SMART

Work package 6, Deliverable D603

Multi-Criteria Optimization (MCO) Module
Version 1.2

J. Bensabat
S. Dror

February, 2013

Corresponding author:

Jacob Bensabat
EWRE Environmental & Water Resources Engineering Ltd.
PO Box 6770 Haifa 31067, Israel. jbensabat@ewre.com , +972-4-838-3919.

Funded by the German Federal Ministry of Education and Research (BMBF):

FKZ 02WM0801

Project Coordination:

Main Coordinator:

Prof. Dr. Nico Goldscheider
Department of Applied Geology
Karlsruhe University
Adenauerring 20b
76131 Karlsruhe
Germany
Phone: +49 (0) 721 608 3096
Fax: +49 (0) 721 606 279

Assistant Coordinators

Prof. Dr. Martin Sauter
Department of Applied Geology
Göttingen University
Goldschmidtstrasse 3
37077 Göttingen
Germany
Phone: +49 (0)551 39 79 11
Fax: +49 (0)551 39 93 79

Dr. Stefan Geyer & Dr. Roland Mueller
Helmholtz Centre for Environmental Research (UFZ)
Permoserstr. 15
04318 Leipzig
Germany
Phone: +49 (0)341 235 30 00
Fax: +49 (0)341 235 2885

http://www.iwrm-smart.org/

TABLE OF CONTENTS

1. Introduction

The Multi-Criteria Optimization (MCO) module provides the ability to formulate, solve and analyze multi-objective optimization problems, either linear or non-linear (both convex and non-convex). While it can be defined as a decision making tool, it is fundamentally different from the two multi-criteria decision making tools that have been deployed in the frame of the SMART-DSS tools platform. MCO is of mathematical nature and its supplements in functionality the MCDM (multi-criteria decision making tools) AHP and ELECTRE-III, which are already implemented and heavily rely on subjective judgment and preference. So the question arising is what is the need for such a tool? How and where can it be used and for what purpose?

The MCDM and MCO tools complement each other and allow a subtle balancing of subjective and objective judgment that could be used for the decision making, as this will be demonstrated in the Scenario Generator tool (to be presented in deliverable D608).

First let's formulate the general form of a MCO problem (which can be found in any textbook or paper dealing with similar subjects):

$$\begin{aligned} &\min_x \mu(x) \\ &s.t. \qquad g(x) \le 0 \\ &\qquad\qquad h(x) = 0 \\ &\qquad\qquad x_l \le x \le x_u \end{aligned}$$

Here μ denotes the vector of the objective functions $\mu = (\mu_1, \mu_2, \ldots\ldots, \mu_m)$ where m denotes the number of objectives functions and $x = (x_1, x_2, \ldots\ldots x_n)$ denotes vector decision variables (of size n). Examples of decision variables could include: the amount of fresh water, brackish water or treated effluent that could be allocated for irrigation, the size arable land for specific crops, the size of greenhouses, the amount of freshwater to be allocated for industrial activity etc. The objective function vector μ are mathematical functions expressing objectives (such as maximum profit, minimum investment, minimum impact on groundwater resources, minimum adverse socio-economic impact etc.). The objectives functions must be independent, often conflicting, in order for the problem to be truly multi-objective. The vectors $g(x)$ and $h(x)$ represent inequality and equality constraints. Finally, x_l and x_u denote the vectors of lower and upper bounds on the decision variables vector x.

2. The mathematical concept

For nontrivial multi-objective problems, it is not possible to identify one unique solution, a vector x that simultaneously optimizes all the objectives. While searching for solutions, one reaches points such that, when attempting to improve an objective further, other objectives contemplate a deterioration. These are called non dominated solutions or Pareto solutions. The goal of this module is to suggest tools for the identifying the set of non-dominated solutions (the Pareto set or the Pareto surface) for a given problem, defined by its decision variables, constraints and objectives. As we shall notice, this approach is far less intuitive than the decision making tools AHP and ELECTRE, which are available in the SMARTDSS platform.

A mathematical description of a non-dominated solution is given by the following definition below.

A solution $x^{(1)}$ is said to dominate a solution $x^{(2)}$ if the following conditions hold:

$$\mu_i\left(x^{(1)}\right) \leq \mu_i\left(x^{(2)}\right) \quad \text{forall}\, i \in \{1,2,3,4,.,.,.,.,.m\}$$

And

$$\mu_j\left(x^{(1)}\right) \leq \mu_j\left(x^{(2)}\right) \quad \text{for at least one} \quad j \in \{1,2,3,4,.,.,.,.,.m\}$$

In simple words, a non-dominated solution (or a Pareto optimal) solution is achieved, when we reach the situation in which the improvement of one objective function results automatically in the deterioration of one or more objective functions. Finding such non-dominated solutions, and quantifying the trade-offs in satisfying the different objectives, is the goal when setting up and solving a multi-objective optimization problem.

Therefore, the set of Pareto solutions is a curve in case of two objective functions and surface in case of three objective functions or a hyper-surface in spaces of dimension greater than 3. For larger number of objective functions, it is not possible to directly visualize the Pareto set and additional options have been suggested (such as Lotov, 2004).

There are many techniques for calculating the Pareto set and in this deliverable we shall focus on three of them with proven application in engineering: 1) the weighted sum (optimizing a linear combination of the objective functions); 2) the Normal Boundary Intersection (NBI) (Das I., 1997) and; 3) The normalized normal constraints (NNC) method (Messac A., 2004). All the algorithms have been implemented in the GPL ACADO software (www.acado.org).

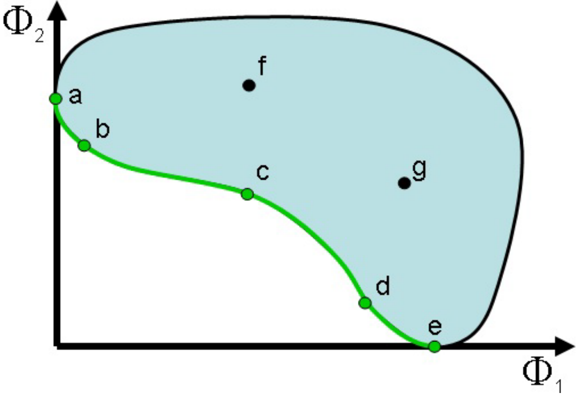
Some concepts of Multi-Objective Optimization and scalarization methods are briefly introduced.

Graphical representation of the Pareto optimality concept

Figure 1 (can be found in numerous references) illustrates the Pareto concept for a two-objective optimization problem. The feasible objective space is depicted in blue and the Pareto set is displayed in green. Hence, all points *a* to *e* are Pareto optimal, while *f* and *g* are not. In this figure, it can be seen that f (g) could be improved without deteriorating g (f), while it is impossible improve either one of the a, b, c, d, and e solutions without a subsequent deterioration of one or more these solutions. These approaches correspond to a scalarization technique. The goal of the methods to be presented below is to construct the Pareto surface or a set of points that characterize this surface.

In what follows, we shall assume that the problems under scrutiny are static, not time dependent. It is possible to define likewise MCO problems having a dynamic structure. However, we claim that most of the IWRM problems have a time response that is relatively slow, so there is less interest in investigating their dynamic structure. Examples of dynamic MCO problems would include the optimization of the trajectory of a high speed flying object (plane, missile etc.) or the control of a chemical reaction. These, optimal control problems, would need a dynamic component.

Figure 1: Pareto concept for a bi-objective optimization problem



The rationale behind this class of solution methods is to convert the original multi-objective optimization problem into a series of parametric single objective optimization problems. By consistently varying the method's parameters an approximation of the Pareto front is obtained. Despite several intrinsic drawbacks (particularly for non-convex problems), the convex Weighted Sum (WS) is still the most popular scalarization method. Alternatively, novel approaches that mitigate the drawbacks of the WS have been reported: **Normal Boundary Intersection** (NBI) and **Normalized Normal Constraint** (NNC).

The resulting Pareto set can be visualized, though visualization is limited to cases with up to three objectives.

There is an extensive documentation on Multi-Objective optimization (dedicated journals, books and numerous papers), as this is a very active field of research and practical implementation, in engineering (chemical, civil, aeronautical, electrical, water resources) and in general in any field in which decisions have to made in the presences of multiple and conflicting objectives. This technique is particularly suited to IWRM, as by its inherent definition, IWRM attempts to reconcile objectives of different natures in the same framework of water resources management. Additional examples would include aquifer management in presence of water bodies of different qualities, such as saline and freshwater bodies. In this case objectives would include maximization of freshwater abstraction while minimizing the salinization process. These two objectives are by definition contradictory (Gordon et. Al, 2000 and 2001).

3. References

1. Das I. and J. Dennis, 1996. Normal boundary intersection: *An alternate method for generating Pareto optimal points in Multi-criteria optimization problems*. ICASE report # 96-62, NASA, Institute for computer applications in science and engineering.

2. Das I. and J. Dennis, 1998. *Normal-Boundary Intersection: A New Method for Generating the Pareto Surface in Nonlinear Multicriteria Optimization Problems*. SIAM Journal on Optimization (8-3), 631-657.

3. Gordon K., Shamir U. and J. Bensabat, 2000. *Optimal management of a regional aquifer under salinization conditions*. Water Resources Research, (36), 3193-3203.

4. Gordon K., Shamir U. and J. Bensabat, 2001. *Optimal extraction of water from regional aquifer under salinization*. Journal of water resources planning and management – ASCE, (2-127), 71-77.

5. Lotov A.V., Bushenkov V.A and G.K. Kamenev, 2004. *Interactive Decision Maps, Approximation and visualization of Pareto frontier*. Kluwer Academic Publishers.

6. Messac A., Ismail-Yahaia A. and C.A. Mattson, 2003. *The Normalized Normal Constraint Method for Generating the Pareto Frontier*. Structural and Multidisciplinary optimization, (25) 86-98.

7. Sanchis J., Martinez M. and X. Blasco, 2008. *A new perspective on multiobjective optimization by enhanced normalized normal constraint method*. Structural and Multidisciplinary optimization, (36-5) 537-546.

4. MCO User's Guide

The MCO module is integrated into the SMART-DSS general graphical user interface (GUI), which, as of now can be run from EWRE web site (www.ewre.com/smartdss/publish.htm) and can be loaded from its main menu bar. It comprises the following functionality:

1. An interface for defining the MCO problem in three main sections, i.e., the decision variables and constraints on their values, the general constraints and the objective functions.
2. An interface for defining parameters to be used in the definition of the constraints and the objective functions.
3. An interface for the definition of aggregated variables for processing of the results.
4. Implementation of four MCO algorithms (from the ACADO library).
5. Visualization of the results.
6. Export to various formats (XML, CSV) and for further processing, such as the scenario generator.

The input of the constraints and the objective functions is enabled by expression parser. The user defines the constraints and objective functions as symbolic expressions using a symbolic (such as matlab and or octave) type of syntax. Once input, MCO will check for syntax errors and if the parsing succeeds, it allows moving to the calculation area.

In order to transform the Multi-Objective Optimal Control Problems concept into an applicable decision making tool, EWRE developed an application, implementing the optimization and scalarization methods.

Similar to the AHP, Electre and HydroBudget applications, the MCO application is Dialog based, that is installed from a cd, and runs on window systems.

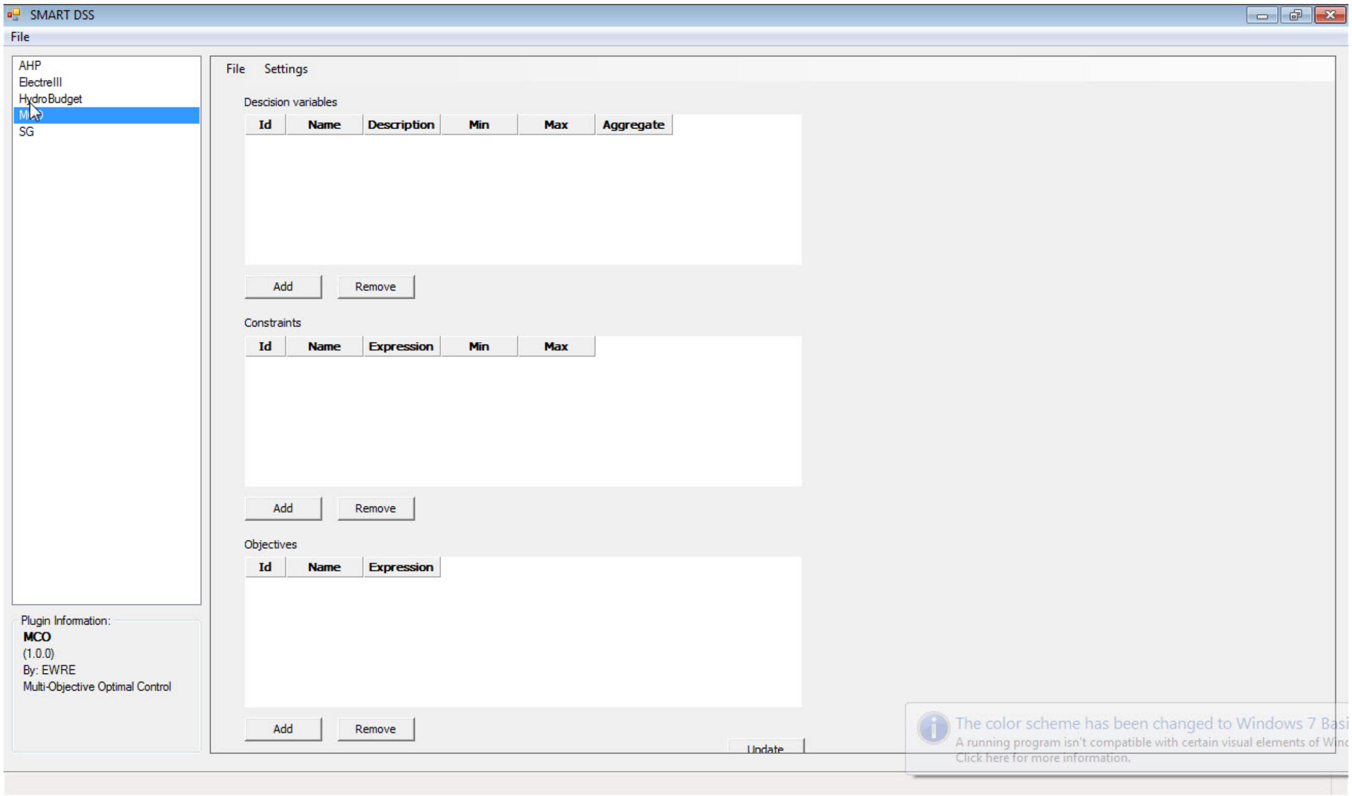
The MCO application is divided into three different phases:

1. Basic definitions (Parameters and Aggregates);
2. Definition of the problem (input of the decision variables, constraints and objective functions);
3. Calculation of the Pareto surface.

The first step deals with the definition of the decision variables, Constraints and the Objectives. In the second phase, Visualization of the MCO Pareto solution is presented. In the two first phases, the involvement of the decision makers (technical, political) is required.

The user can download an existing project or create a new one. A MCO project is saved as an xml file.

4.1 The MCO main window



Plugin Information:
MCO
(1.0.0)
By: EWRE
Multi-Objective Optimal Control

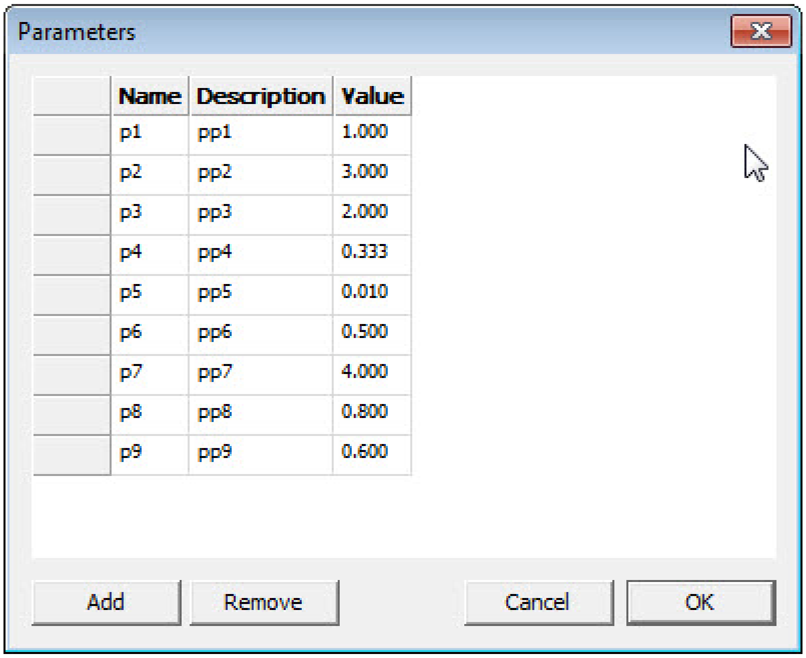
The color scheme has been changed to Windows 7 Basi
A running program isn't compatible with certain visual elements of Wind
Click here for more information.

Upon opening the SMART-DSS application, we obtain the general structure of the software. By choosing the MCO module we switch to the MCO graphical user interface (GUI). We have two menu items.

Menu Item	**SubMenu Item**	**Functionality**
File	Open	Open a MCO workspace. This is a file that stores all the definitions of a MCO problem, for further use or modification, wihtout the need for definitions from scracth. The format of this file is XML, which is a versatile format for the storage and transfer of information.
	Save	Save a MCO workspace in the XML format under the current file name (default).
	Save As	Save a MCO workspace in the XML format under another name (XML format).
Settings	Parameters	Load the dialog for the definition of parameters (see below)
	Aggregation	Load the dialog for the definition aggregated variables (see below).

In the right pane of the MCO GUI, the screeen is divided in three separate areas: 1) definition of decision variables; 2) definition of constraints; and 3) definition of objective functions.

4.2 The Parameters Dialog



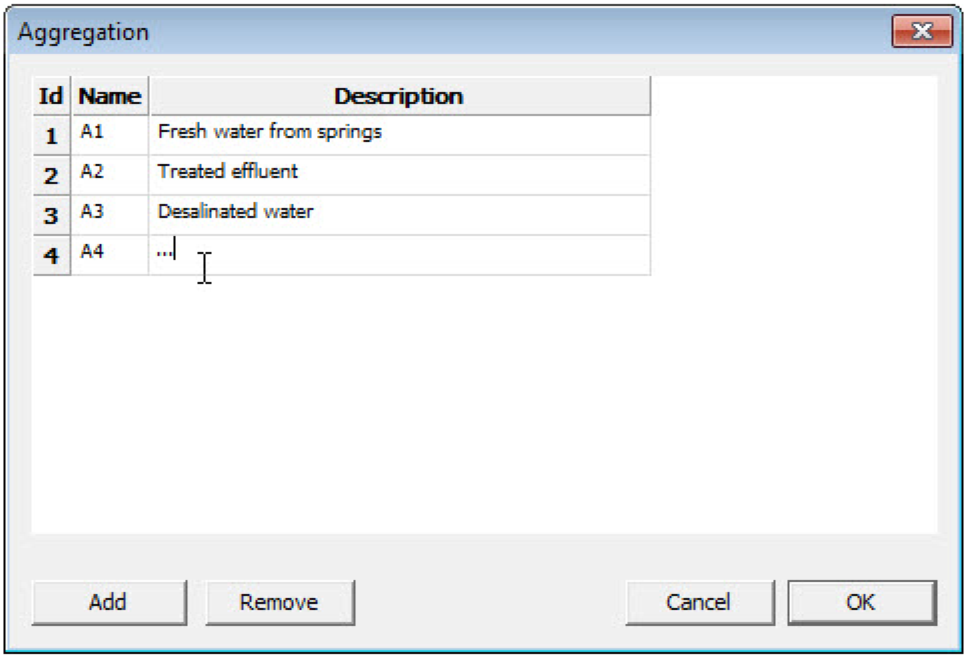
Name	Description	Value
p1	pp1	1.000
p2	pp2	3.000
p3	pp3	2.000
p4	pp4	0.333
p5	pp5	0.010
p6	pp6	0.500
p7	pp7	4.000
p8	pp8	0.800
p9	pp9	0.600

The Parameters dialog can be loaded from the MCO main window (settings->Parameters). This dialog provide the ability to define parameters to be used in the definition of the MCO constraints and or objective functions. This allows fast update of the MCO problem for the purpose of sensitivity analysis and it allows a more consistent and bug free definition of problems with relatively large size.

One record in the Parameters dialog requires three input fields (two of them mandatory):

Field	Type	Purpose
Name	Mandatory	Provide a name to parameter for further use in the MCO problem.
Description	Optional	Provide a description of the parameter.
Value	Mandatory	Provide a numerical value for the parameter.

4.3 The Aggregation Dialog



Id	Name	Description
1	A1	Fresh water from springs
2	A2	Treated effluent
3	A3	Desalinated water
4	A4	...

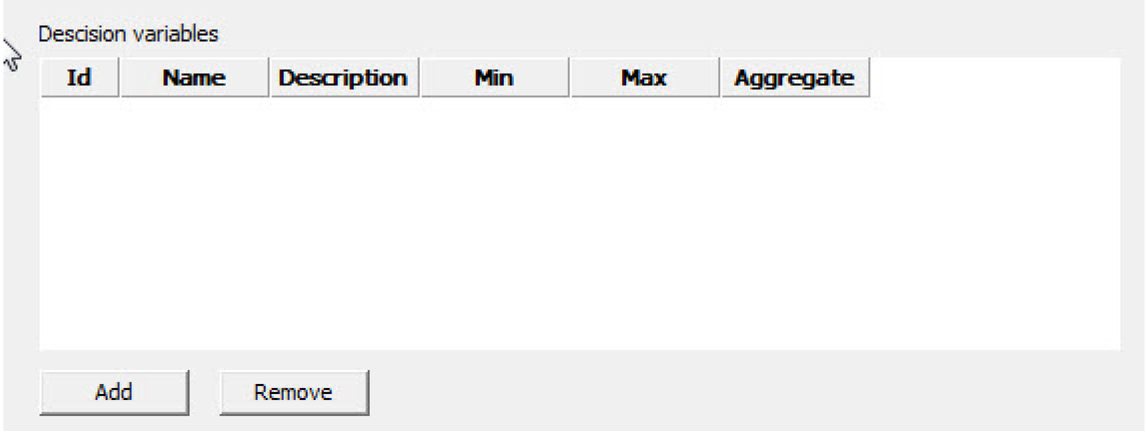
This dialog allows to define aggregation patterns for the decision variables. This functionality is useful when the MCO problem comprise a relatively large number of variables, which can be aggregated (summed) for the purpose of reporting.

The Aggregation dialog can be loaded from the MCO main window (settings->Aggregation). This dialog provide the ability to aggregation classes for the post-processing of MCO results. This particularly useful, when the MCO problem comprises a large number of decision variables, which makes the understanding and post-processing of the MCO results more cumbersome. Therefore the Aggregation functionality applies to the decision variables only.

One record in the Parameters dialog requires three input fields (two of them mandatory):

Field	Type	Purpose
ID	Mandatory	Provide a unique identity (number) to the class of aggregated variables.
Name	Mandatory	Provide a name of the aggregation class.
Value	optional	Provide a description of the aggregation class.

4.4 Defintion of decision variables



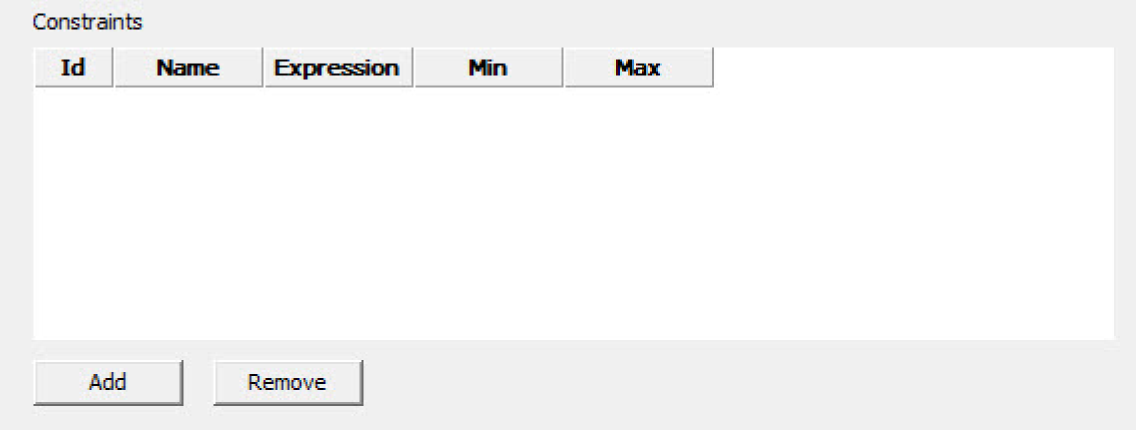
Descision variables

Id	Name	Description	Min	Max	Aggregate

In the upper part of the MCO input area, we have a form for the definition of decision variables. The definition of a decision variable requires six fields as follows:

Field	Description	Type	Purpose
ID	Variable id (unique) supplied by the software	Non editable	This is the unique id of the variable
Name	Name of the variable	Mandatory	Provide a hint on the nature of the variable
Description	Describe the nature of the variable	Optional	Provide full description of the variable to be defined.
Min	Lower bound of the variable (the minimum possible value)	Optional	Set a lower limit to the value of a decision variable.
Max	Upper bound of the variable (the maximum possible value)	Optional	Set an upper limit to the value of a decision variable.
Aggregate	Set the aggregation class to which the variable belongs	Optional	Set the aggregation class to which this variable belongs (classes must be previously defined). For post processing of MCO results.

4.5 Definition of constraints



This is the second step of the MCO problem definition. A constraint is a mathematical description of the behavior (values) that the decision variable must meet. A constraint is therefore a mathematical expression of the decision variables. The input of a constraint includes five fields as follows:

ID	Name	Expression	Min	Max
Id set by MCO	Name of the constraint to be set by the user	Mathematical expression of the constraint	Lower bound of the constraint	Upper bound of the constraint

In order to input the mathematical expression of the constraint in a way that is the most convenient to the user, we have chosen to use the expression and extend the expression syntax developed in ACADO. In this software, MCO problems are hard-coded, i.e. the user must write a c++ or (matlab) program to be linked to the ACADO library. Mathematical expressions are then translated into c++ statements by ACADO. In this application we decided to waive the need for program writing, as most of the potential users do not have the required skills. Therefore we developed an additional layer of expression interpretation, from the GUI directly to the c++ code. There are syntax rules for the definition of expressions as follows:

1. The user can build an expression only with predefined decision variables and optionally with predefined parameters;
2. The following operators can be used:

Operator	Description
+	Addition
-	Subtraction
*	Multiplication
/	Division
^	Power

For Example the mathematical expression $3x_1^2 - {4x_1^4}\big/{\left(3x_3^2 - 17x_5^4\right)}$ would be translated to:

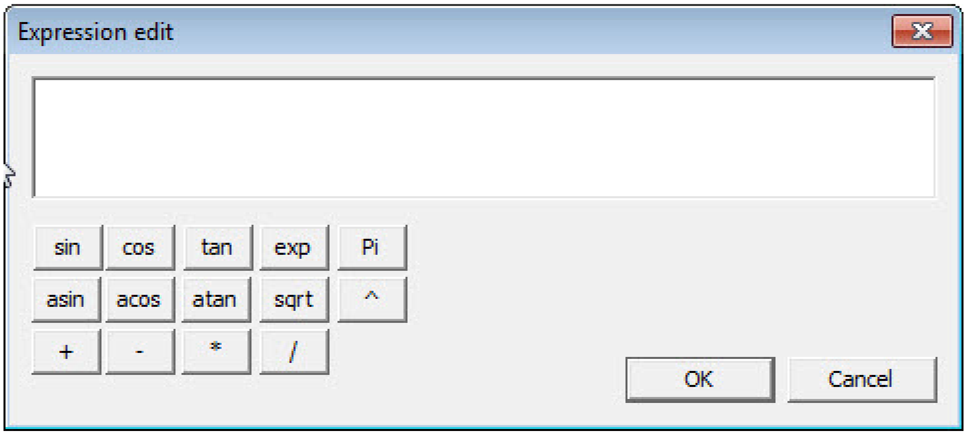
3* X1^2 – 4* X1^4 /(3* X3^2 – 17* X5^4)

Remark: strict constraints are not allowed, i.e., constraints having either "<" or ">" conditions. In order to load the expression editor, the user has to click the right button on the mouse on the expression definition cell (in selected mode). The expression editor allows easing the process of input of complex mathematical expressions.

Accepted functions include:

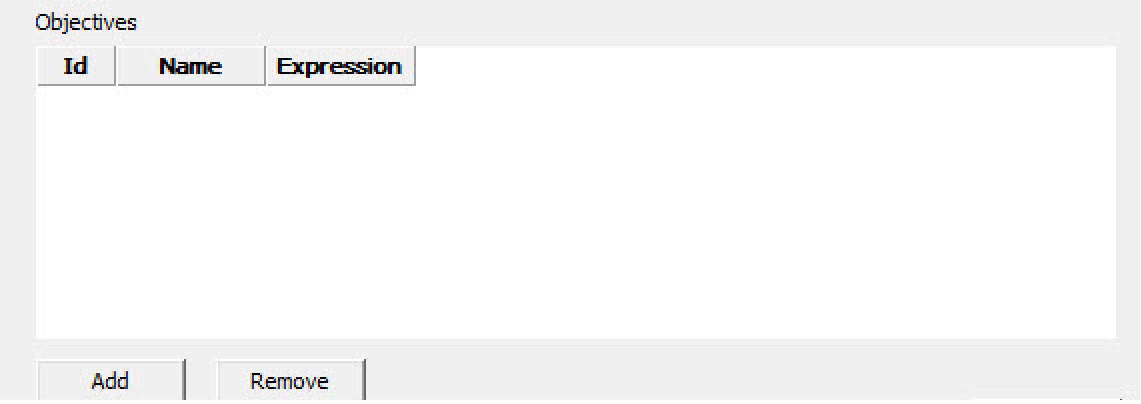
Function	Expression
Sinus	Sin
Cosinus	Cos
Tangent	Tan
Arc-sinus	Asin
Arc-Cosinus	Acos
Arc-tangent	Atan
Exponential	Exp
Square root	Sqrt
Natural logarithm	Ln
Power	^

A number of mathematical functions can be input and these are listed in the frugal expression editor provided by MCO. The figure below illustrates the GUI of the expression editor.



4.6 Definition of objective functions

Objective functions are defined as mathematical expressions in way that is identical to the definition of the constraints.

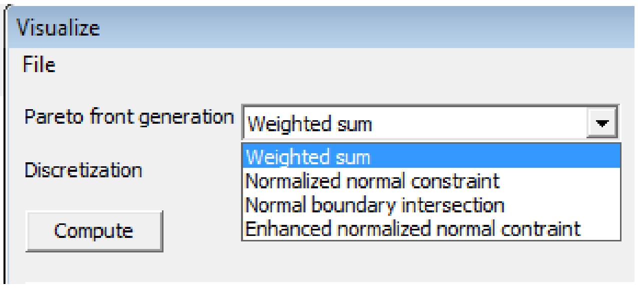


The definition of an objective function requires the input of three fields per objective function.

ID	Description	Functionality
Id of the objective functions Defined by the software	This is the unique id of the objective function, non-editable.	None
Name	Name of the objective function – optional, for the user purpose	None
Expression	Mathematical expression of the objective function	This mathematical expression is created in a way that is similar to the constraints and using the same tools described above (including the limitations).

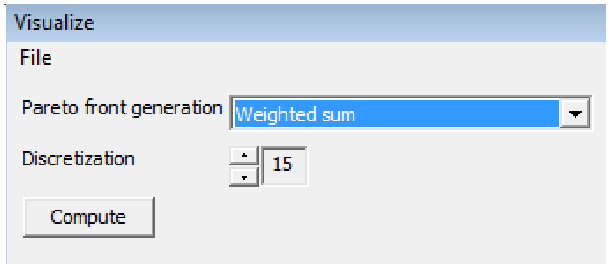
4.7 Selection of the calculation method

After clicking the “update” button, the “visualization” window appears. The user is asked to choose a scalarization method and define the size of the discretization grid. The last step in this phase is to click on the “Compute” button in order to visualize the Pareto calculation


Visualize
File
Weighted sum
Normalized normal constraint
Normal boundary intersection
Enhanced normalized normal contraint
Compute

Pareto front generation: Weighted sum

Discretization



Pareto front generation: Weighted sum

Discretization: 15

We have implemented four techniques”

1. Weighted sum: the most popular and simple, but definitely problematic technique, particularly when the defined problem is not convex;
2. Normalized normal constraint (Messac et al., 2003);
3. Normal boundary intersection (Das et al., 1998);
4. Enhance normalized normal constraint (Sanchis et al., 2008).

There is no real recommendation with regard to the most suitable technique. However, from our experience, we suggest to try all the techniques. Visually one will notice that some techniques are more suitable than others for the specific problem under scrutiny.

The size of the discretization grid depends of the quality of the shape that is required for the visualization or on the needs for the further use (such as the Scenario generator application). Therefore, it is recommended to proceed by trying all the techniques, taking into consideration that some may not work for a specific problem and then varying thr grid size until a suitable (to the problem on hand) representation of the Pareto surface is achieved.

5. Demonstration runs

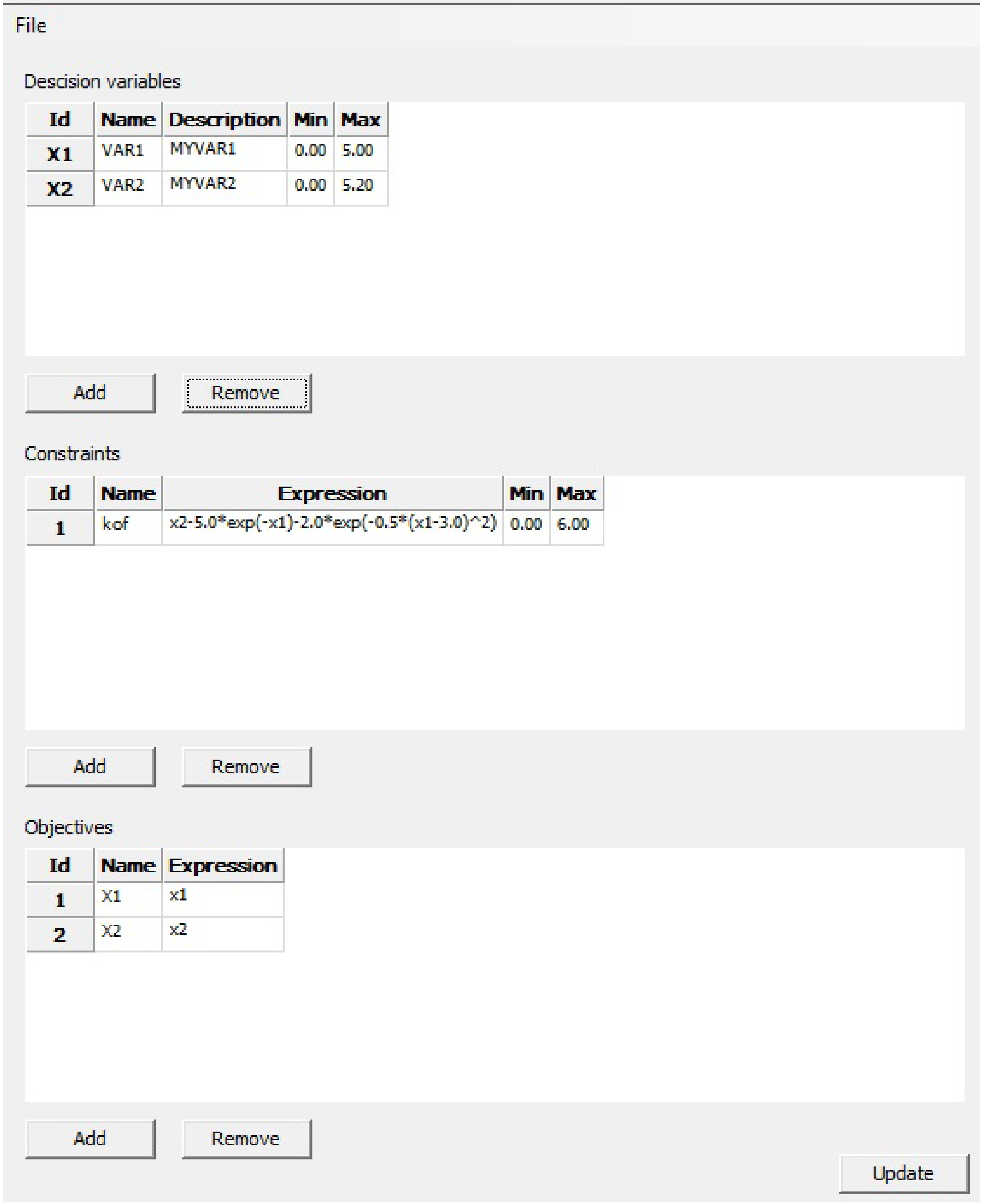
This chapter illustrates how to set up the bi-objective optimization problem and the tri-objective problem. These examples have no real link to the IWRM reality but they used for the validation of the function of the code and are extensively used for this purpose in MCO publications.

5.1 Optimization problem with two objectives

In this case we have two variables, x_1 and x_2. The MCO problem to be solved is:

$$
\begin{aligned}
&\min\left(x_1, x_2\right)\\
&s.t.\\
&\qquad 0 \le x_1 \le 5.0\\
&\qquad 0 \le x_2 \le 5.2\\
&\qquad 0 \le x_2 - 5\mathrm{e}^{(-x_1)} + 2\mathrm{e}^{\left(-0.5(x_1-3)^2\right)} \le 6
\end{aligned}
$$

While this problem has only two variables and two objective functions, it is a rather complex one, by being non-linear and non-convex. The lack of convexity results in a Pareto surface that has an irregular shape (which requires careful selection of the discretization grid). This problem was then translated to the MCO GUI (see figure below).



Descision variables

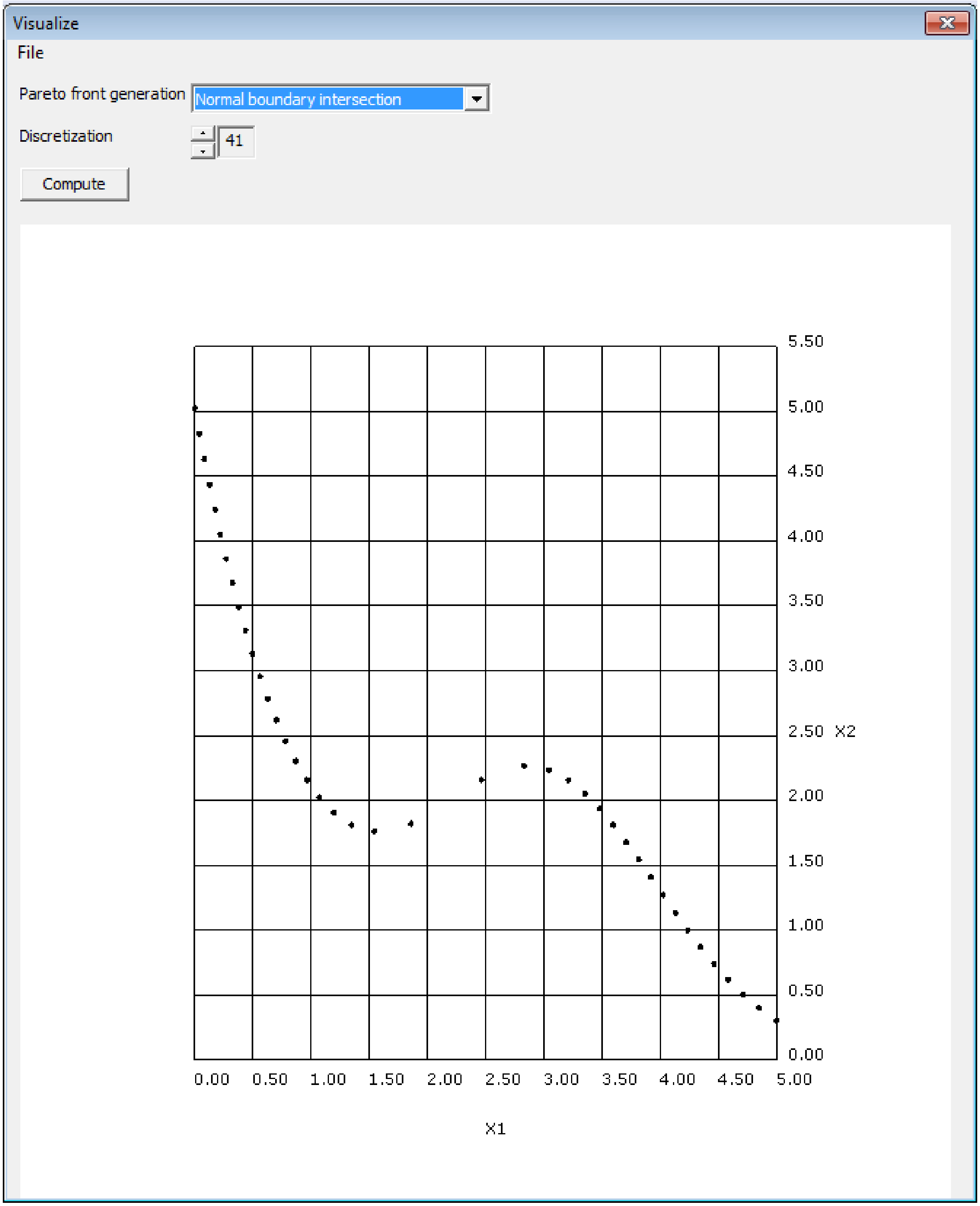
Id	Name	Description	Min	Max
X1	VAR1	MYVAR1	0.00	5.00
X2	VAR2	MYVAR2	0.00	5.20

Constraints

Id	Name	Expression	Min	Max
1	kof	x2-5.0*exp(-x1)-2.0*exp(-0.5*(x1-3.0)^2)	0.00	6.00

Objectives

Id	Name	Expression
1	X1	x1
2	X2	x2



Visualize

File

Pareto front generation: Normal boundary intersection

Discretization: 41

Compute

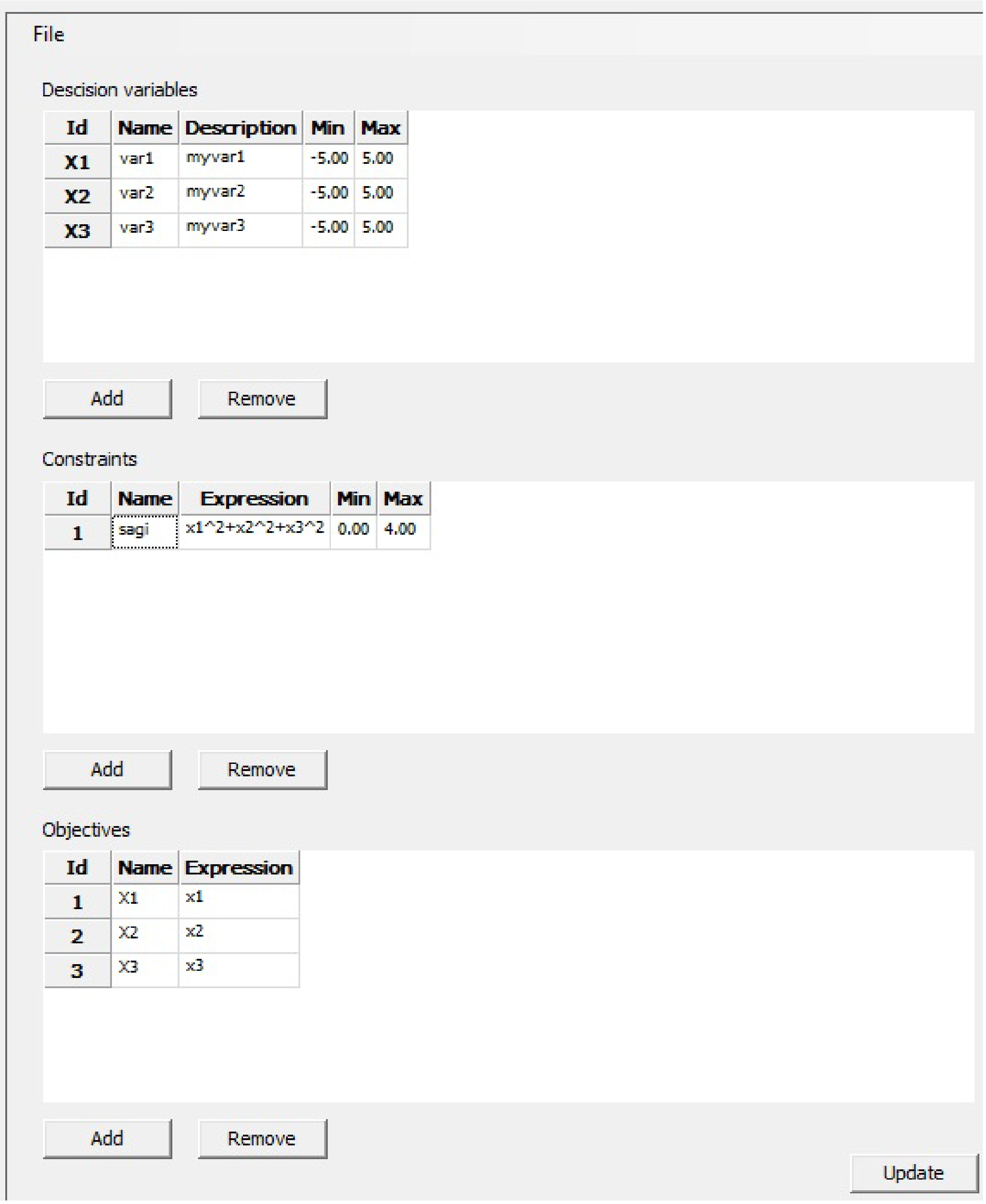
5.2 Optimization problem with three objectives

For the tri-objective problem only three scalar variables are involved: x_1, x_2 and x_3.

The aim is to simultaneously minimize these three variables. However, all are bounded and have to satisfy a nonlinear constraint:

$$
\begin{aligned}
&\min\left(x_1, x_2, x_3\right)\\
&s.t.\\
&\qquad -5 \leq x_1 \leq 5.0\\
&\qquad \text{-5} \leq x_2 \leq 5.0\\
&\qquad \text{-5} \leq x_3 \leq 5.0\\
&\qquad 0 \leq x_1^2 + x_2^2 + x_3^2 \leq 4
\end{aligned}
$$

This problem has a higher dimensionality, but is mathematically simpler to resolve (convex). The figure below illustrates the input of the problem. We can here that the weighted sum can handle the problem as it is smooth and regular.



Descision variables

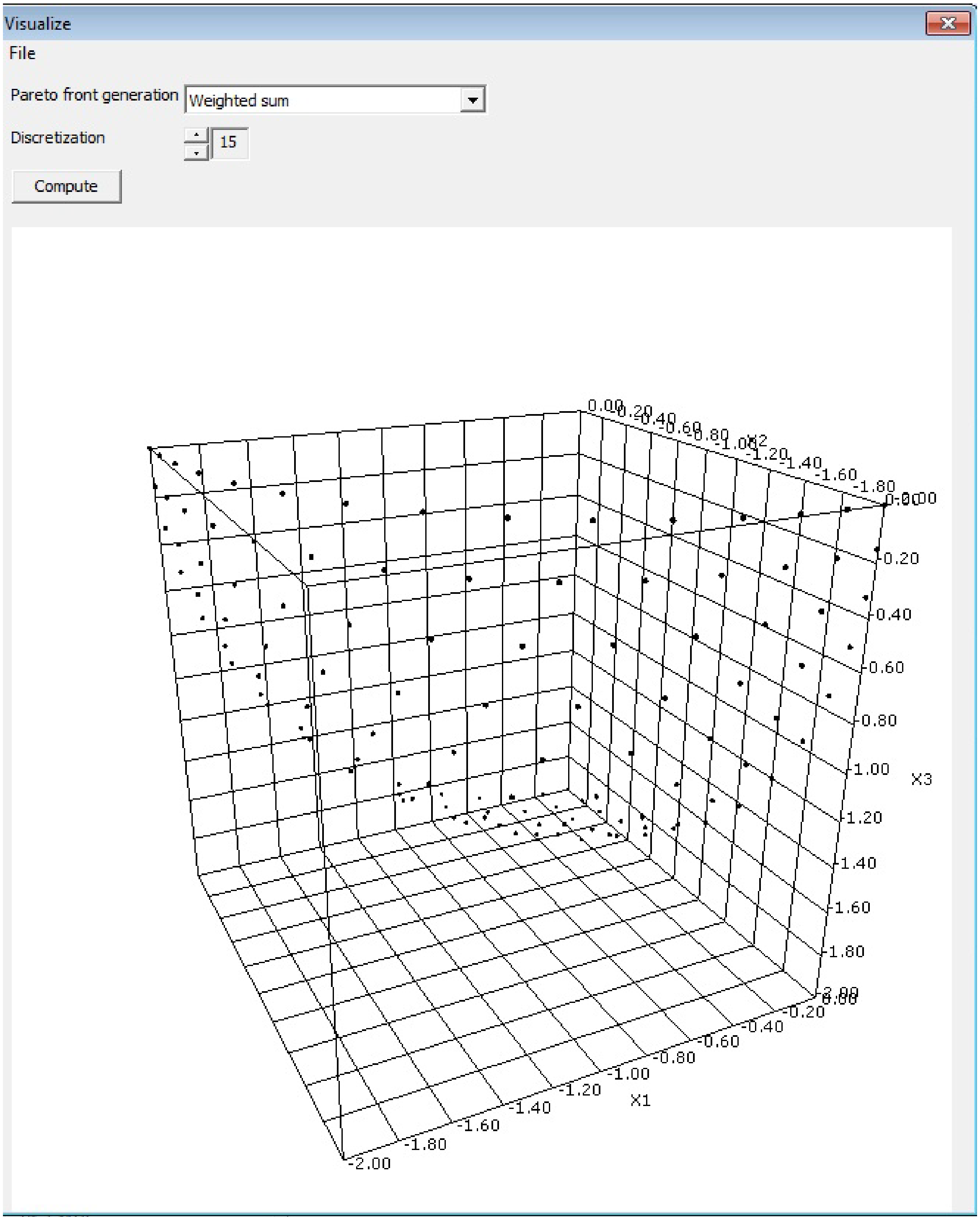
Id	Name	Description	Min	Max
X1	var1	myvar1	-5.00	5.00
X2	var2	myvar2	-5.00	5.00
X3	var3	myvar3	-5.00	5.00

Constraints

Id	Name	Expression	Min	Max
1	sagi	x1^2+x2^2+x3^2	0.00	4.00

Objectives

Id	Name	Expression
1	X1	x1
2	X2	x2
3	X3	x3



Pareto front generation: Weighted sum

Discretization: 15

6. Conclusions

This deliverable allows to define in what we believe the simplest way possible, complex problems of multi-criteria optimization that would otherwise require the need for programming skills and or the need for additional optimization languages. Instead, we have adopted the approach suggested by ACADO developers and further processed it so the user can define MCO problems from a graphical user interface.

The implemented methods (borrowed from ACADO) are among the more robust, efficient and state of the art. Even though, they may fail for specific problems and care has to be taken by the user to test various approaches and discretization grids.

Altogether, this module allows constructing the Pareto surface of MCO problems, thus allowing to make decisions and judgements on actual optima, quantitatively evaluated. This is fundamentally different from MCDM (multi-criteria-decision making) techniques that are mostly based on subjective and mostly qualitative judgement of alternatives.

The MCO module requires some expertise, at the difference of the MCDM techniques, but is rewarding in the sense that it provide actual values for decisions.

A full demonstration of the use of MCO for one of the sub-basins of the SMART study area is provided in WP08.