UM Experiments Module

User's manual

2014

Contents

6. UM Experiments module

6.1. Introduction

UM Experiments module is developed to help the researcher to analyze dynamical behavior of models and fulfill their parametrical optimization. Manual optimization according to Figure 6.1 is a quite time-consuming and laborious process where researcher has to control many different parameters of a model and settings that often leads to mistakes.

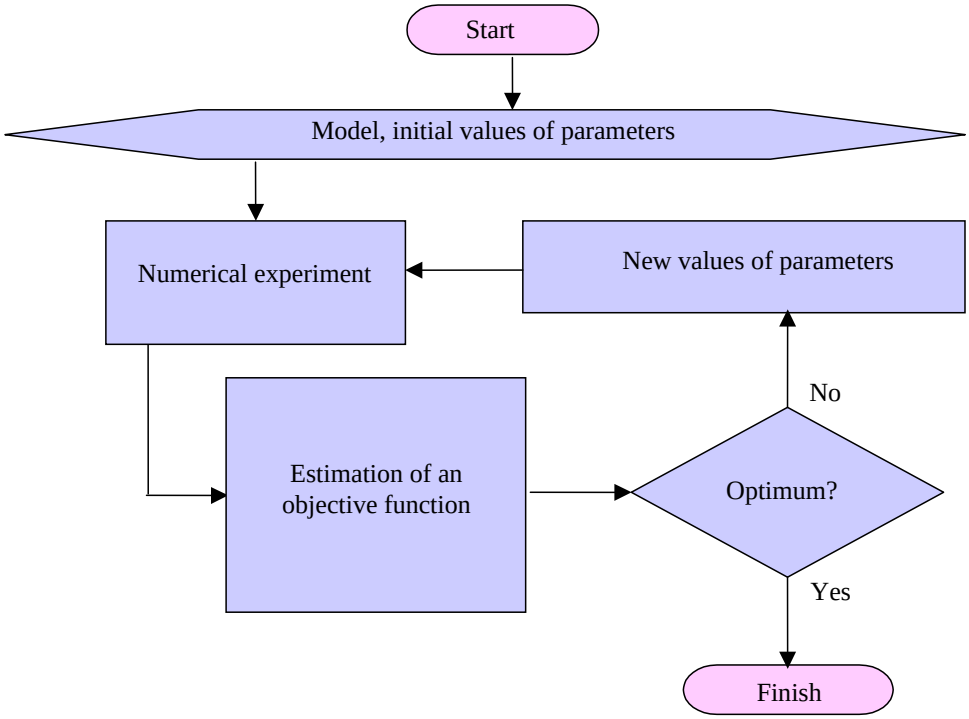

Figure 6.1. Optimization process

UM Experiments module is built in the **UM Simulation** program and includes the tool for describing and executing scanning projects (*scanning*) and service of distributed calculations.

It is often required in engineering practice to carry out series of numerical experiments, for example to analyze dynamical behavior and sensitivity of mechanical system or to find out optimal parameters of the system. **UM Experiments** includes a set of tools (*scanning, service of distributed calculations*) for advanced analysis of dynamics of mechanical systems.

All tools automate fulfillment of series of numerical experiments and save results of experiments on a hard disk for posterior analysis. Thus, the designer is released from monotonous execution of series of numerical experiments "manually" what saves working hours and removes errors, which people unfortunately usually do. In other words, the researcher defines the design of experiments for scanning and approximation or parameters, its limits, precision and goal function for optimization. Then the project is started and executed automatically. Current process statistics is available during the execution: number of executed experiments, time left. Series of numerical experiments are resistant to shut-down of power supply. In that case all results are saved on a hard disk and only results of the latter experiments are lost.

There is a possibility to plot an oscillogram of any saved performance of dynamical behavior of a mechanism. Moreover, the designer can plot so-called summary graphs and surfaces. All implemented tools have no limitation in number of parameters. In other words, they all are multi-parametrical. Please, note that the demo version allows describing the projects of scanning (optimization, approximation) with no more than one parameter. Dimensions of projects are set by the designer so as to solve the specified problem. But on the other hand the designer has to take into account calculating efforts that are necessary for the project and find a compromise between them.

Every tool has its own merits and demerits. However they all give the designer possibilities to solve quite many problems devoted to optimization of mechanical systems.

After the execution of series of numerical experiments for scanning the problem of choosing the optimal parameters arises. How can the designer find out the best solution from lots of alternatives? It becomes much more difficult in the case of conflicting criteria. In order to help the designer to formalize his/her opinion the special tool, which helps the designer to sort alternatives according to his/her opinion about optimality, was developed. This tool is based on the analytic hierarchy process by Tomas L. Saaty. It is a multi-criteria method, which supports weighting of criteria, conflicting criteria and verification of correctness of expert's opinion.

There is a special extension of the module - service of distributed calculations. It allows using all computational powerful of a network for execution of series of numerical experiments that decreases time efforts correspondingly. This possibility is very easy and effective to use in computer centers and laboratories. Server of distributed calculations is based on using TCP/IP that allows employing for the needs of your project any computer not only in local or corporate network, but also in Internet.

6.3. Overview

Scanning of the design space is the most simple and reliable method to find the optimal solution or analyze dynamical behavior of the system. Scanning gives the researcher clear idea about response surface and global optimum. However practical application of scanning is limited up to 3-4 parameters due to dramatically increasing the number of numerical experiments:

$N = m^k$, where

k is the dimension of the problem,

m is the number of levels for each parameter.

Let us consider basic features of the scanning how it realized in UM. The researcher can unite several models within on scanning project, see Figure 6.2. Set of numerical experiments is generated automatically. For each model the researcher describes default values of parameters, initial conditions, problem-specific parameters (road vehicles, railway vehicles, and flexible bodies), finish conditions, parameters of numerical method and list of variables that will be saved for each numerical experiment.

List of calculated variables can include all kinds of dynamical performances: coordinates, velocities, accelerations, components and modules of applied forces and moments etc.[1] Read *<Sect. 4.3.3. List of variables>* for details.

Finish conditions are formulated as follows:

<Variable1><logical operation1><Value1> OR

<Variable2><logical operation2><Value2> …, where

<Variable i> is time or any UM variable.

After fulfillment of all numerical experiments their results are available for analyzing, see Figure 6.3. It is possible to get an oscillogram of any variable from the list of variables as well as summary graphs, surfaces and tables.

Let us consider usual work flow.

- Researcher adds all necessary models to the project and set all parameters for them.
- A list of numerical experiments is generated automatically.
- Scanning project starts and fulfills all numerical experiments, possibly with the help of Service of distributed calculations.

[1] Practical experience shows that the list of variables can contain up to several decades and even hundreds different variables.

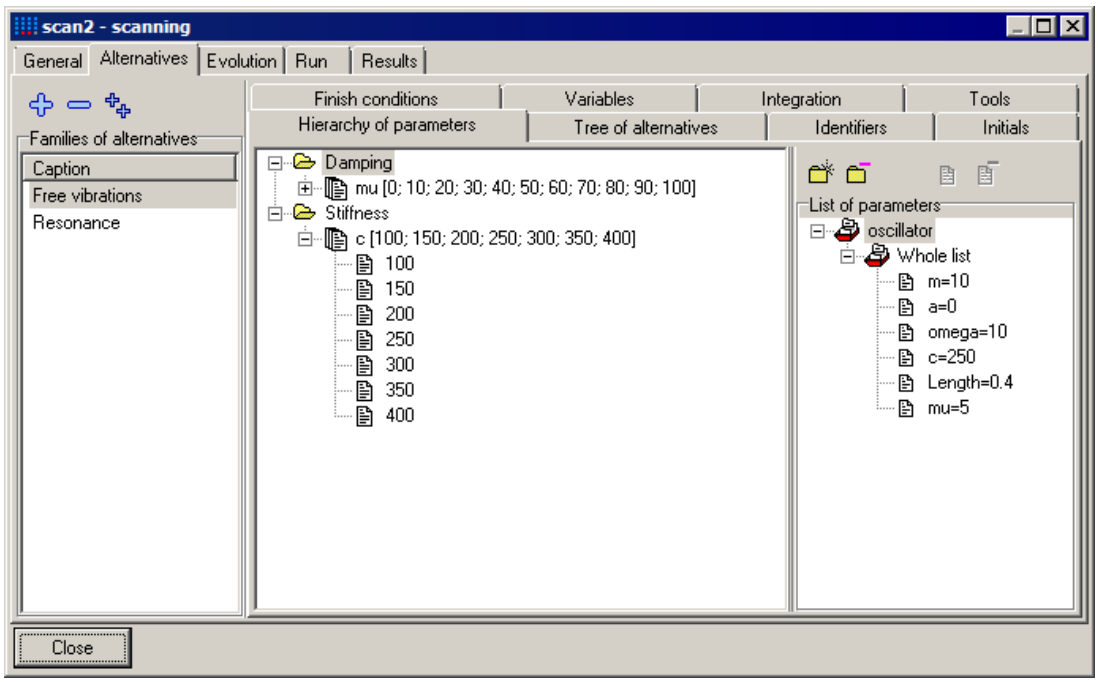

Figure 6.2. Scanning: hierarchy of parameters

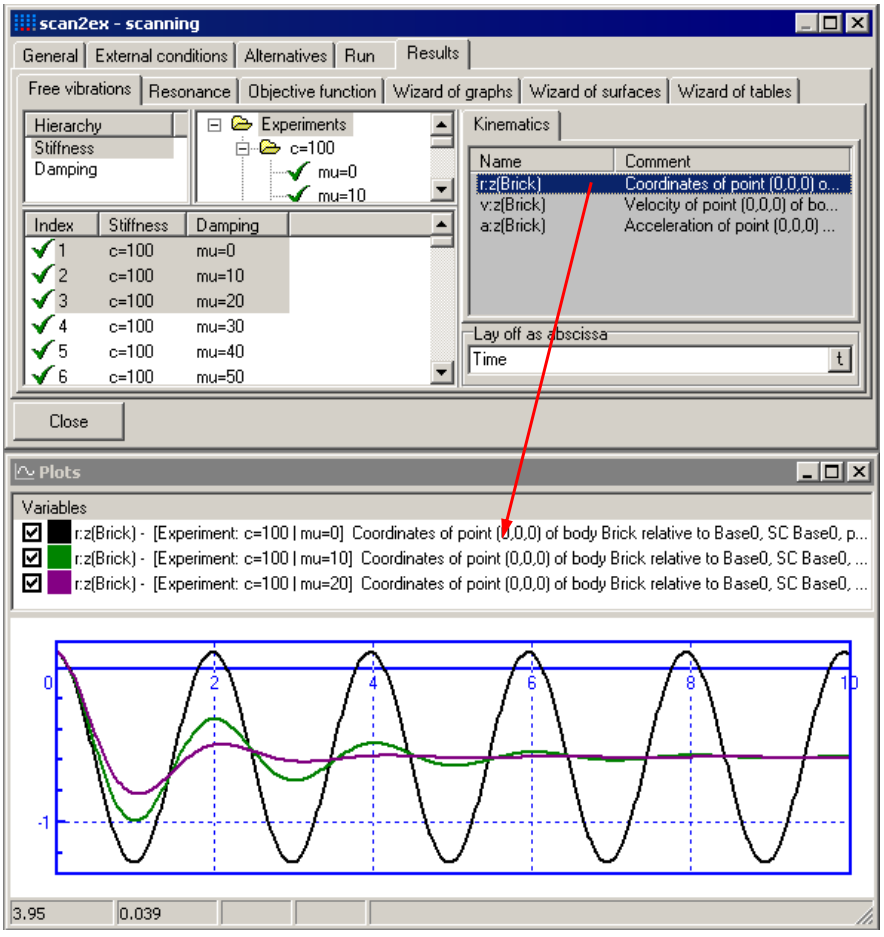

Figure 6.3. Scanning: processing results

6.4. Describing a new scanning project

Use the menu command **Advanced analysis | Scanning: new project…** to create a new scanning project. Every project is situated in a separate directory. Therefore it is necessary to point out the directory when you create new scanning project, see Figure 6.4.

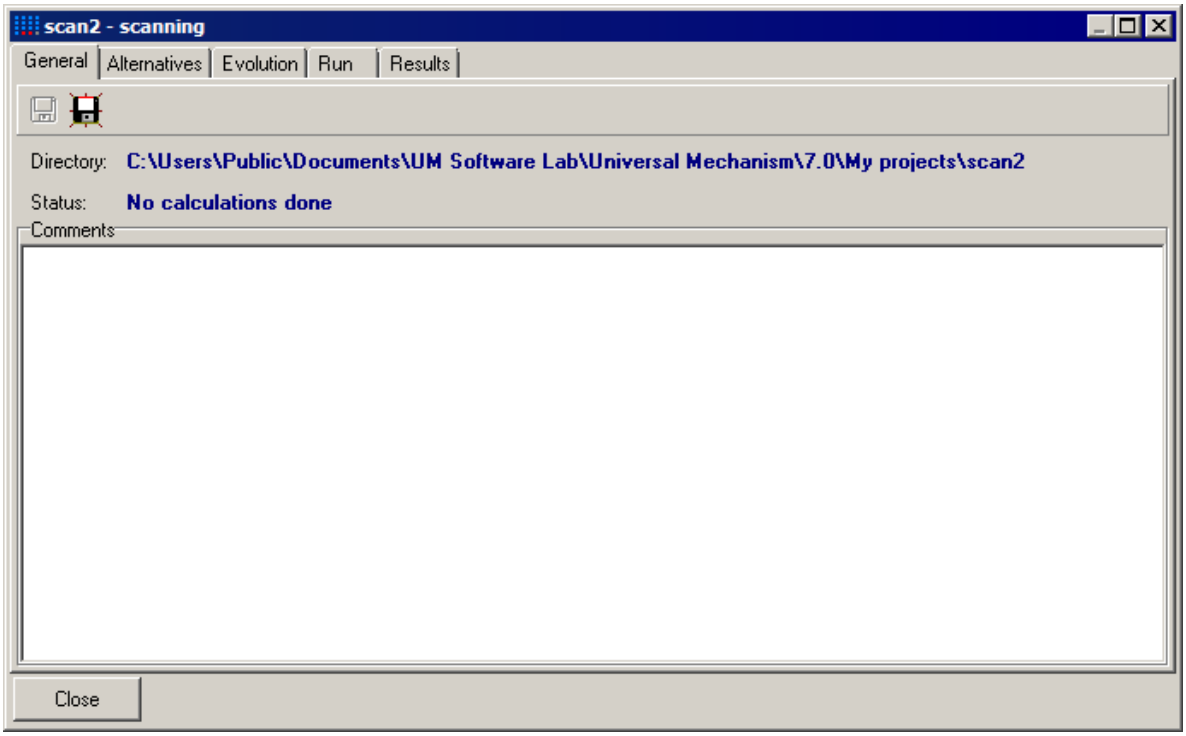

Figure 6.4. New scanning projects

6.5. Alternatives

A tab for description design parameters is shown in Figure 6.6. **Family of alternatives** is a group of alternatives generated by one model. Use and buttons or context menu (see Figure 6.5, Figure 6.6) to add/remove families. It is necessary to describe hierarchy of parameters for each family. Hierarchy of parameters can be compare to nested loops. It means that for each value of upper level all values of lower level will be tabulated. Within a level, the parameters are changed jointly; number of iterations is determined by number of values of the first parameter at the level. If the number of iterations in the first parameter is less than ones of other parameters then final extra values are ignored, else the values of other parameters are taken over again.

Possibly it might be more suitable not to add new *families* and set all their parameters but just copy families (the button) and alter them.

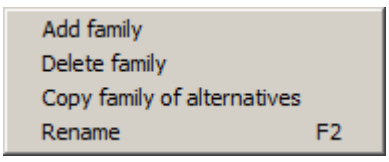

Figure 6.5. Family of alternatives: context menu

Setting considered in 6.5.1-6.5.7 are valid for active (current) family only. You should go through these steps for every family that you introduced in the project.

6.5.1. Hierarchy of parameters

Use and buttons to add/remove levels of hierarchy, and and buttons to add/remove parameters to/from current level. Double click onto any parameter from the list adds that parameter to current level of hierarchy. For example, the hierarchy in Figure 6.6. consists of two levels: **Damping** and **Stiffness**. **Stiffness** level consists of one parameter c having values [100, 200, 300, 400]. **Damping** level consists of one parameter **mu** with values [0, 10, 20, 30..100].

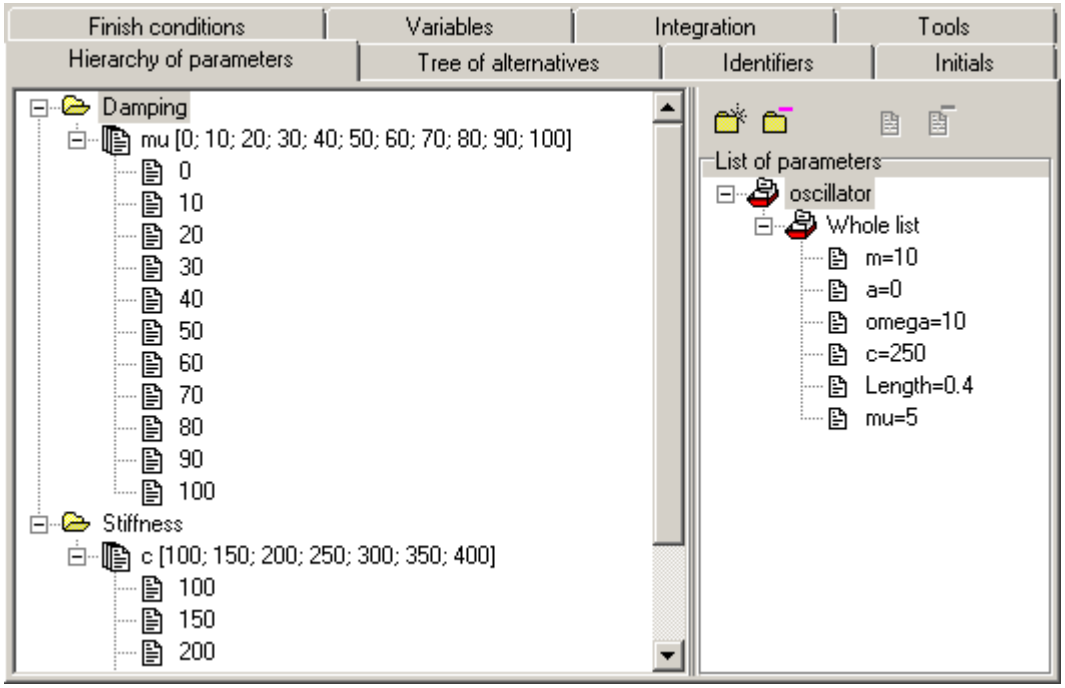

Figure 6.6. Hierarchy of parameters

6.5.2. Tree of alternatives

After description of hierarchy of parameters the full list (tree) of alternatives is generated automatically (see Figure 6.7). Tree of alternatives is on the top and list of alternatives is on the bottom.

6.5.3. Identifiers

Here you can find all of the system identifiers (see. Sect. 4.4.2.4). These values of identifiers will be used as default values for numerical experiments. The values are loaded before each numerical experiment. Of course, if an identifier changes as a design parameter then such an identifier will be redefined with the correspondent value.

6.5.4. Initial conditions

Here you can describe initial conditions each numerical experiment starts with (see Sect. 4.4.2.5).

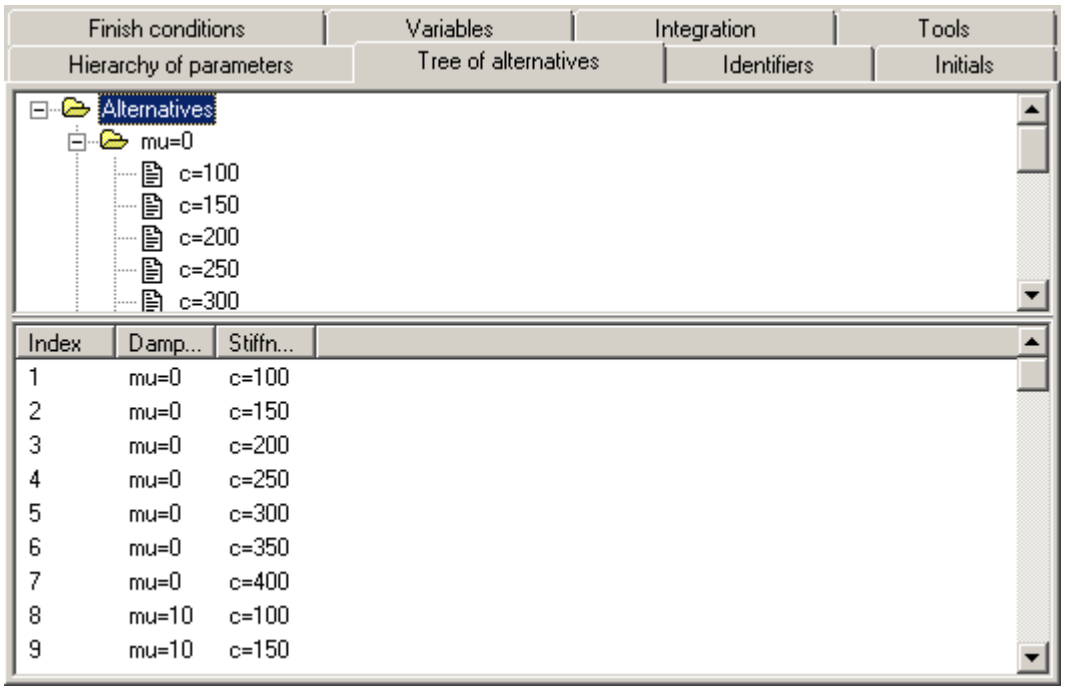

Figure 6.7. Tree of alternatives

6.5.5. Finish conditions

Here you can describe finish conditions for each numerical experiment in the current family, see Figure 6.8, Figure 6.9. Finish conditions are formulated in the following way: “Interrupt a numerical experiment if finish conditions are satisfied”. For example, the default finish conditions in Figure 6.8 means “Interrupt a numerical experiment if simulation time is more than 10 seconds”. Simulation time in this sense is the model time, not the real time of the user.

You can use time and any other variables as criteria for finish conditions. Create the necessary variable in the **Wizard of variables** (see Sect. 4.3.2. Wizard of variables) and drag it to the correspondent field, see Figure 6.9. Use ··· button to set time as a criterion again.

Buttons / are used to brace two conditions in brackets to change the priority of calculation. Example, that is given in Figure 6.9, means *Interrupt any numerical experiment, if*

(Time > 1.27 AND a:z(Brick) > 1) OR v:z(Brick) < 2.

The first button braces the first and the second finish conditions; the second button braces the second and the third finish conditions. It has no sense to turn on two buttons simultaneously – it makes no difference in calculation of finish conditions.

Please note that the logical operator *AND* has higher priority than the *OR* operator.

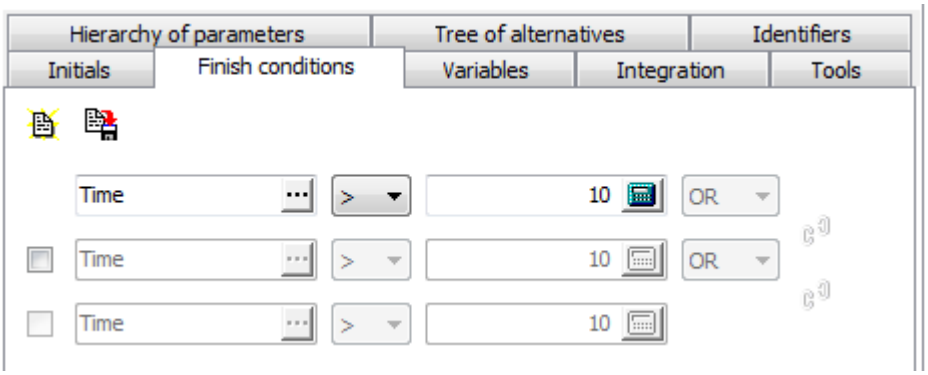

Figure 6.8. Default finish conditions

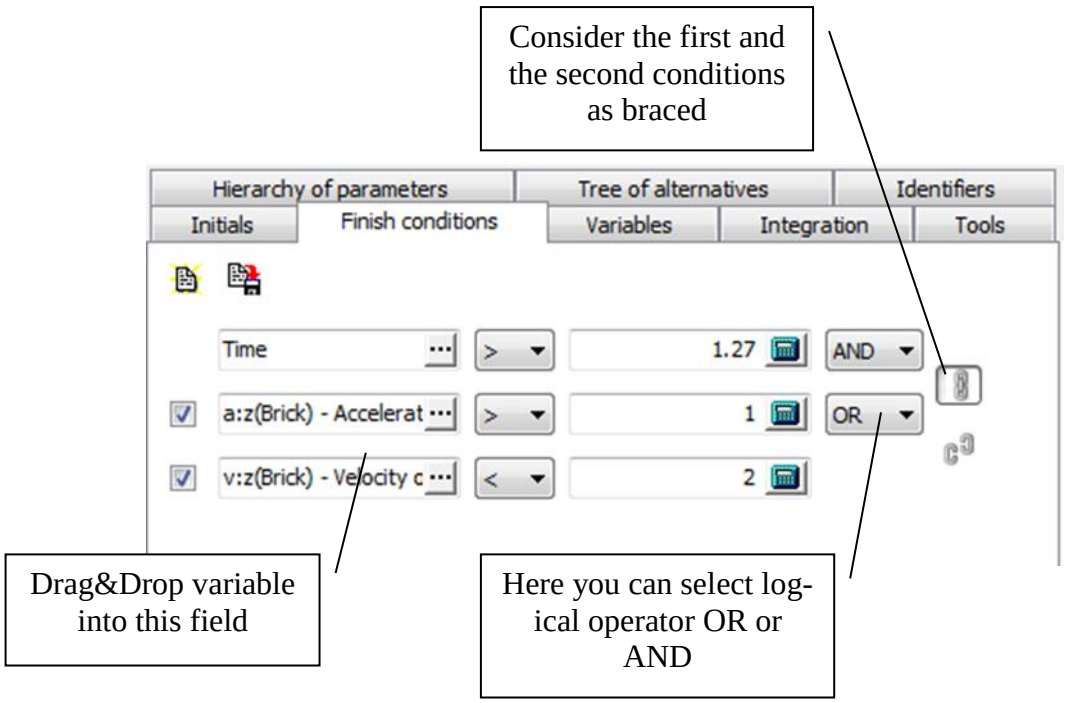

Figure 6.9. User-defined finish conditions

6.5.6. Variables

Here you should form a *list of variables* (see Sect. 4.3.3), which will be stored for every numerical experiment. Variables from this list will be available as results of *scanning*.

6.5.7. Integrator

Here you should choose integration method and set its parameters. This setting will be used for numerical simulation (see Sect. 4.4.2.1).

6.6. Running

When you activate the **Run** tab (see Figure 6.10), checking the description of scanning project is performed. You can see its results in the **Report** box. If some errors are found, you should fix them and activate **Run** tab again.

When scanning project starts, the following information for each numerical experiment is available: time information, values of identifiers of hierarchy of parameters, summary information. A progress bar shows the ratio between number of done experiments and total number of experiments. Use the **Stop** button to interrupt execution. If any software or hardware breakdown appears, the results of the scanning project are not lost and execution will go on from the last unfulfilled experiment.

Let us consider an example, see Figure 6.10. This project includes 27 numerical experiments and 4 of them are already done.

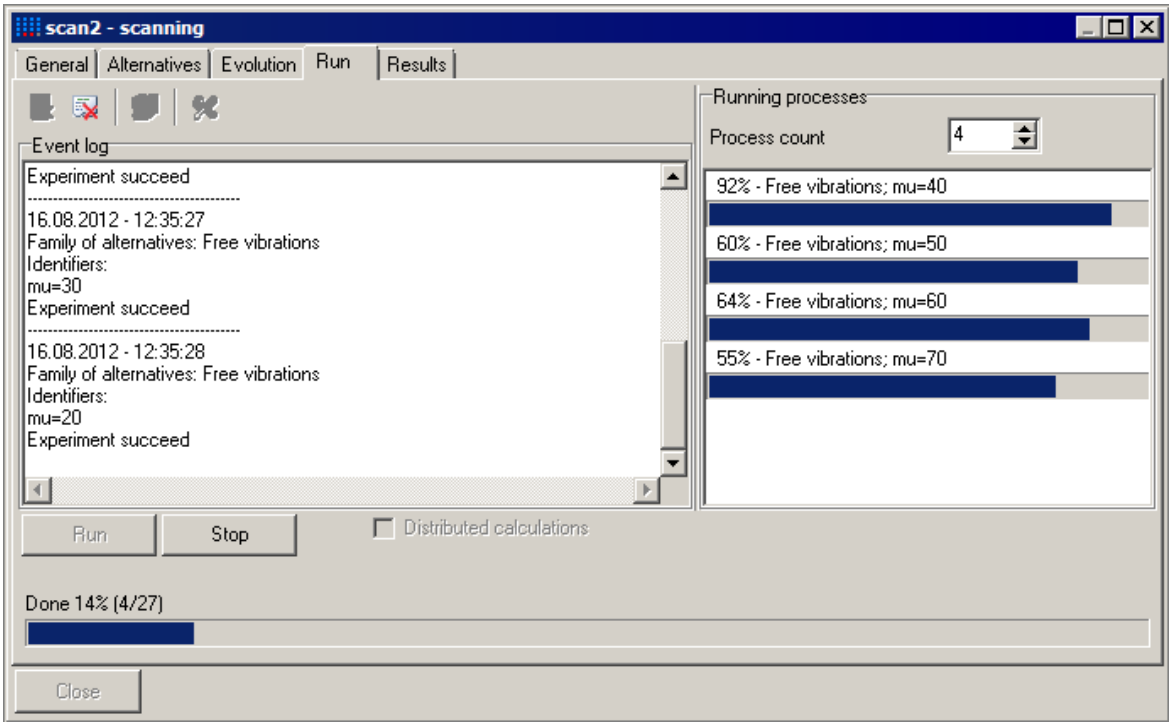

Figure 6.10. Running the project

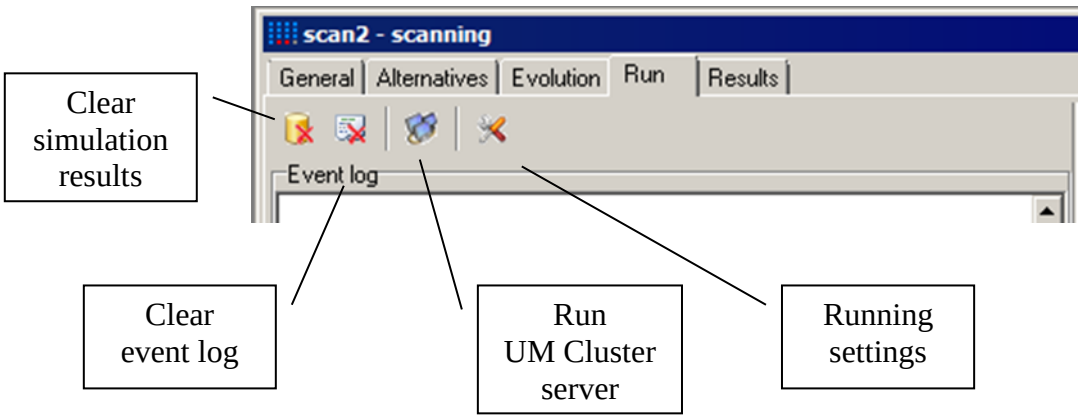

Figure 6.11. Toolbar

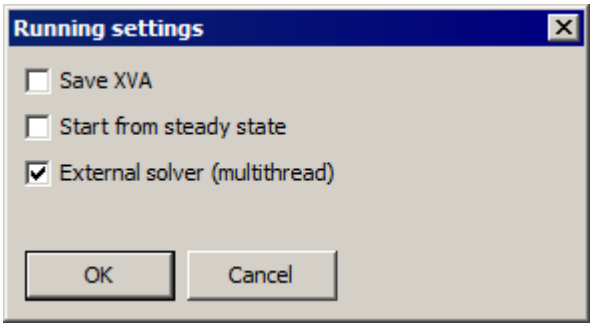

Figure 6.12. Running settings dialog

6.7. Results

If at least one numerical experiment is done then its results are available in the **Results** tab (see Figure 6.13). For each family of alternatives, there is the tab of the same name. There is a list of calculated variables on the right. On the left there is the list of numerical experiments. Fulfilled experiments are marked with the ✓ sign, and unfulfilled ones are marked with the ⊘ sign.

Select necessary experiments (use the **Shift** and the **Ctrl** keys in a standard way) from the list of experiments, then select necessary variables from the list and perform drag them to a graphical window. Variables in the graphical window have a correspondent comment, see Figure 6.13.

The tree of experiments corresponds to the list of experiments. Selection of a branch in the tree leads to selection the corresponding items in the list. Use the context menu of the hierarchy to change position of its levels and, as a result, fast selection of necessary experiments.

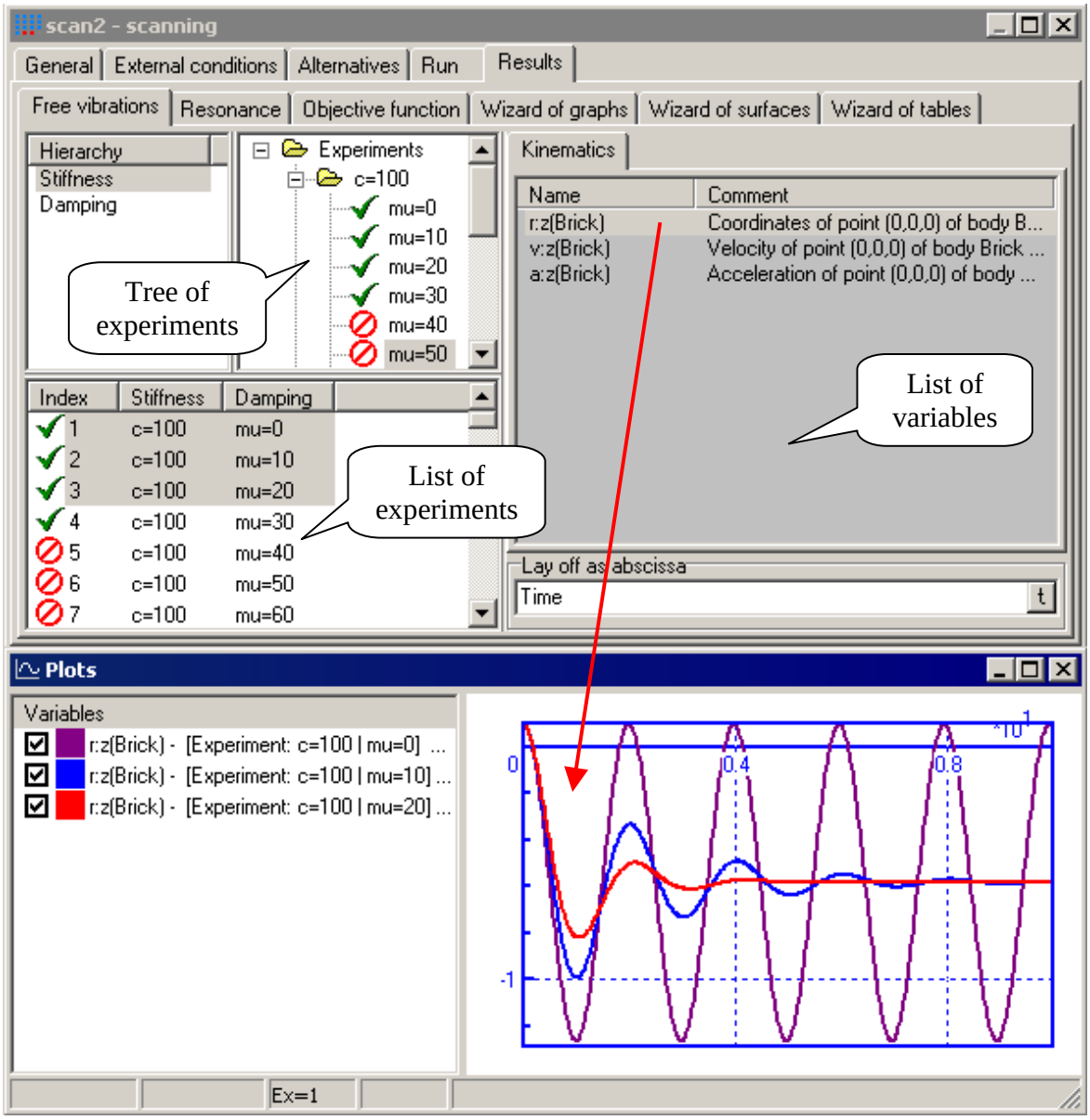

Figure 6.13. Processing results

6.7.1. Wizard of graphs

The *wizard of graphs* (see Figure 6.14) is intended for forming dependence between a functional of any calculated variable on any parameter, when other parameters are fixed. Comments in the right bottom corner of the window will help you.

To add a summary graph you should go through the following steps.

- Choose a family, a variable to be analyzed, functional and parameter for computing a dependence, see Figure 6.14.
- Optionally you can use limit for abscissa that can be used for excluding transient processes, as well as scale factors/units and interval for abscissa.
- Click the button to plot the summary graph.
- To add new plot change the **Other parameters** if necessary and double click on the same or new variable in the **Criteria and variables** tree.

Note. **Other parameters** list is enabled if only scanning project has more than one level.

Let us consider an example, see Figure 6.14. Here we can see the root mean square of the vertical position of the 1 d.o.f. oscillation system in dependence on *omega* parameter. In this case of forced oscillation *omega* means the frequency of the driving force. The resonance case is clearly shown.

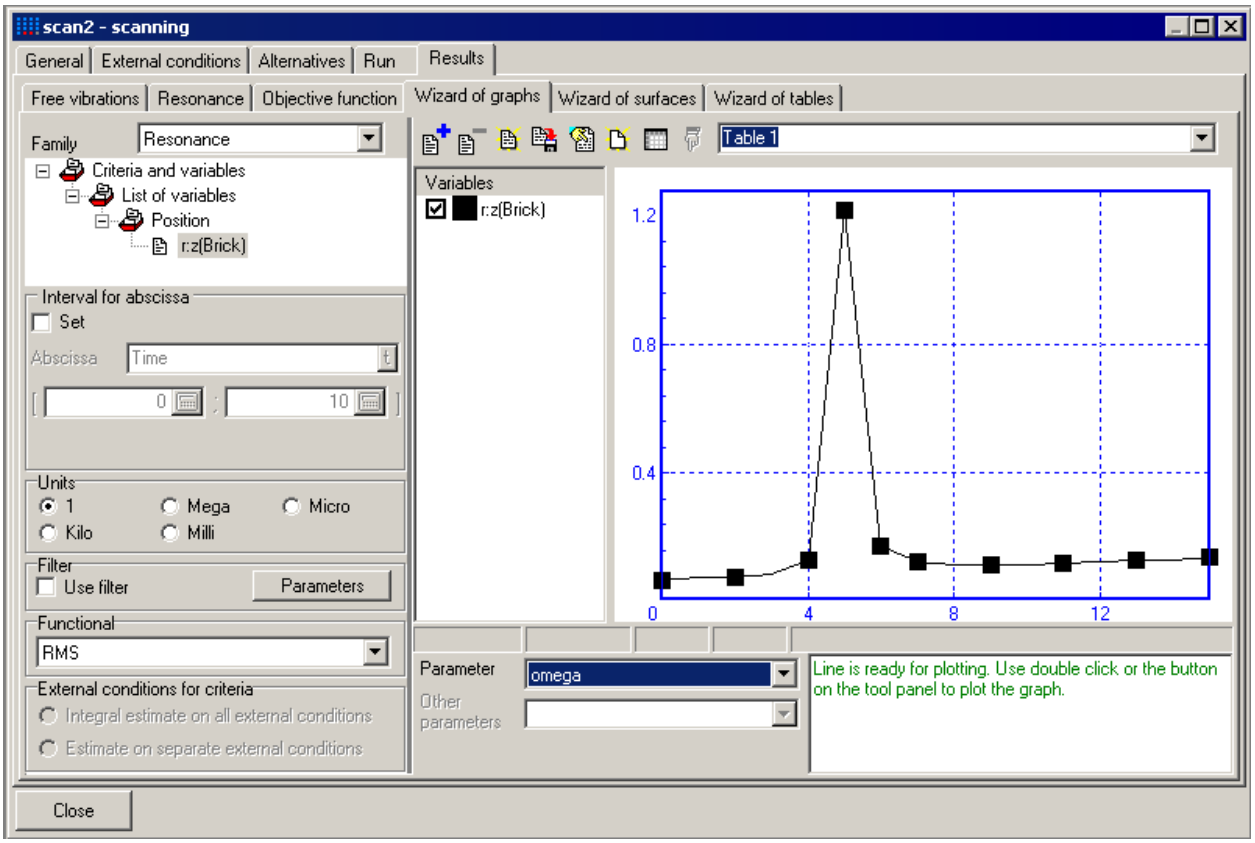

Figure 6.14. Wizard of graphs

You can switch between graphical and table mode with the help of / button. In the table mode you can drag any column of the table to the graphical window. Drag the first column to plot all graphs.

Note.

- Choosing a new family of alternatives or a new parameter clears the table/graphs.
- Changing variables in the **Criteria and variables** tree keeps the table contents and can be used to obtain a group of dependences.

You can determine settings for **Abscissa limit, Units** and **Filter** during describing the scanning project in the **Variables** tab. To show these processing options use the button. In the presence of such processing options they will be used automatically.

It is usually more suitable to use different tables for summary graphs of variables with different dimensions. Use , buttons to add/delete tables. To save/read template (configuration) of the wizard of graphs use and buttons.

6.7.2. Wizard of surfaces

Wizard of surfaces is intended for creating the response surface as functional of any calculated variable in dependence on two any design parameters keeping other parameters constant. Working with **Wizard of surfaces** make sense if scanning has at least two design parameters. **Wizard of surfaces** is quite similar to **Wizard of graphs**. An example of using the **Wizard of surfaces** is given in Figure 6.15. Corresponding surface in *Microsoft Excel* is given in Figure 6.16.

Working with the **Wizard of surfaces** use the following work flow.

- Select the family, calculated variable or criterion and necessary functional, see Figure 6.15.
- Then choose two parameters for the surface and fix values of **other parameters** if any.
- Set **Abscissa limit**, **Units** and **Filter** parameters if necessary.
- You can export prepared surface to text file or clipboard as text table and to the *Microsoft Excel* as a diagram.

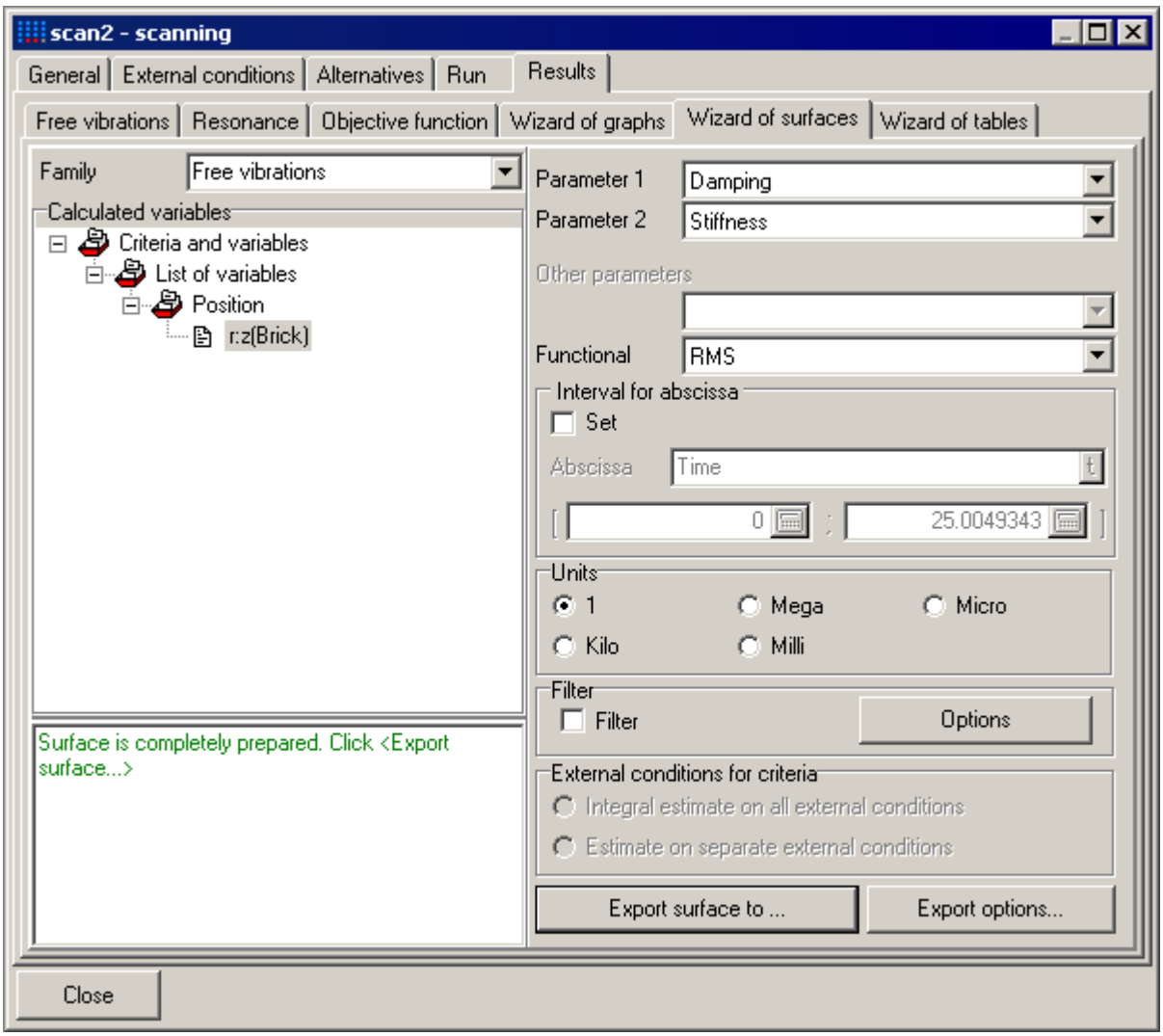

Figure 6.15. Wizard of surfaces

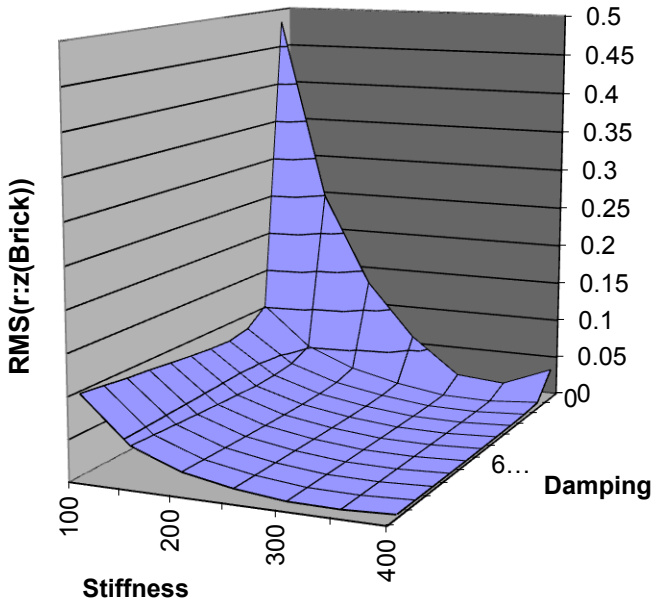

Figure 6.16. Exported surface

6.8. Methodical remarks

Project structure

The root directory of the scanning project includes subdirectories with models that are considered in this project. Once a model is added to the scanning project its source files are copied to the project directory. Then all numerical experiments are fulfilled for this copy. All changes in original model are ignored.

Portability

To copy/move the project it is necessary to copy/move its root directory. Such copying does not break the integrity of the project, you will be able to start/continue numerical experiments and analyze results.

Save as

Carrying out applied researches it is often necessary to fulfill the same scanning project with small differences. The simplest way to save the project to another directory ( button), modify the project and start its fulfillment.

Data files

After fulfillment every numerical experiment several data files are saved. Time history for every variable from the *list of variables* is saved, see Sect. 6.5.6. The more variables in the *list of variables* and the more simulation time the more size of data files. Practical experience shows that data files can reach *Gigabytes*. So if you have no enough free disk space you have to find compromise between the number of variables in the *list of variables*, simulation time and size of data files of the scanning project.

Deleting data files

To delete all data files use  button on the **General** tab. You also can delete data files for some numerical experiments to recalculate them. Select these numerical experiments in the list of experiments (see Figure 6.13) on the **Results** tab and select **Clear results** from a context menu.

Project modification

You can modify the project at any stage of its fulfillment. For example, after fulfillment the project you can add a new model to a project. In this case to finish the project it is necessary to fulfill numerical experiments for the new model only.

You can also add some new levels for parameters for any model (family of alternatives). All already calculated experiments will keep their results. It gives a possibility to add new points in design space and thus specify dynamical behavior of the models step by step if necessary.

Also please note that all numerical experiments for every family are fulfilled with the same initial conditions, identifiers, parameters of numerical method, railway, road and other parameters. If any changes in any family settings are found the program shows dialog and asks the user saves these changes with or without clearing data files. Normally you are recommended to clear data files and fulfill all numerical experiments again.