vancouver, BC, Canada. 2006

SCHAFFER, J. D.; Some Experiments in Machine Learning Using Vector Evaluated Genetic Algorithms. 1981. (Tese Doutorado), Vanderbilt University, Nashville, TN.

SIMONOVIC, S. Reservoir systems analysis: closing gap between theory and practice. Journal Water Resources Planning and Management, v. 118, n. 3, p. 262-280, 1992.

SIMONOVIC, S. P. Managing Water Resources - Methods and Tools for a Systems

Approach. London: UNESCO Publishing. 2009. 640 p.

SIMONOVIC, S. P. Systems Approach to Management of Disasters: Methods and Applications, New York: Wiley, 2010.

SRINIVAS, N. e DEB, K. Multi-Objective function optimization using non-dominated sorting genetic algorithms, Evolutionary Computation, 2(3):221–248. 1994

STRON, R. M. e PRICE, K. V. 1995 Differential Evolution – A simple and efficient adaptive scheme for global optimization over continuous spaces. Technical Report TR-95-012, ICSI.

SUEN, J.-P., EHEART, J.W., AND HERRICKS, E.E.. "Integrating Ecological Flow Regimes in Water Resources Management Using Multiobjective Analysis." Proc. ASCE/EWRI World Water and Envmtl. Resour. Cong., ASCE, Reston, VA., 2005

TANG, Y. e REED, P.M. Multiobjective Tools and Strategies for Calibrating Integrated Models. Proc. ASCE/EWRI World Water and Envmtl. Resour. Cong., ASCE, Reston, VA. 2005

USACE, 2001 Hydrologic Modeling System HEC-HMS. User's Manual, Version 2.1. US Army Corps of Engineers, Hydrologic Engineering Center, 178 p.

VASAN A. e SIMONOVIC S. P. (2008). Optimization Using Differential Evolution. Water Resources Research Report no. 060, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 38 pp.

VIENNET, C;. FONTIEX, e MARC, I.. New Multicriteria Optimization Method Based on the Use of a Diploid Genetic Algorithm: Example of an Industrial Problem. In J. M. Alliot, E. Lutton, E. Ronald, M. Schoenauer, and D. Snyers, editors, Proceedings of Artificial Evolution (European Conference), pages 120–127. Springer-Verlag, Brest, France, September. 1995.

WURBS, R. A. Reservoir-System Simulation and Optimization Models, Journal of

Water Resources Planning and Management, ASCE, v.119, n.4, p.455-472, 1993

YANDAMURI, S.R.M., SRINIVASAN, K., AND BHALLAMUDI, S.M. Multibojective Optimal Waste Load Allocation Models for Rivers Using Nondominated Sorting Genetic Algorithm-II. J. Water Resour. Plng. and Mgmt., ASCE, Reston, VA, 132 (3) 133-143. 2006

YEH, W.W-G. Reservoir management and operation models: a state-of-the-art review. Journal Water Resources Planning and Management, v. 21, n. 12, p. 1797-1818, 1985.

ZITZLER, E., LAUMANNS, M. e THIELE, L., SPEA2: Improving the Strength Pareto Evolutionary Algorithm, Technical Report TIK-Report 103, Swiss Federal Institute of Technology, 2001.

ZITZLER, E. e THIELE, L., “Multiobjective evolutionary algorithms: a comparative ase study and the strength Pareto approach,” IEEE Trans. Evol. Comput., vol. 3, p. 257-271, 1999.

7 ANEX I – Decision Support System for Multi-Objective Analysis

The Decision Support System described in this report is built for use on Windows XP or higher operation systems. It can be used for analysis of multi-objective problems and it’s free for education and personal use only.

The source code for the algorithms is provided upon request to the author (Andre Schardong at andreschardong@gmail.com). The user interface (presented in this report) installer can be downloaded from www.andreschardong.com/MODSS/MODSSInstall.exe. The installation processes is very simple and straightforward. After first installation the UI will automatically check for new updates.

8 List of Previous Reports in the Series

ISSN: (print) 1913-3200; (online) 1913-3219

(1) Slobodan P. Simonovic (2001). Assessment of the Impact of Climate Variability and Change on the Reliability, Resiliency and Vulnerability of Complex Flood Protection Systems. Water Resources Research Report no. 038, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 91 pages. ISBN: (print) 978-0-7714-2606-3; (online) 978-0-7714-2607-0.

(2) Predrag Prodanovic (2001). Fuzzy Set Ranking Methods and Multiple Expert Decision Making. Water Resources Research Report no. 039, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 68 pages. ISBN: (print) 978-0-7714-2608-7; (online) 978-0-7714-2609-4.

(3) Nirupama and Slobodan P. Simonovic (2002). Role of Remote Sensing in Disaster Management. Water Resources Research Report no. 040, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 107 pages. ISBN: (print) 978-0-7714-2610-0; (online) 978-0-7714- 2611-7.

(4) Taslima Akter and Slobodan P. Simonovic (2002). A General Overview of Multiobjective Multiple-Participant Decision Making for Flood Management. Water Resources Research Report no. 041, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 65 pages. ISBN: (print) 978-0-7714-2612-4; (online) 978-0-7714-2613-1.

(5) Nirupama and Slobodan P. Simonovic (2002). A Spatial Fuzzy Compromise Approach for Flood Disaster Management. Water Resources Research Report no. 042, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 138 pages. ISBN: (print) 978-0-7714-2614-8; (online) 978-0-7714-2615-5.

(6) K. D. W. Nandalal and Slobodan P. Simonovic (2002). State-of-the-Art Report on Systems Analysis Methods for Resolution of Conflicts in Water Resources Management. Water Resources Research Report no. 043, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 216 pages. ISBN: (print) 978-0-7714-2616-2; (online) 978-0-7714-2617-9.

(7) K. D. W. Nandalal and Slobodan P. Simonovic (2003). Conflict Resolution Support System – A Software for the Resolution of Conflicts in Water Resource Management. Water Resources Research Report no. 044, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 144 pages. ISBN: (print) 978-0-7714-2618-6; (online) 978-0-7714-2619-3.

(8) Ibrahim El-Baroudy and Slobodan P. Simonovic (2003). New Fuzzy Performance Indices for Reliability Analysis of Water Supply Systems. Water Resources Research Report no. 045, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 90 pages. ISBN: (print) 978-0-7714-2620-9; (online) 978-0-7714- 2621-6.

(9) Juraj Cunderlik (2003). Hydrologic Model Selection for the CFCAS Project: Assessment of Water Resources Risk and Vulnerability to Changing Climatic Conditions. Water Resources Research Report no. 046, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 40 pages. ISBN: (print) 978-0-7714- 2622-3; (online) 978-0-7714- 2623-0.

(10) Juraj Cunderlik and Slobodan P. Simonovic (2004). Selection of Calibration and Verification Data for the HEC-HMS Hydrologic Model. Water Resources Research Report no. 047, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 29 pages. ISBN: (print) 978-0-7714-2624-7; (online) 978-0-7714-2625-4.

(11) Juraj Cunderlik and Slobodan P. Simonovic (2004). Calibration, Verification and Sensitivity Analysis of the HEC-HMS Hydrologic Model. Water Resources Research Report no. 048, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 113 pages. ISBN: (print) 978- 0-7714-2626-1; (online) 978-0-7714- 2627-8.

(12) Predrag Prodanovic and Slobodan P. Simonovic (2004). Generation of Synthetic Design Storms for the Upper Thames River basin. Water Resources Research Report no. 049, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 20 pages. ISBN: (print) 978- 0-7714-2628-5; (online) 978-0-7714-2629-2.

(13) Ibrahim El-Baroudy and Slobodan P. Simonovic (2005). Application of the Fuzzy

Performance Indices to the City of London Water Supply System. Water Resources Research Report no. 050, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 137 pages. ISBN: (print) 978-0-7714-2630-8; (online) 978-0-7714-2631-5.

(14) Ibrahim El-Baroudy and Slobodan P. Simonovic (2006). A Decision Support System for Integrated Risk Management. Water Resources Research Report no. 051, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 146 pages. ISBN: (print) 978-0-7714-2632-2; (online) 978-0-7714-2633-9.

(15) Predrag Prodanovic and Slobodan P. Simonovic (2006). Inverse Flood Risk Modelling of The Upper Thames River Basin. Water Resources Research Report no. 052, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 163 pages. ISBN: (print) 978-0-7714-2634-6; (online) 978-0-7714-2635-3.

(16) Predrag Prodanovic and Slobodan P. Simonovic (2006). Inverse Drought Risk Modelling of The Upper Thames River Basin. Water Resources Research Report no. 053, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 252 pages. ISBN: (print) 978-0-7714-2636-0; (online) 978-0-7714-2637-7.

(17) Predrag Prodanovic and Slobodan P. Simonovic (2007). Dynamic Feedback Coupling of Continuous Hydrologic and Socio-Economic Model Components of the Upper Thames River Basin. Water Resources Research Report no. 054, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 437 pages. ISBN: (print) 978-0-7714-2638-4; (online) 978-0-7714-2639-1.

(18) Subhankar Karmakar and Slobodan P. Simonovic (2007). Flood Frequency Analysis Using Copula with Mixed Marginal Distributions. Water Resources Research Report no. 055, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 144 pages. ISBN: (print) 978-0-7714-2658-2; (online) 978-0-7714-2659-9.

(19) Jordan Black, Subhankar Karmakar and Slobodan P. Simonovic (2007). A Web-Based Flood Information System. Water Resources Research Report no. 056, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 133 pages. ISBN: (print) 978-0-7714-2660-5; (online) 978-0-7714-2661-2.

(20) Angela Peck, Subhankar Karmakar and Slobodan P. Simonovic (2007). Physical, Economical, Infrastructural and Social Flood Risk – Vulnerability Analyses in GIS. Water Resources Research Report no. 057, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 80 pages. ISBN: (print) 978-0- 7714-2662-9; (online) 978-0-7714-2663-6.

(21) Predrag Prodanovic and Slobodan P. Simonovic (2007). Development of Rainfall Intensity Duration Frequency Curves for the City of London Under the Changing Climate. Water Resources Research Report no. 058, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 51 pages. ISBN: (print) 978-0- 7714-2667-4; (online) 978-0-7714-2668-1.

(22) Evan G. R. Davies and Slobodan P. Simonovic (2008). An integrated system dynamics model for analyzing behaviour of the social-economic-climatic system: Model description and model use guide. Water Resources Research Report no. 059, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 233 pages. ISBN: (print) 978-0-7714-2679-7; (online) 978-0-7714-2680-3.

(23) Vasan Arunachalam (2008). Optimization Using Differential Evolution. Water Resources Research Report no. 060, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 42 pages. ISBN: (print) 978-0-7714- 2689-6; (online) 978-0-7714-2690-2.

(24) Rajesh Shrestha and Slobodan P. Simonovic (2009). A Fuzzy Set Theory Based Methodology for Analysis of Uncertainties in Stage-Discharge Measurements and Rating Curve. Water Resources Research Report no. 061, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 104 pages. ISBN: (print) 978-0-7714-2707-7; (online) 978-0-7714-2708-4.

(25) Hyung-Il Eum, Vasan Arunachalam and Slobodan P. Simonovic (2009). Integrated Reservoir Management System for Adaptation to Climate Change Impacts in the Upper Thames River Basin. Water Resources Research Report no. 062, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 81 pages. ISBN: (print) 978-0-7714-2710-7; (online) 978-0-7714-2711-4.

(26) Evan G. R. Davies and Slobodan P. Simonovic (2009). Energy Sector for the Integrated

System Dynamics Model for Analyzing Behaviour of the Social- Economic-Climatic Model. Water Resources Research Report no. 063. Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada. 191 pages. ISBN: (print) 978-0-7714-2712-1; (online) 978-0-7714-2713-8.

(27) Leanna King, Tarana Solaiman, and Slobodan P. Simonovic (2009). Assessment of Climatic Vulnerability in the Upper Thames River Basin. Water Resources Research Report no. 064, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 61pages. ISBN: (print) 978-0-7714-2816-6; (online) 978-0-7714- 2817-3.

(28) Slobodan P. Simonovic and Angela Peck (2009). Updated Rainfall Intensity Duration Frequency Curves for the City of London under Changing Climate. Water Resources Research Report no. 065, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 64pages. ISBN: (print) 978-0-7714-2819-7; (online) 987-0-7714-2820-3.

(29) Leanna King, Tarana Solaiman, and Slobodan P. Simonovic (2010). Assessment of Climatic Vulnerability in the Upper Thames River Basin: Part 2. Water Resources Research Report no. 066, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 72pages. ISBN: (print) 978-0-7714-2834-0; (online) 978-0-7714-2835-7.

(30) Christopher J. Popovich, Slobodan P. Simonovic and Gordon A. McBean (2010). Use of an Integrated System Dynamics Model for Analyzing Behaviour of the Social-Economic-

Climatic System in Policy Development. Water Resources Research Report no. 067, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 37 pages. ISBN: (print) 978-0-7714-2838-8; (online) 978-0-7714-2839-5.

(31) Hyung-Il Eum and Slobodan P. Simonovic (2009). City of London: Vulnerability of Infrastructure to Climate Change; Background Report 1 – Climate and Hydrologic Modeling. Water Resources Research Report no. 068, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 102pages. ISBN: (print) 978-0-7714-2844-9; (online) 978-0-7714-2845-6.

(32) Dragan Sredojevic and Slobodan P. Simonovic (2009). City of London: Vulnerability of Infrastructure to Climate Change; Background Report 2 – Hydraulic Modeling and Floodplain Mapping. Water Resources Research Report no. 069, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 147 pages. ISBN: (print) 978-0-7714-2846-3; (online) 978-0-7714-2847-0.

(33) Tarana A. Solaiman and Slobodan P. Simonovic (2011). Quantifying Uncertainties in the Modelled Estimates of Extreme Precipitation Events at the Upper Thames River Basin. Water Resources Research Report no. 070, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 167 pages. ISBN: (print) 978-0-7714-2878-4; (online) 978-0-7714-2880-7.

(34) Tarana A. Solaiman and Slobodan P. Simonovic (2011). Assessment of Global and Regional Reanalyses Data for Hydro-Climatic Impact Studies in the Upper Thames River

Basin. Water Resources Research Report no. 071, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 74 pages. ISBN: (print) 978-0-7714-2892-0; (online) 978-0-7714-2899-9.

(35) Tarana A. Solaiman and Slobodan P. Simonovic (2011). Development of Probability Based Intensity-Duration-Frequency Curves under Climate Change. Water Resources Research Report no. 072, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 89 pages. ISBN: (print) 978-0-7714-2893-7; (online) 978-0-7714-2900-2.

(36) Dejan Vucetic and Slobodan P. Simonovic (2011). Water Resources Decision Making Under Uncertainty. Water Resources Research Report no. 073, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 143 pages. ISBN: (print) 978-0-7714-2894-4; (online) 978-0-7714-2901-9.

(37) Angela Peck, Elisabeth Bowering and Slobodan P. Simonovic (2010). City of London: Vulnerability of Infrastructure to Climate Change, Final Report - Risk Assessment. Water Resources Research Report no. 074, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 66 pages. ISBN: (print) 978-0-7714-2895-1; (online) 978-0-7714-2902-6.

(38) Akhtar, M. K., S. P. Simonovic, J. Wibe, J. MacGee, and J. Davies, (2011). An integrated system dynamics model for analyzing behaviour of the social-energy-economic-climatic system: model description. Water Resources Research Report no. 075, Facility for

Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 211 pages. ISBN: (print) 978-0-7714-2896-8; (online) 978-0-7714-2903-3.

(39) Akhtar, M. K., S. P. Simonovic, J. Wibe, J. MacGee, and J. Davies, (2011). An integrated system dynamics model for analyzing behaviour of the social-energy-economic-climatic system: user's manual. Water Resources Research Report no. 076, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 161 pages. ISBN: (print) 978-0-7714-2897-5; (online) 978-0-7714-2904-0.

(40) Millington, N., S. Das and S. P. Simonovic, (2011). The Comparison of GEV, Log-Pearson Type 3 and Gumbel Distributions in the Upper Thames River Watershed under Global Climate Models. Water Resources Research Report no. 077, Facility for Intelligent Decision Support, Department of Civil and Environmental Engineering, London, Ontario, Canada, 53 pages. ISBN: (print) 978-0-7714-2898-2; (online) 978-0-7714-2905-7.