ePolicy

Engineering the Policy-making Life Cycle

Policy reasoning prototype evaluation V2

Document type:	Deliverable
Dissemination Level:	Public
Editor:	Michela Milano
Document Version:	0.1
Contributing Partners:	UNIBO, UNIFE
Contributing WPs:	WP3
Estimated P/M (if applicable):	5
Date of Completion:	30 June 2014
Date of Delivery to EC	30 June 2014
Number of pages:	16

ABSTRACT

The aim of this document is to evaluate the second version of the global optimization service producing a regional energy plan, described in the deliverable 3.4. The evaluation is performed by considering both functional and non functional requirements.

SEVENTH FRAMEWORK PROGRAMME

The project is co-funded by the European Community under the Information and Communication Technologies (ICT) theme of the Seventh Framework Programme (FP7/2007-2013). Grant Agreement n°288147.

Authors of this document:

Michela Milano[1], Marco Gavanelli [2]

[1]: DISI, University of Bologna
email: michela.milano@unibo.it
[2]: ENDIF, University of Ferrara
email: marco.gavanelli@unife.it

Contents

1 Introduction

This deliverable extends deliverable D3.3 [6], and details the evaluation of the version 2 of the Global Optimizer component of the ePolicy platform. We first recap the features of the Global Optimizer, then we list the new features of version V2 of the Global Optimizer in Section 3. We recap the requirements for evaluation in Section 4, and we report the results of the evaluation in Section 5.

2 Recap of the Global Optimization Service V1

In the ePolicy project, one of the components of the Decision Support System is the global optimizer [8, 4], that encapsulates the regional perspective for the definition of the Regional Energy plan. There are mainly two users of the service: the **policy maker** and the **environmental (domain) expert**, each accessing the service with different access roles and different functionalities. The policy maker can access the system by providing three inputs:

1. the total energy produced by renewable energy sources expected in the plan;
2. the minimum and maximum amount of energy per energy source;
3. a number of objective functions (taken from a list of environmental receptors and other factors such as cost, energy produced).

Note that the minimum amount of energy per energy source can be set for obtaining energy source diversification while the maximum amount of energy should respect a constraint on the regional receptivity (stated by the environmental expert). In output, the policy maker obtains a number of different scenarios, i.e., a number of regional plans that satisfy the given constraints and that are optimal (in the Pareto sense) with respect to the given objective functions.

The environmental expert instead has all the functionalities of the policy maker, but in addition he/she can configure the system by providing data on the policy domain. The environmental expert can provide:

1. matrices for performing the environmental assessment of plans;
2. the definition of primary activities for the plan, the definition of secondary activities and their relations thereof;
3. the costs related to primary and/or secondary activities;
4. maximum amount of energy per energy source that can be produced in the considered region; these numbers represent an upper limit to the maximum amount of energy that can be set by the policy maker.

The environmental expert can run the optimizer with the main goal of evaluating if the introduced numbers and matrices are indeed reasonable or should be adjusted.

In the following we report the UML diagrams representing the use case of the global optimizer utilization. The service in this case is considered as a stand-alone application, i.e., not integrated with other services.

In the requirement document, we have reported a detailed description of all these activities.

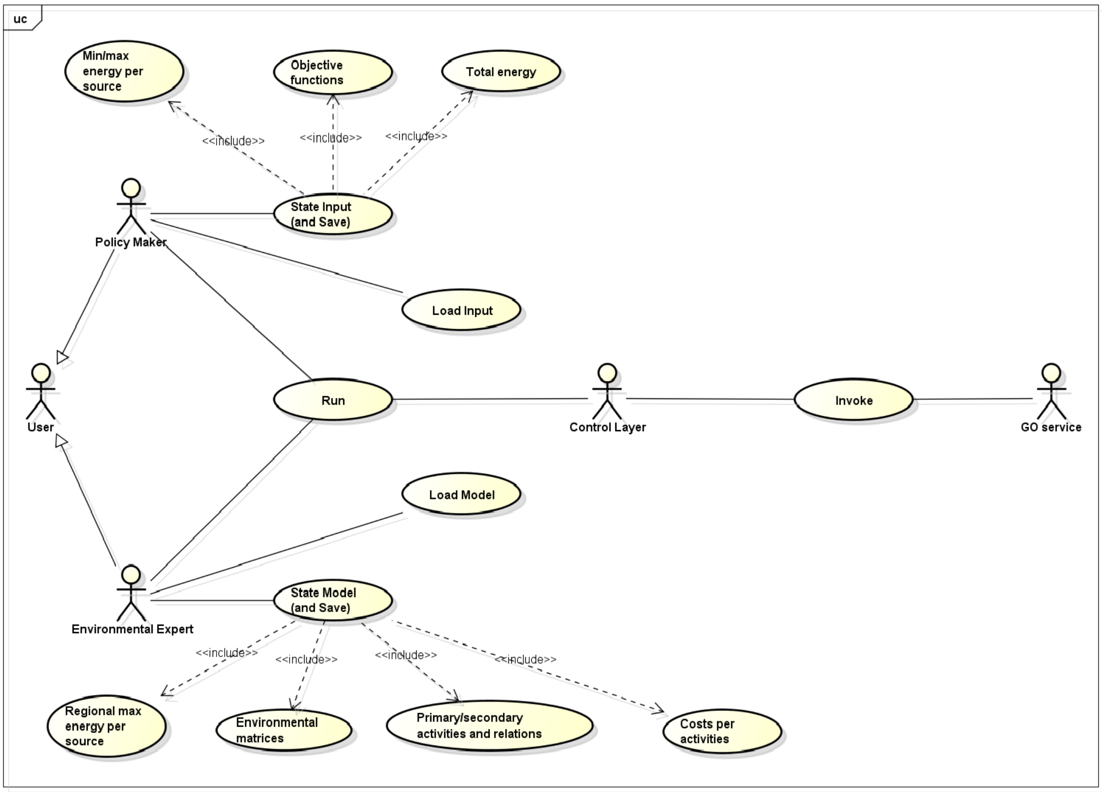

Figure 1: Use cases

3 New features of the Global Optimizer V2

The second version of the global optimizer was detailed in Deliverable 3.4 [5]; in short, the new features are:

1. while in version V1 (the magnitude of) activities could only take positive values (e.g., building new power plants, new buildings, etc.), in version V2 the policy makers asked to have the capability of decommissioning existing buildings, because it is foreseen that the next regional energy plan will require the decommissioning of existing non-renewable plants. This was implemented, in Version 2, by letting activities take negative values. This also implied that some of the constraints become non-linear, as explained in detail in deliverable 3.4 [5].
2. In version V1, all the environmental assessment was based on qualitative information (namely, the co-axial matrices). In version V2, we added some quantitative information, namely, the quantity of pollutants emitted in the atmosphere by the various types of power plants.
3. In order to compare different plans, with different types of emissions (e.g., comparing one plan in which power plants emit heavy metals in the air with a plan that emits carbon monoxide), we added the computation of indicators taken from an EU official document [3]; these indicators provide the effect of the plan on human health, global warming, and acidification.
4. We provided a software able to compute multi-objective optimization. While in version V1 the software was able to perform bi-objective optimization only with

two predefined objective functions, in version V2 the user can choose any number of objective functions (so the new version is not limited to two functions); moreover the functions that can be optimized are not predefined, but they can be

- the cost
- the electrical/thermal installed power
- the electrical/thermal energy produced in a year
- any receptor
- any emission
- any indicator
- any linear combination of the above.

4 Requirements

In Deliverable 2.3 [7], we identified a number of functional and nonfunctional requirements for the Global Optimizer. We recap them here.

4.1 Functional Requirements

Concerning functional requirements, we have identified a number of functionalities of the system concerning its configuration and its usage.

We test the functional requirements by considering two lists of requirements: one for the functionalities required by the environmental expert and one for the functionalities required by the policy maker. By analyzing the code and testing it on data from the RER Energy plan 2011-2013 plus other instances, we tick the implemented functionalities.

At the end of the test, the evaluation deliverable 2.3 [7] states that if the number of functional requirements implemented is lower or equal to the 60% of the requirements contained in the requirement document, then the component evaluation has to be considered unsatisfactory. If this number is between 60 and 80%, the evaluation has to be considered sufficient, while if the number is higher than 80% the evaluation is successful.

4.2 Non Functional Requirements

We report here the section on non functional requirements reported in the deliverable 2.3 on Means for Project Evaluations [7], which has been submitted to the commission at month 20.

> The nonfunctional requirements will be evaluated by running tests and by performing field tests involving users and asking them to evaluate the software through interviews and questionnaires. In particular, we will consider the following means for evaluation of non-functional requirements:
>
> - Scalability, performance and efficiency: we plan to run a battery of tests, both on the first version of the prototype (whose evaluation is due at month 24, D3.3) and on the final version (whose evaluation is due at month 33, D3.5) by using the data from the RER regional plan of 2011, the

new data of 2014 and also a set of synthetic instances generated by changing the matrix coefficients, increasing the number of objective functions, changing constraints, and increasing the number of activities. For these tests we will extract the running time, the number of explored nodes, the mean and the variance of the tests. We consider this test satisfactory if for running the system on a real plan, the system has a response time less than five minutes.

- Cost of development and time of development: these figures will be extracted by the project time sheets measuring the time and the cost of MM exposes on these activities. We consider the evaluation of these parameters succesful if the numbers are in line with what expected in the project proposal. In particular, if the variation of time and cost is lower than the 5% with respect to the data provided in the budget.
- Accuracy, precision and re-configurability: these figures will be measured by using questionnaires, directed to policy makers (that test the first two requirements) and to environmental experts (evaluating all of them). If more than the 70% of users are satisfied with these features (derived from questionnaires) the evaluation has to be considered successful.

For each category of users we will evaluate the component by providing questionnaires customized for policy makers and for environmental experts that are devoted not to assess the visual interface of the component and its friendliness, but rather to assess the correctness of the results. We plan to involve policy makers of the RER, which is a partner of the project, and, after a training period, to use the global optimizer for the Regional Energy Plan 2014, therefore on a real plan. This will give us the possibility to evaluate the system on real setting, provide alternatives that could be evaluated on environmental, energy and economic aspects by the policy maker. We will report the experience of this test both on the web site and in a summary document, deliverable D3.5, due at month 33.

We also plan to make a trial by contacting policy makers from other regions contacted by the RER. The idea is to guide them in defining a plan by checking the functionalities they expect from a system like that. It is worth mentioning that the system requirements have been designed by looking at several plans, but they have been mainly tailored on the requirement stated by RER experts. Therefore, this test is devoted to understand how the functionalities of system is suitable for other regions, or if new requirements arise from this analysis. Again these policy makers will be appointed and their evaluation could be captured through a questionnaire and reported in the Exploitation Plan, due at month 28.

In parallel, we plan to measure the configurability of the system by asking several experts from ARPA Emilia-Romagna, which is partner of the Advisory and Dissemination Board of the project, to configure the system. During this activity, the environmental experts will also query the system for generating regional plans in order to check if the configuration is correct. Therefore the questionnaire of the environmental experts will be a superset of the one devoted to policy makers. Again this evaluation will be reported in the deliverable D3.5 due at month 33.

5 Prototype Evaluation V2

In order to separate the evaluation of the Global Optimizer prototype from the one of the visual interface, contained in deliverable 7.2, we have implemented a web service wrapping the solver. The web service provides an interface for entering the data and a GUI that is devoted to ease the system use for a non IT expert. We are fully aware that this GUI is indeed not as user friendly as the interface described in deliverable 7.1, but it is a way to easily evaluate the correctness of the solutions provided by the system.

Beside the graphic front-end version 1, that has been described in deliverable 2.3 [7] and is available at

`http://globalopt.epolicy-project.eu/GlobalOptClient/ePolicy.jsp`,

we developed a graphic front-end version 2 that can be used by accessing the following link

`http://globalopt.epolicy-project.eu/Pareto/`

and that includes the new features detailed earlier.

The service available at this link is for testing purposes only. Notice that it does not represent the real Graphical User Interface (GUI) of the ePolicy prototype.

5.1 Functional Requirement Evaluation

The system has been evaluated as far as functional requirements are concerned. We have considered two tables: one concerning the features available for the environmental expert (Figure 2) and one concerning the tasks of the policy maker (Figure 3).

As we can see, the system implements 100% of the functional requirements identified in the requirement document. As far as the graphic front-end, which was not required nor described in the Description of Work of ePolicy [1], some configuration tasks were not implemented in the version 1. The second version, instead, accepts in input an Excel file containing the co-axial matrices, the relationships between primary and secondary activities, and the costs of the activities. The import of Excel spreadsheets is described in Deliverable 8.2 [2]. Notice that the use of Excel spreadsheets is the method currently used by the environmental experts at ARPA Emilia-Romagna to contain all the information about the matrices used for the environmental assessment; for this reason, this choice was appreciated very much by our environmental experts.

As far as the policy maker is concerned, the system version V1 already implemented

Activity	System	GUI V1	GUI V2
Set regional max energy per source	yes	no	yes
Change environmental matrices	yes	no	yes
Change primary/ sec activity matrix	yes	no	yes
Def. cost per activity	yes	no	yes
Min and max energy	yes	yes	yes
State objective functions	yes	yes	yes
Set total energy	yes	yes	yes
Run system	yes	yes	yes
Obtain results	yes	yes	yes

Figure 2: Environmental expert activity implementation

Activity	System	GUI
Min and max energy	✔	✔
State objective functions	✔	✔
Set total energy	✔	✔
Run system	✔	✔
Obtain results	✔	✔

Figure 3: Policy Maker activity implementation

100% of the functional requirements identified in the requirement document and these functionalities are also handled by the graphic front end.

For these reasons, we consider that the functional requirement evaluation has been successful.

5.2 Non functional Requirement Evaluation

As stated in Deliverable 2.3 [7], a large part of non functional requirements involve an evaluation from experts.

Performance, efficiency and scalability. As far as performance and efficiency are concerned, we have run a battery of tests by using the data from the RER regional plan of 2011; for all these tests, the running time on the single objective case was less than a sec-

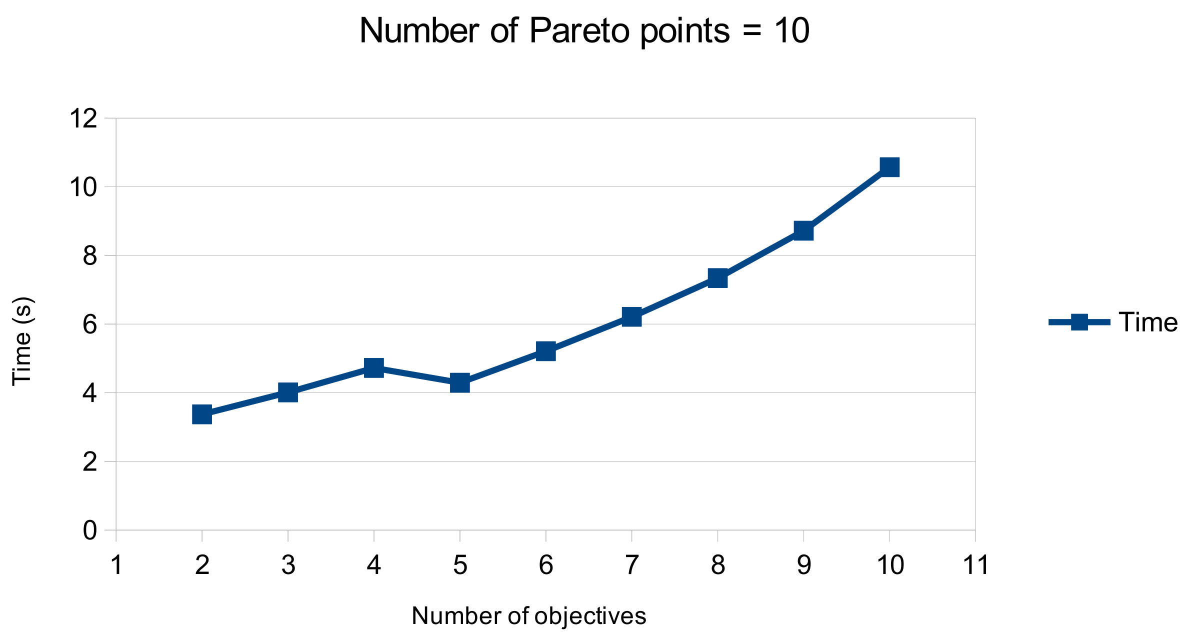

Figure 4: Tests on the real instance of the 2011-13 Regional Energy Plan of the Emilia-Romagna region, varying the number of objective functions. Number of requested Pareto points = 10.

ond. In the multi-objective case, the runtime depends both on the number of objective functions, and on the number of points that are requested on the Pareto front. In Figure 4, we show the runtime versus the number of objective functions, fixing a number of Pareto points to 10. As we can see, the runtime grows almost linearly with the number of objectives.

In order to assess the scalability of the software, we adopted the same procedure explained in Deliverable 3.3 for the mono-objective version, but adapted to a multi-objective context. Namely, we performed a series of tests by randomly generating a set of data, including the co-axial matrices, the matrix relating primary and secondary activities, the activity costs, etc. In this way, we were able to stress-test the software with instances containing a number of activities, pressures and receptors larger than those in the actual data provided by ARPA.

The matrices currently used for the environmental assessment by ARPA contain 93 activities, 48 pressures and 23 receptors.

In the random instances, we used a parameter N that represents the size of the instance. Given a value N, we randomly generated N activities, N pressures and N receptors. Amongst the activities, a random number was selected as *primary*, and the rest are *secondary* activities. For each pair primary-secondary activity, we randomly decided weather that primary activity required the secondary with probability 0.5. If the primary required that secondary activity, the amount of secondary activity required for one unit of the primary was randomly generated in the interval 0-10.

Similarly, we generated the two matrices relating activities and pressures, and the one relating pressures and receptors. These matrices contain, in the real instance, qualitative values *high*, *medium* and *low* (or there can be no value, meaning that there is no relation between the given activity and pressure, or pressure and receptor). In the random instances, each cell of these matrices contains no value (meaning no relation) with probability $1/2$, and, in case the cell contains a non-null value, the values *high*, *medium* and *low* were selected with equal probability.

While in the mono-objective case we had optimized a single objective function, in the multi-objective case, we considered a number of objectives. In particular, we selected those objectives that are most likely to be of interest to a policy maker, namely:

1. maximizing the electric energy produced in a year
2. minimizing the cost
3. maximizing one of the environmental receptors.

We ran a number of tests varying the number of objective functions in the order depicted above, i.e., the first objective was always maximizing the electric energy, the second (if present) was minimizing the cost, the third was maximizing the first receptor, the fourth was maximizing the second receptor, and so on.

The run time of the adopted algorithm also depends on the number of points that the user requests on the Pareto front; for details about the implemented algorithm, please refer to Deliverable 3.4 [5].

The experiments were performed on a laptop computer running Linux with a 8x Intel Core i7-3720QM CPU at 2.60GHz; note however that only one core was used in the experiments.

In Figure 5, we report the run times obtained when computing 5 points on the Pareto frontier consisting of 2 objective functions. Each point in the plot represents a test. The x-axis is the size of the instance, i.e., the N parameter mentioned earlier. The y-axis represents the running time of the instance, i.e., the time required to find the optimal solution. We can see that the runtime grows with the size of the instance. However, even in the largest instances, the runtime was below 30 seconds, compared with the 5 minutes that were considered as success in the Means for Project Evaluation [7]. Note that in the largest considered instance, the co-axial matrices have size $160 \times 160 = 25,600$ while the largest of the two co-axial matrices in the real instance has size $93 \times 48 = 4,464$, which is about 6 times smaller.

In the following graphs, we experimented with a higher number of objective functions.

In Figure 6 we plot a series of experiments that consider 5 objective functions, with a variable number of Pareto points, taken from the set $\{10, 20, 40, 50\}$. As expected, the running time increases as the size of the size of the instance grows, and as the number of requested Pareto points grows. Still, with sizes of about 130 (which gives matrices of $130 \times 130 = 16,900$, about four times larger than the largest matrix in the real instance),

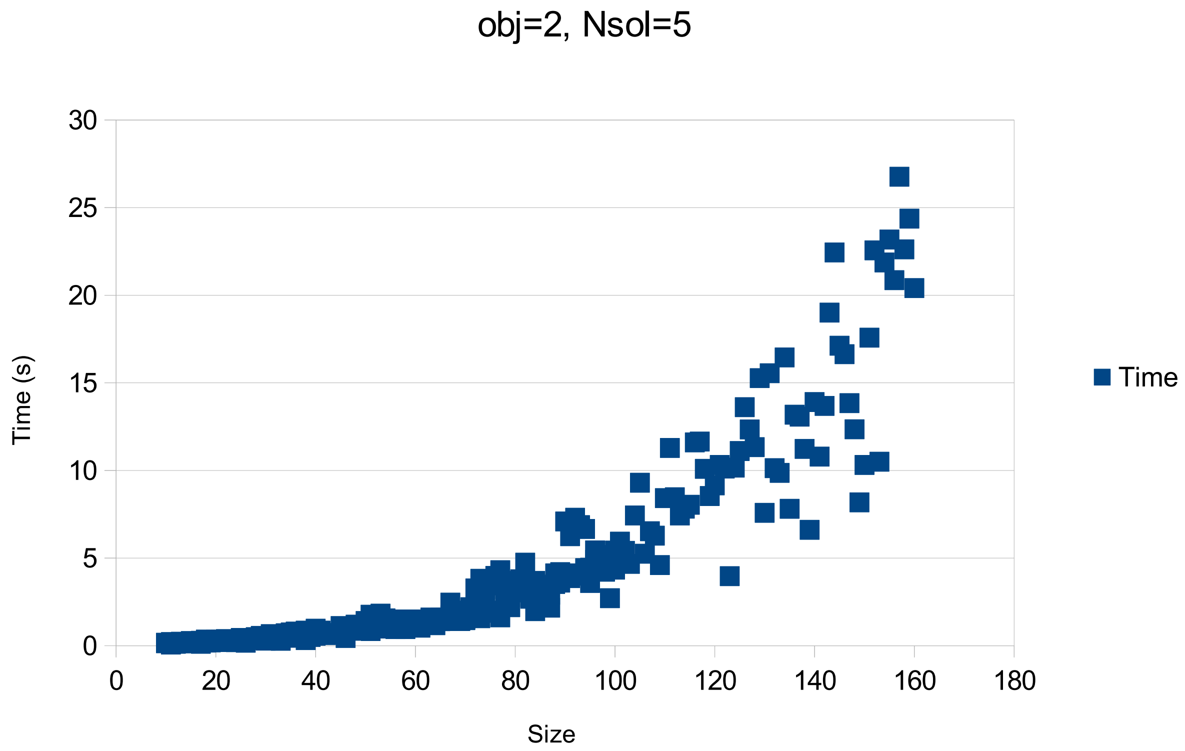

Figure 5: Stress test on randomly generated problems. The y axis plots the running time in seconds, while the x-axis is the size N of the problem, considering the same number N of activities, pressures and receptors. Number of objective functions=2, number of requested Pareto points = 5.

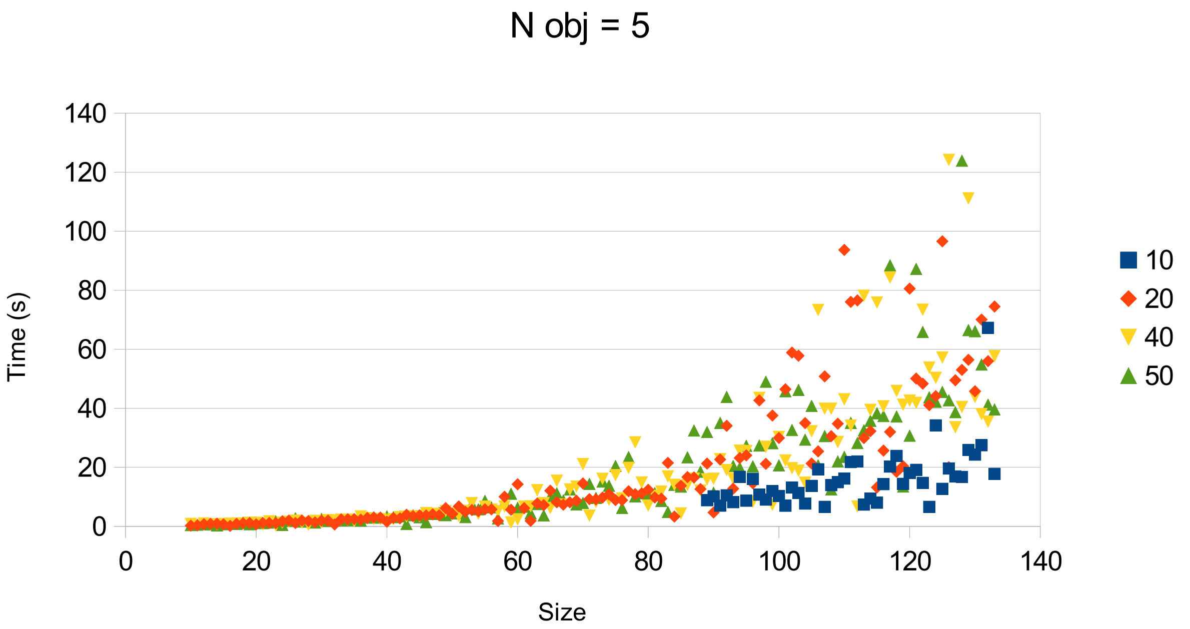

Figure 6: Stress test on randomly generated problems. The y axis plots the running time in seconds, while the x-axis is the size N of the problem, considering the same number N of activities, pressures and receptors. In all instances, the number of objective functions is 5, while the number of requested Pareto points varies from 10 to 50.

with a number of Pareto points of 50 (which is a very large number of points to compare), the runtime is about 2 minutes.

We then experimented with a larger numbers of objective functions. In Figure 7 we show the experiments with 10 objectives and 20 Pareto points. As we can see, the graph is qualitatively similar to the previous ones, and we can see that for problems of size 100 (with matrices of size 10,000, about twice the size of the real instance), we can solve the instances in reasonable time, well below the limit of 5 minutes declared in deliverable 2.3 [7]. The first instance that took over 5 minutes (300s) has size 139, much larger than the real instance.

Since the system has a response time less than five minutes, according to the requirement given in [7], the test is considered satisfactory also for the multi-objective case.

Finally, we experimented increasing the number of objective functions, for a large instance, of size 100, to test the scalability with respect to the number of objectives while considering an instance significantly larger than the real one. The result is plot in Figure 8; the first instance that took more than 5 minutes has 19 objective functions, a number of objectives that is rather complex for a policy maker to consider and compare.

Cost and time of development. As far as the cost of development and time of development are concerned, the person months are more than expected (114% of expected time);

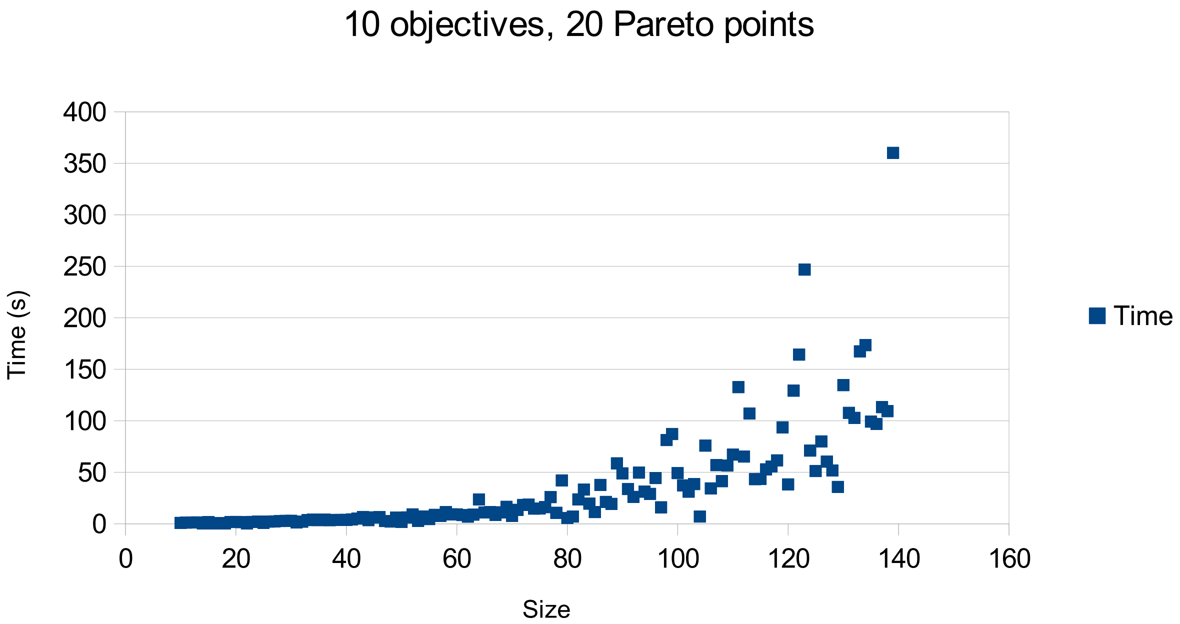

Figure 7: Stress test on randomly generated problems. The y axis plots the running time in seconds, while the x-axis is the size N of the problem, considering the same number N of activities, pressures and receptors. Number of objective functions=10, number of requested Pareto points = 20.

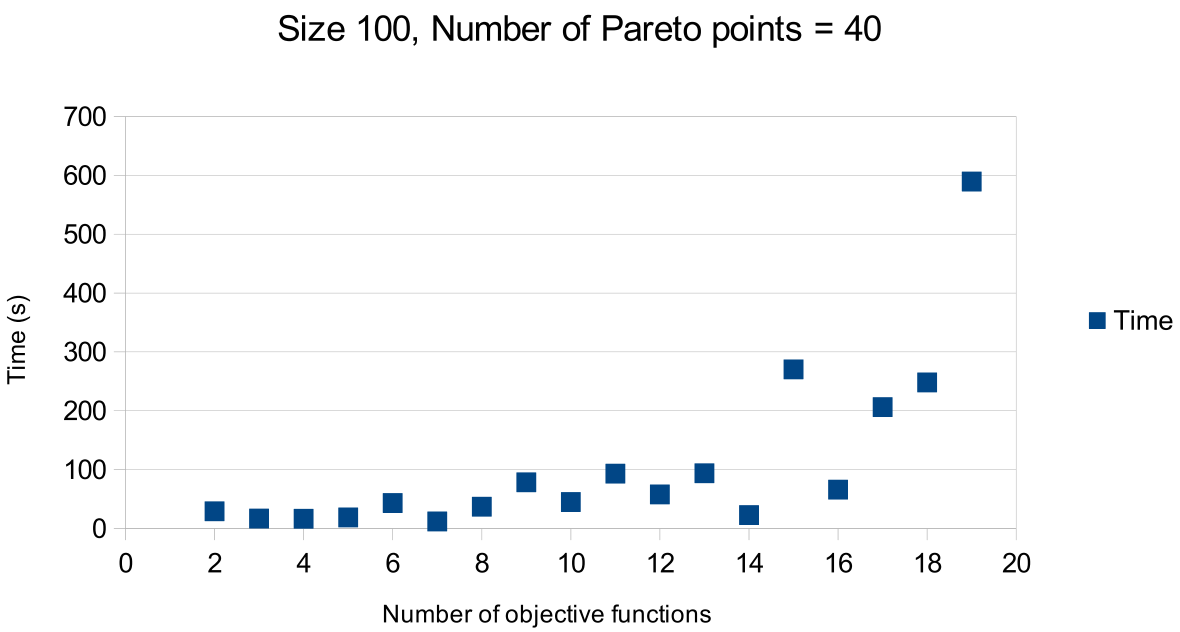

Figure 8: Stress test on randomly generated problems. The y axis plots the running time in seconds, while the x-axis is the number of objective functions. We considered 20 points on the Pareto front.

however the cost of development was not increased with respect to what was expected. In fact, in the project some persons with lower experience were hired, and, due to the lower experience, they had higher development time. Note also that all the deadlines were met, so we believe that also the time of development can be considered satisfactory.

Accuracy, precision and re-configurability. Concerning accuracy, precision and re-configurability, we have contacted policy makers in the Emilia-Romagna Region, and environmental experts in ARPA, which is also a partner of the advisory and dissemination board.

To measure re-configurability, we assigned to the environmental experts in ARPA the task to modify the data in the Excel file containing all the matrices, and to import them on the web application. All the experts were able to complete this task, so we believe the operation was successful.

We are currently providing questionnaires to participants to the *Samos 2014 Summit on ICT-enabled Governance*, that is an international event on governance; in 2014 it is held from the 30th of June to the 5th of July. We will report the results in Deliverable D9.3.

The software was also shown to

- policy makers in the Piemonte region of Italy.
  We are currently running a series of tests considering the data extracted from a previous Regional Energy Plan of the Piemonte region. The global optimizer has been shown to the policy makers and they found the prototype very user friendly and usable.
- Companies: ABB and VSY. In particular ABB is a multi-national company working in a number of fields related to energy, with interests in sustainability. VSY instead is a naval shipyard and they are interested in marine policies and in the assessment of boat construction process using the ePolicy methodology. Also in this case both presentations have sorted a very good feedback as far as the usability and user friendliness are concerned.
- The members of the Advisory and Dissemination Board. ARPA and ENEL in particular have tested the system. The feedback has been very positive. They have been able to use the system and to find and assess alternative energy plans.
- experts at the Italian Ministry for economic development: The global optimizer has been shown to the policy makers and they found the prototype very user friendly and usable.

We will provide a comprehensive list of the result in Deliverable D9.3; however, the first impressions were very encouraging.

References

[1] Description of work. ePolicy project, June 2011.

[2] Federico Chesani, Ambra Molesini, Michela Milano, Luis Torgo, Tobias Ruppert, Tina Balke, Marco Gavanelli, and Barry O'Sullivan. Integrated policy decision support system implementation. In Federico Chesani and Ambra Molesini, editors, *Deliverable of the ePolicy project*, number 8.2. ePolicy project, June 2014.

[3] European Commission. Integrated pollution prevention and control reference document on economics and cross-media effects, July 2006. Available at `http://eippcb.jrc.es/reference/ecm.html`.

[4] Marco Gavanelli, Michela Milano, Federico Chesani, and Elisa Marengo. Prototype of the global level policy reasoning system V1. In Marco Gavanelli, editor, *Deliverable of the ePolicy project*, number 3.2, pages 1–31. ePolicy project, May 2013.

[5] Marco Gavanelli, Michela Milano, Federico Chesani, Elisa Marengo, and Stefano Bragaglia. Prototype of the global level policy reasoning system V2. In Marco Gavanelli, editor, *Deliverable of the ePolicy project*, number 3.4. ePolicy project, March 2014.

[6] Michela Milano. Policy reasoning prototype evaluation V1. In Michela Milano, editor, *Deliverable of the ePolicy project*, number 3.3. ePolicy project, September 2013.

[7] Michela Milano, Federico Chesani, Stefano Bragaglia, Tina Balke, Nigel Gilbert, Lars Kotthoff, Barry O'Sullivan, Luís Torgo, Tobias Ruppert, Sabrina Franceschini, Daniele Sangiorgi, Tony Woods, and Marco Gavanelli. Means of project evaluation. In Michela Milano, editor, *Deliverable of the ePolicy project*, number 2.3, pages 1–19. ePolicy project, May 2013.

[8] Michela Milano, Attilio Raimondi, Marco Gavanelli, and Fabrizio Riguzzi. Design of the global regional model and its instantiation on the energy plan. In Michela Milano, editor, *Deliverable of the ePolicy project*, number 3.1, pages 1–42. ePolicy project, October 2013.