OMOptim

OpenModelica Optimization Editor

User Manual
Version 0.9

April 2011

Peter Fritzson
Hubert Thieriot, Mohsen Torabzadeh-Tari

Supported by:
Open Source Modelica Consortium

Table of contents

Chapter 1 : Introduction

OMOptim is a tool dedicated to optimization of Modelica models. By optimization, one should understand a procedure which minimizes/maximizes one or more objectives by adjusting one or more parameters.

OMOptim 0.9 contains meta-heuristic optimization algorithms which allow optimizing all sorts of models with following functionalities:

- One or several objectives optimized simultaneously
- One or several parameters (integer or real variables)

However, the user must be aware of the large number of simulations and optimization might require.

Chapter 2 : Preparing model

Before launching OMOptim, one must prepare its model in order to optimize it.

2.1 Parameters

An optimization parameter is picked up from all model variables. The choice of parameters will be done in the OMOptim interface.

For all intended parameter, please note that:

- The corresponding variable is *constant* during all simulation. The OMOptim optimization in version 0.9 only concerns static parameters' optimization *i.e.* values found for these parameters will be constant during all simulation time.
- Corresponding variable should play an *input* role in the model *i.e.* its modification influences model simulation results.

2.2 Constraints

If some constraints should be respected during optimization, they must be defined in Modelica model itself.

For instance, if mechanical stress must be less than 5 $N.m^{-2}$, one should write in the model:

```
assert( mechanicalStress < 5, "Mechanical stress too high");
```

If during simulation, the variable *mechanicalStress* exceeds 5 $N.m^{-2}$, simulation will stop and be considered as a failure.

2.3 Objectives

As parameters, objectives are picked up from model variables. Objectives' values are considered by the optimizer at the final time.

Chapter 3 : Set problem in OMOptim

3.1 Launch OMOptim

OMOptim can be launched using the executable placed in `OpenModelicaInstallationDirectory/bin/OMOptim/OMOptim.exe`. Alternately, choose `OpenModelica > OMOptim` from the start menu.

3.2 Create a new project

To create a new project, click on menu `File -> New project`.

Then set a name to the project and save it in a dedicated folder. The file created is has a .min extension. It will contain information's regarding model, problems and results loaded.

3.3 Load models

First, you need to load model(s) you want to optimize. To do so, click on *Add .mo* button on main window or select menu *Model -> Load Mo file…*

When selecting a model, file will be loaded in OpenModelica which runs in background.

While OpenModelica is loading the model, you could have a frozen interface. This is due to multi-threading limitation but delay should be short (few seconds).

You can load as many models as you want.

If an error occurs (indicated in log window), this might be because:

- Dependencies have not been loaded before (e.g. modelica library)
- Model use syntax incompatible with OpenModelica.

3.3.1 Dependencies

OMOptim should detect dependencies and load corresponding files. However, if some errors occur, please load by yourself dependencies. You can also load Modelica library using `Model->Load Modelica library`.

When model correctly loaded, one should see a window similar to Figure 1.

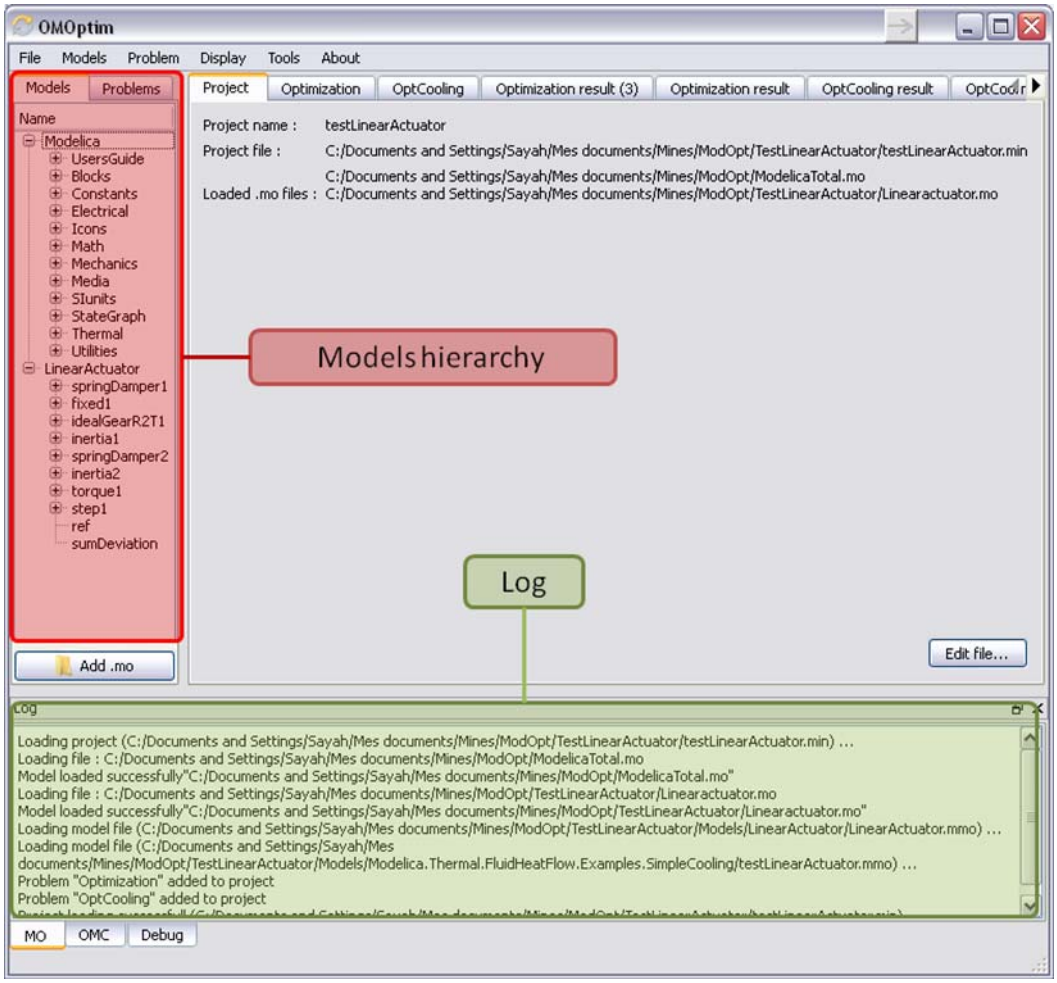

Figure 1 - OMOptim window after having loaded model

3.4 Create a new optimization problem

`Problem->Add Problem->Optimization`

A dialog should appear. Select model you want to optimize. Only Model can be selected (no Package, Component, Block…).

A new form will be displayed. This form has two tabs. One called *Variables*, one called *Optimization*.

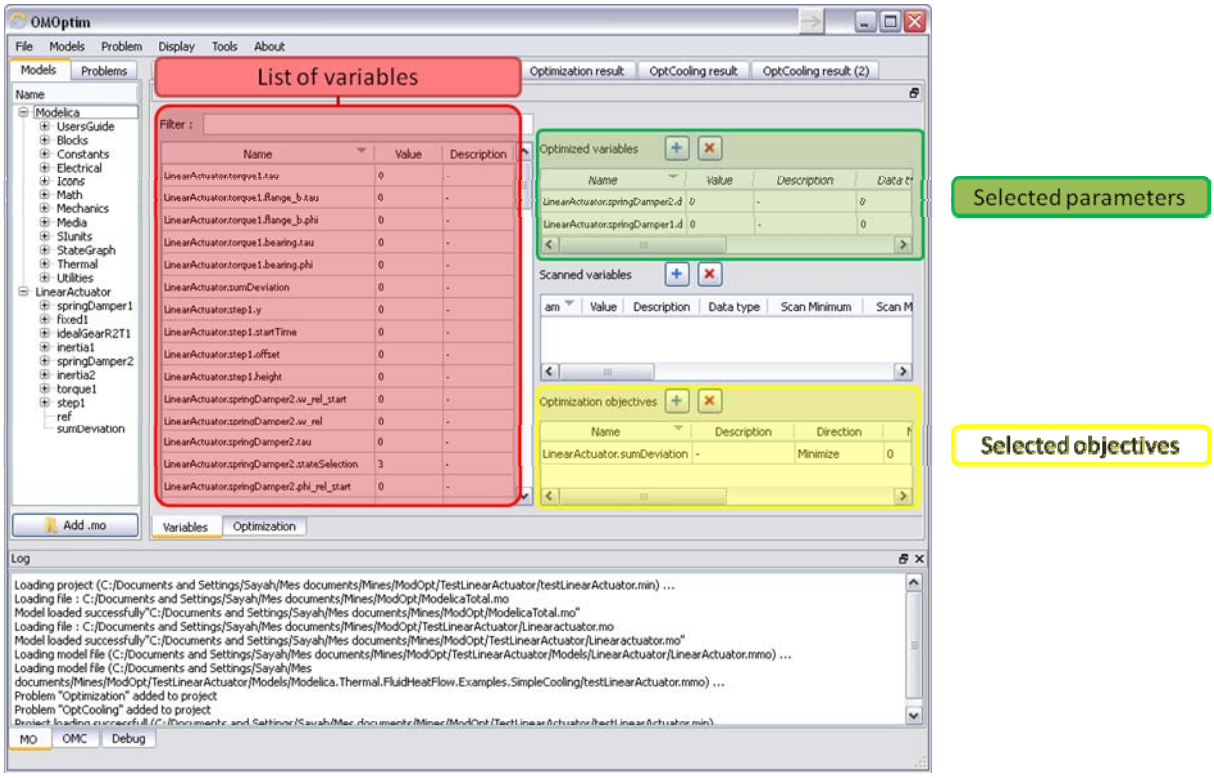

Figure 2 - Forms for defining a new optimization problem.

3.4.1 List of variables is empty

If variables are not displayed, right click on model name in model hierarchy, and select *Read variables*.

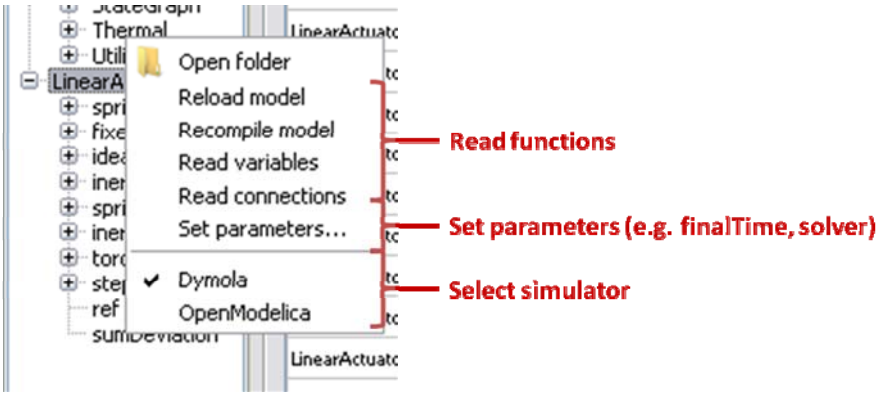

Figure 3 - Selecting read variables, set parameters, and selecting simulator.

3.5 Select optimized variables

To set optimization, we first have to define variables optimizer will consider as free *i.e.* those he should find best values of. To do this, select in the left list, the variables concerned. Then, add them to *Optimized variables* by clicking on corresponding button (✚).

For each variable, you must set minimum and maximum values it can take. This can be done in the *Optimized variables* table.

3.6 Select objectives

Objectives correspond to the final values of chosen variables. To select these last, select in left list variables concerned and click + button of *Optimization objectives* table.

For each objective, you must:

- Set minimum and maximum values it can take. If a configuration does not respect these values, this configuration won't be considered. You also can set minimum and maximum equals to "-" : it will then
- Define whether objective should be minimized or maximized.

This can be done in the *Optimized variables* table.

3.7 Select and configure algorithm

After having selected variables and objectives, you should now select and configure optimization algorithm. To do this, click on *Optimization* tab.

Here, you can select optimization algorithm you want to use. In version 0.9, OMOptim offers three different genetical algorithms. Let's for example choose SPEA2Adapt which is an auto-adaptative genetical algorithm.

By clicking on *parameters*… button, a dialog is opened allowing defining parameters. These are:

- Population size: this is the number of configurations kept after a generation. If it is set to 50, your final result can't contain more than 50 different points.
- Off spring rate: this is the number of children per adult obtained after combination process. If it is set to 3, each generation will contain 150 individual (considering population size is 50).
- Max generations: this number defines the number of generations after which optimization should stop. In our case, each generation corresponds to 150 simulations. Note that you can still stop optimization while it is running by clicking on *stop* button (which will appear once optimization is launched). Therefore, you can set a really high number and still stop optimization when you want without losing results obtained until there.
- Save frequency: during optimization, best configurations can be regularly saved. It allows to analyze evolution of best configurations but also to restart an optimization from previously obtained results. A Save Frequency parameter set to 3 means that after three generations, a file is automatically created containing best configurations. These files are named iteraion1.sav, iteration2.sav and are store in *Temp* directory, and moved to *SolvedProblems* directory when optimization is finished.
- ReinitStdDev: this is a specific parameter of EAAdapt1. It defines whether standard deviation of variables should be reinitialized. It is used only if you start optimization from previously obtained configurations (using *Use start file* option). Setting it to yes (1) will, in most of cases, lead to a spread research of optimized configurations, forgetting parameters' variations' reduction obtained in previous optimization.

3.7.1 Use start file

As indicated before, it is possible to pursue an optimization finished or stopped. To do this, you must enable *Use start file* option and select file from which optimization should be started. This file is an *iteration_.sav* file created in previous optimization. It is stored in corresponding *SolvedProblems* folder (*iteration10.sav* corresponds to the tenth generation of previous optimization).

Note that this functionality can only work with same variables and objectives. However, minimum, maximum of variables and objectives can be changed before pursuing an optimization.

3.8 Launch

You can now launch Optimization by clicking *Launch* button.

3.9 Stopping optimization

Optimization will be stopped when generation counter will reach generation number defined in parameters. However, you can still stop optimization while it is running without losing obtained results. To do this, click on *Stop* button. Note that this will not immediately stop optimization: it will first finish current generation.

This stop function is especially useful when optimum points do not vary anymore between generations. This can be easily observed since at each generation, optimum objectives values and corresponding parameters are displayed in log window.

Chapter 4 : Results

Result tab appear when optimization is finished. It consists in two parts: a table where variables are displayed and a plot region.

4.1 Get all variables values

During optimization, the values of optimized variables and objectives are memorized. The others are not. To get these last, you must recomputed corresponding points. To achieve this, select one or several points in point's list region and click on *recompute*.

For each point, it will simulate model setting input parameters to point corresponding values. All values of this point (including those which are not optimization parameters neither objectives).

Chapter 5 : Annex: Window regions

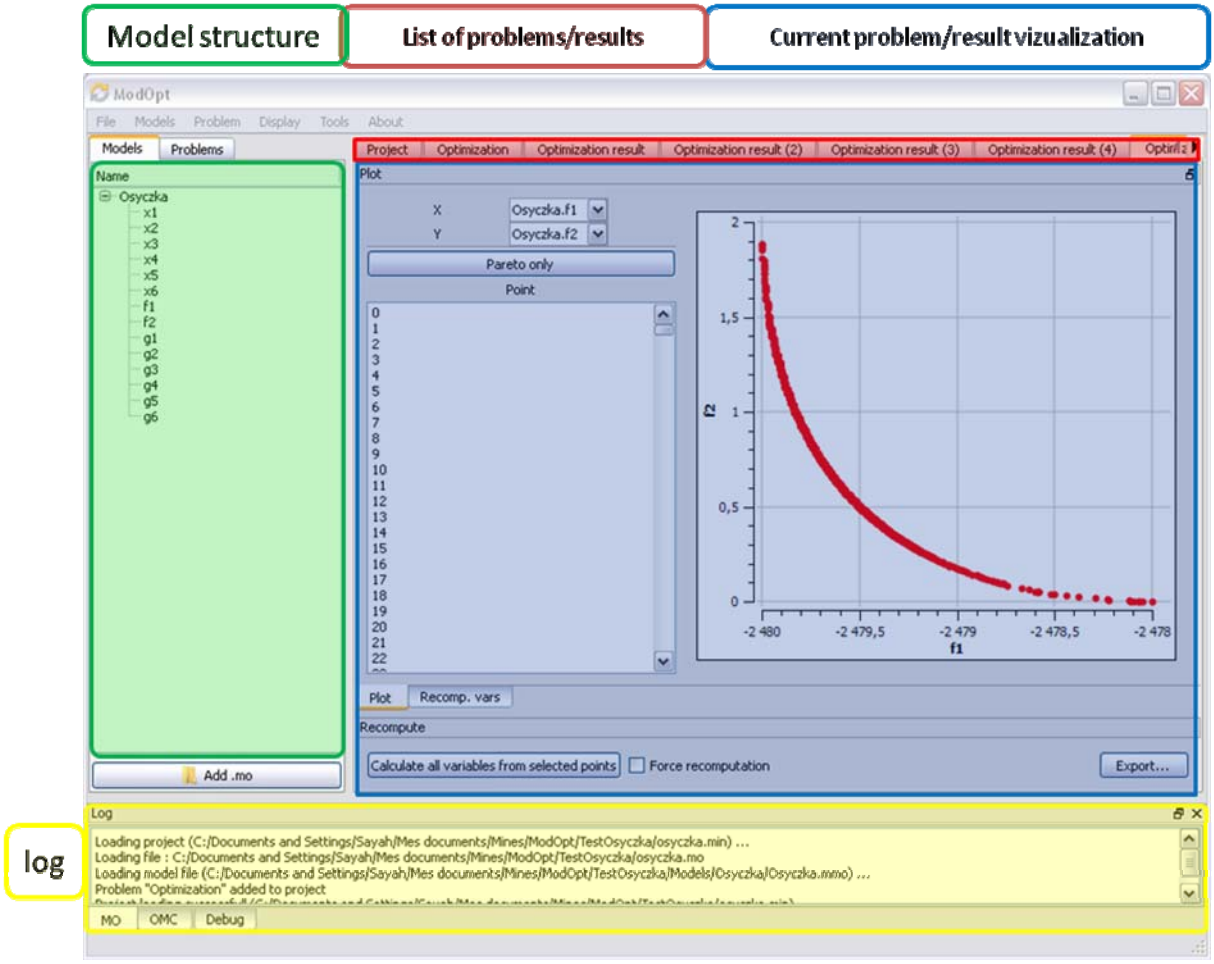

Figure 4 - Window regions in OMOptim GUI.