Material & Structure Analysis Suite

Simulation-Optimization GUI Handbook

Version 9.1

Z-set software is distributed by

Transvalor S.A.
Centre d'affaires La Boursidière
rue de le Boursidière
Bâtiment Q, 1er étage
92350 Le Plessis-Robinson
France

http://www.zset-software.com
support@zset-software.com

Contents

Chapter 1

Generalities

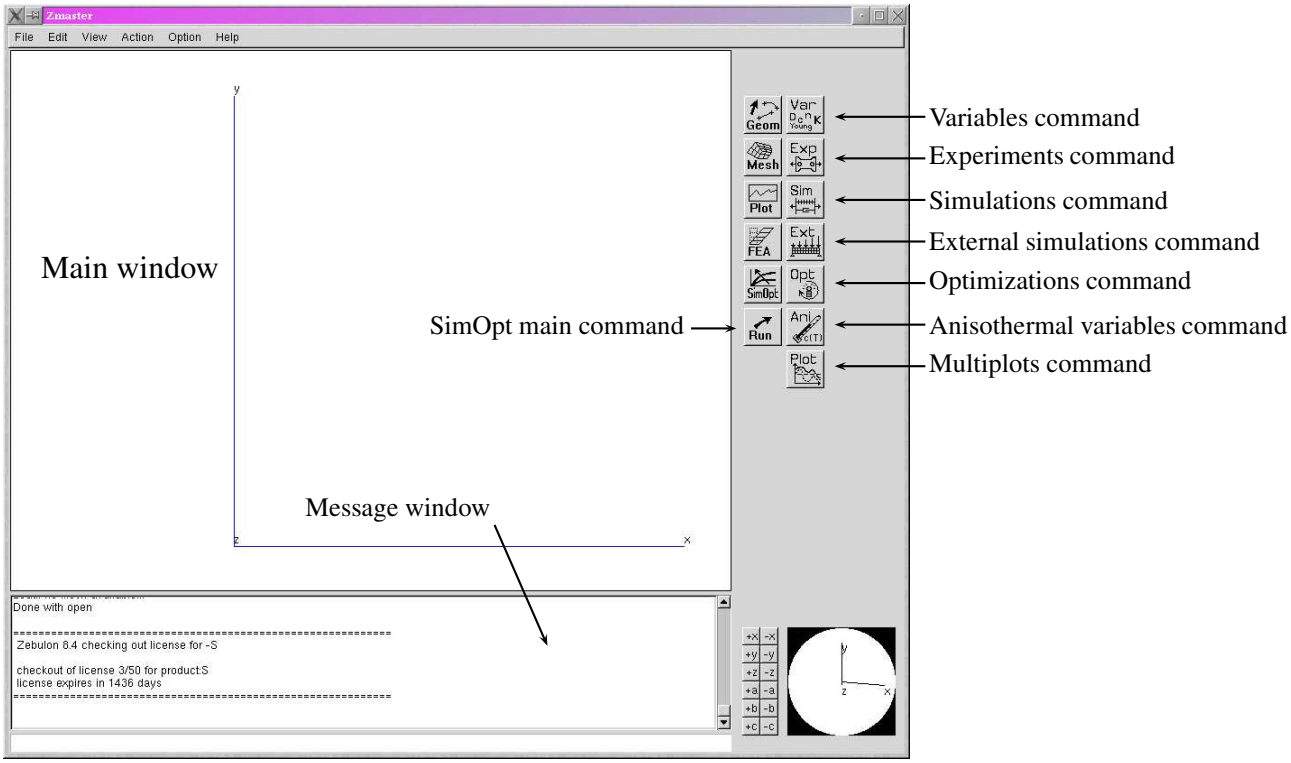
Introduction

Starting with version 8.3, a graphics user interface (GUI) is available as a Zmaster plugin to drive the Simulation (Zsim) and Optimization (Zopt) modules.

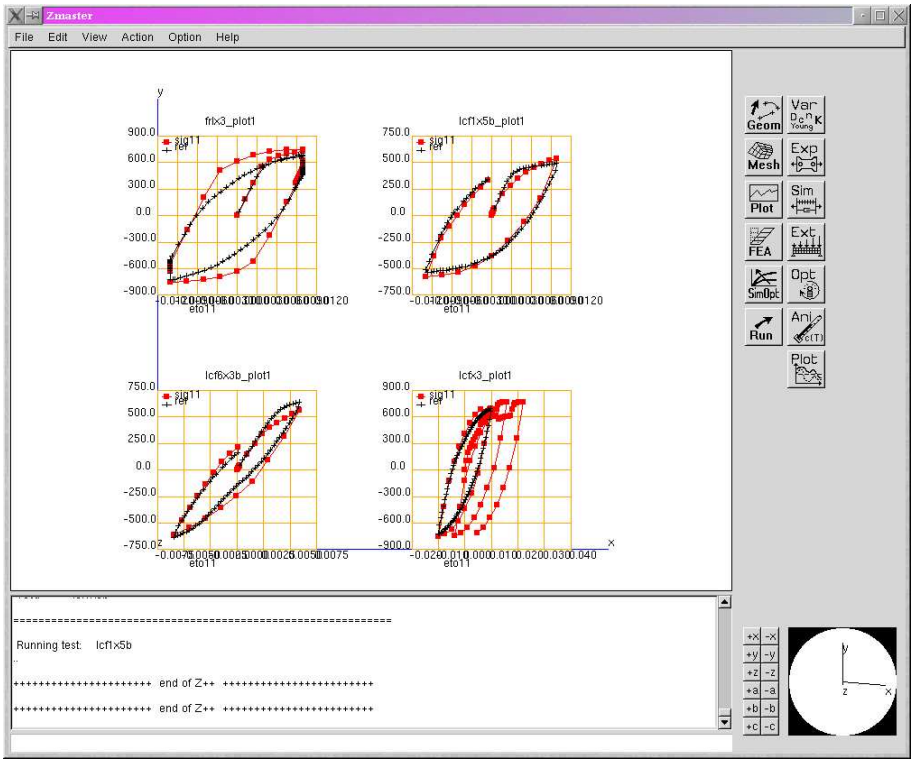
The interactive capabilities offered by such an interface should be a great help for material coefficients identification.

As the Sim-Opt Plugin has been included in the distribution, a new command button is available in the toolbar to the right of the Zmaster window. As shown in the figure below, a click on the Sim-Opt button has the effect to fill the sub-command area with new commands associated with the seven main parts of the application :

- Variables management sub-commands
- Experimental results management sub-commands
- Simulation items management sub-commands
- External simulation items management sub-commands
- Optimization items management sub-commands
- Anisothermal variables management sub-commands
- Multiplots management sub-commands



The graphics window is mainly used by the application to plot the curves attached to the `experiment`, `simulation`, `external simulation` or `optimization items` selected in the corresponding dialogs. The curves are tiled in the X-Y plane, and the `ctrl-LB` and `ctrl-MB` mouse buttons can be used to move or zoom the image to get a better visualization of a particular curve.



Template files

The Sim-Opt GUI makes an extensive use of the template files mechanism introduced with the optimization module (Zopt). Templates (file extension .tmpl) are ASCII files where numerical values are replaced by a character string of the form ?*var*, where *var* is the name of an optimization `variable`.

The optimization module makes use of those template files to update the simulation programs input files with the current value of the design `variables` before re-launching the simulations : from a given template file named `input.tmpl`, a file named `input` (obtained by removing the `.tmpl` extension from the template name) is generated where numerical values are substituted to ?*var* definitions.

Typically, template files will correspond to Zmat material files where material coefficients values are replaced by `variables` definitions.

Example:
Included hereafter is a template file named `chaboche.tmpl` with design `variables` definition for all material coefficients associated to the viscoplastic part of the model.

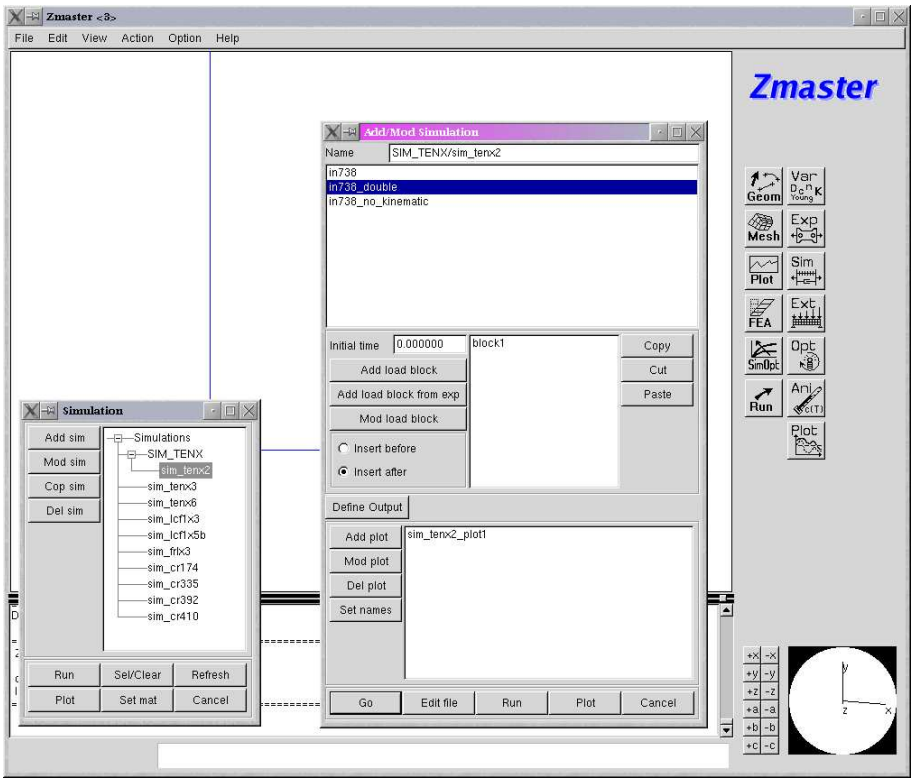
```
***behavior gen_evp
 **elasticity
   young 200000.
   poisson 0.3
 **potential gen_evp ev
  *criterion mises
  *flow norton
   n  ?n
   K  ?K
  *kinematic nonlinear
   C  ?C
   D  ?D
  *isotropic constant
   R0 ?R0
***return
```

Selection trees

Selection trees allow to list items, and offer additional functionalities. Selection trees present information as a hierarchy, with items that may be expanded to show further items. Selection trees are made of two types of structure : directories and files. Directories are considered as nodes that may in turn contain sub-directories or files, while files are considered as leaves. Sub-directories may be opened out or closed (hidden).

On the one hand, in a selection tree, the user may select an item or a set of items or a directory to apply appropriate commands on. If a directory is selected, the appropriate commands will be applied on all the files contained by itself and by its sub-directories and their own sub-directories ...

On the other hand, when the main dialog contains a `Name` textfield, the user can create or suppress directories and move files. The next figure shows an example of such a dialog.



For instance, from a `Name` textfield filled with `directory_A/file_A`, if the user write `directory_B/file_B`, hitting the `Go` command will move the file named `file_A` and located in the `directory_A` directory to the file named `file_B` and located in the `directory_B` directory. The `Name` textfield allows then to create directories in the files system.

When moving a file lets an empty directory, the empty directory is automatically removed from the files system. Note that when a file is moved, all the execution files defined with the

same name but different extensions will be moved too.

Finally, the main way the user should use selection trees may be to organize the different directories containing the numerous files created by the SimOpt GUI.

Function items

`Function items` are new `Simulation items` that allow to launch simulations based on functions implying the defined variables (material coefficients for instance) and/or new parameters then automatically included in the database as variables (as described at page 2.6).

Chapter 2

Variables management

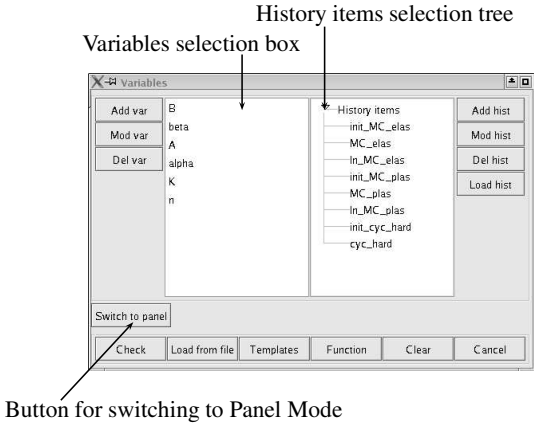
Variables

The `Variables command` allows to create, delete and modify optimization `variables` and `history items`. The `history items` can be created to store the current state of the `variables` values, and used to regenerate the database at any point with the previously saved values.

Two modes of operation are available, the `Creation Mode` and the `Panel Mode`. A button at the bottom on the left of each dialog allows to switch from one mode to the other. When the `Variables command` is hit for the first time, the application opens the `Variables dialog` in `Creation Mode`. If the dialog is closed, a new click on the `Variables command` will open the dialog in the last mode selected.

Creation mode dialog

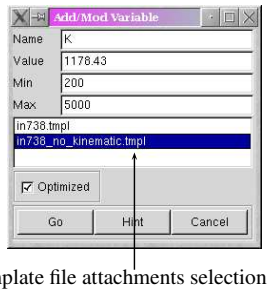
The dialog is represented hereafter. `Variables` previously defined are inserted into an appropriate selection box, while `history items` are integrated into a selection tree.



Various sub-commands can be activated from this dialog, as described in the next pages.

Add and Mod var commands:

Both commands pop up the `Add/Mod var dialog`. A `variable` must be selected in the `Variables selection box` before clicking on the `Mod var command`.



The `variables` attributes (name, current and min/max values) can be modified in the corresponding textfields. The `Optimized` checkbox can be used to activate/deactivate optimization of the `variable`, without changing anything to the `optimization items` defined (as described at page 6.2) in the application.

The `Template file selection box` (see page 1.4) can be used to attach the `variable` to an arbitrary number of template files ; note that a `variable` can be attached to several template files.

Modifications are assigned to the `variable` database only after the `Go` button is hit. An automatic check is performed to verify that the `variable` is actually defined in the template files to which it is attached. An error message is printed otherwise, and the update of the `variable` database is canceled.

The Hint button can be used to initialize the current and min/max values by reading the realistic corresponding values in the typical_coef_values file located in the $Z7PATH/lib/materials directory.

The Cancel button closes the dialog without modification.

Del var command:

All variables selected in the Variables selection box (page 2.3) are deleted from the variables database.

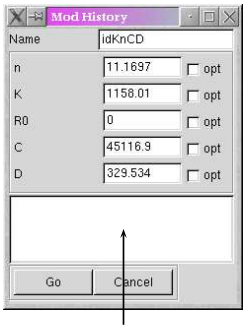
Add hist command:

This command is used to save the current state of the variables database into an history item. The Add/Mod history dialog is opened that includes text fields containing the current values of all design variables defined in the database. The history item is given a default name that should be changed in the corresponding text field to something more meaningful to the user. A text area allows user's comments. The history database is only updated after the Go button is clicked. The Cancel button closes the dialog without modification.

Note that the variables database can be re-initialized at any time with the values stored into an history item, by selecting the corresponding item in the History items selection tree, and clicking on the Load history button of the Variables Creation Mode (page 2.3) and Variables Panel Mode dialogs (page 2.8).

Mod hist command:

An history item must be selected before in the History items selection tree (page 2.3). A window pops up allowing to change the name, or the values of variables stored in this history item. A text area allows user's comments. The history database is only updated after the Go button is clicked. The Cancel button closes the dialog without modification.



Del hist command:

An history item must be selected before in the History items selection tree (page 2.3). The selected item is removed from the database.

Load hist command:

This command re-initializes the variables current values with the ones stored in the history item selected in the History items selection tree.

Check command:

This command verifies the coherency of the template file attachments assigned to the optimization `variables`. In particular the check covers the following points :

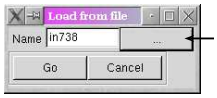
- The application verifies that each `variable` is indeed defined in the template file to which it is attached. This check is also performed each time a `variable` is updated by the `Add/Mod var commands`, but if the template file definition is changed afterwards the attachment may no longer be valid.
- The application checks that all `variable` definitions in active template files (ie. those attached to at least one `variable`) are resolved by an attachment to a `variable`.

Load file command:

This command can be used to load a `variables` database from :

- An input template file. In this case, all `variables` defined in the template file (with the *?var* syntax, see page 1.4) are automatically created in the `variables` database with the proper attachment. `Variable` value and min/max bounds are set to the ones found in the file `$Z7PATH/lib/materials/typical_coef_values`. Afterwards, those defaults should be manually changed by the user.
- An input material file. In this case, for each coefficient in the material file (ie. a character string followed by a floating point numerical value), a `variable` is created in the database and the numerical value read in the file is assigned. An appropriate template file, whose name is obtained by adding a *.tmpl* extension to the input material file name, is also automatically created in the current working directory and attached to all created variables. Care must be taken by the user that this mechanism does not erase a pre-existing file, since no check is attempted in the current version. `Variables` min/max bounds are set to the ones found in the file `$Z7PATH/lib/materials/typical_coef_values`.

A new dialog is opened by this command, where the user must define the name of the input file. The browse button can be used to look for an available file in the directory architecture.



Templates command:

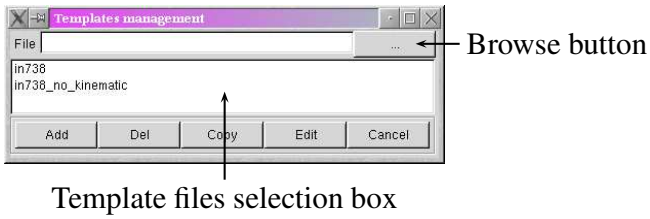
This command allows some management of the template files attached to `variables`. The `Templates management dialog` includes a selection box with all the active template files, ie. files with a *.tmpl* extension in the current working directory.

The `Templates management dialog` sub-commands are summarized hereafter :

- `Add command` : this command activates a template file whose name is specified in the `File` textfield. The browse button can be used to select a file in a directory different from the current working directory. In this case, the selected file is copied to the working

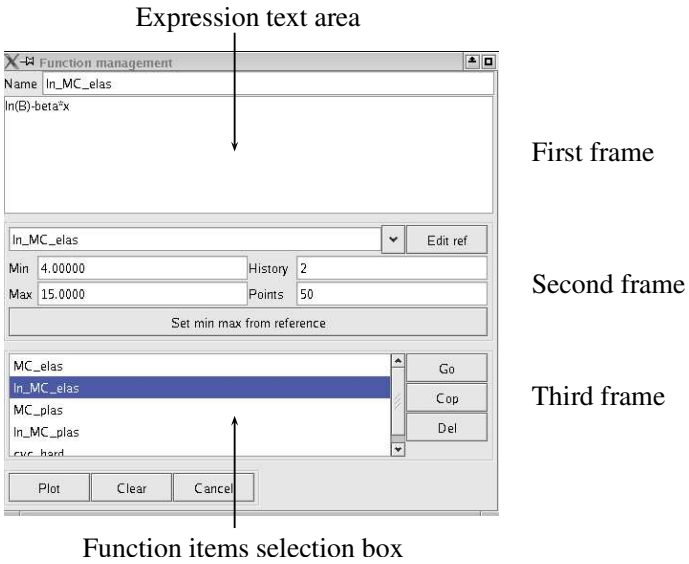
directory. If a file whose name is specified in the `File` textfield does not exist in the current working directory, a new empty file is created. Then the `Edit command` can be used to modify the file content.

- `Del command` : this command removes the template file selected in the selection box from the current working directory.
- `Copy command`: this command copies the template file selected in the selection box to a file whose name is specified in the `File` textfield.
- `Edit command` : this command edits the file selected in the selection box.
- `Cancel command` : this command closes the `Templates management dialog`.



Function command:

This command may be used to create or modify `function items` in the database by opening the `Function management dialog`. A window pops up allowing to define the function parameters. The dialog is shown in the next figure.



Function name may be defined in the `Name` textfield, and function expression may be written in the function expression text area.

The second frame allows to define the plot parameters of `function items`. A combo box filled with the `experiment items` defined beforehand by means of the `Experiments command` (page 3.2), where a particular item can be selected to be used as a reference curve drawn on

the same plot as the function curve. The `Edit ref` button edits the experimental data corresponding to the `experiment item` selected in the combo box. The `Min` and `Max` textfields allow to define the min and max values of the function abscissa. The `History` textfield corresponds to the number of more recent function simulations that will be drawn on the same plot (default value is 2). This type of plots is a convenient way to visualize for instance the influence of coefficients on the simulated response. The `Points` textfield defines the number of points the function curve will be constituted of (default value is 20). The `Set min max from reference` button allows to use the min and max values of the experimental data to set the min and max values of the function abscissa.

The third frame includes functions handling. The selection box on the left lists `function items`, on which the sub-commands on the right can be activated. The `Go` button updates the database with the definition of the `function item` currently edited, by creating or modifying it. Note that the `Go` command also updates and checks the definition (attachments for example) of the variables implied by the function expression, or create in the database the new variables implied by the function expression. Using the `Cop` button, the user can create a new item by copying the selected `function item` (without suppressing it). The copied item then appears at the end of the selection box. The `Del` button deletes from the database the selected `function item` and checks the definition of the variables implied by the function expression.

The `Plot` button draws the selected `function item` and its reference experimental curve. Plot is tiled in the graphics area across the X-Y plane. The `Clear` button clears the current `function item` selected in the selection box. The `Cancel` button closes the `Function management dialog` without further modification to the database.

Clear command:

This command clears at the same time selections in `Variables selection box` and in `History items selection tree` (see page 2.3).

Go to panel command:

This command closes the `Variables Creation mode dialog` and opens the `Variables panel dialog` (see page 2.8).

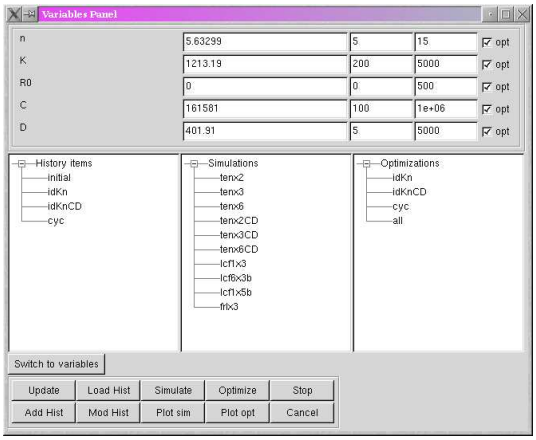
Cancel command:

This command closes the `Variables Creation mode dialog` without further modification to the database.

Variables panel dialog

Once a valid `variables` database has been created in `Creation mode`, the `Panel mode` is a much more convenient way to update the `variables` values, and can also be used to launch simulations and optimizations.

As represented hereafter, the dialog includes textfields for all `variables` available in the database. Those can be used to change the current and min/max values. A check box is also included to activate/deactivate the treatment of the `variable` during optimizations.



Switch to variables command:

This command closes the `Variables panel dialog` and opens the `Variables Creation mode dialog` (see page 2.3).

Update command:

The user can use this command to update at the same time `variables`, and `history`, `simulation` and `optimization items selection tree` from the `Variables Panel dialog`.

Add hist command:

This command is used to save the current state of the `variables` database into `an history item`. The `Add/Mod history dialog` is opened that includes text fields containing the current values of all design `variables` defined in the database. The `history item` is given a default name that should be changed in the corresponding text field to something more meaningful to the user. For each `variable`, a check box is also included to activate/deactivate the treatment of the `variable` during optimizations. A text area allows user's comments. The history database is only updated after the `Go` button is clicked. The `Cancel` button closes the dialog without modification (see page 2.4).

Load hist command:

This command re-initializes the `variables` current values with the ones stored in the `history item` selected in the `History items selection tree` (see page 2.4).

Mod hist command:

An `history item` must be selected before in the `History items selection tree` (page 2.3). A window pops up allowing to change the name, or the values of `variables` stored in this `history item`. For each `variable`, a check box is also included to activate/deactivate the treatment of the `variable` during optimizations. A text area allows user's comments. The history database is only updated after the `Go` button is clicked. The `Cancel` button closes the dialog without modification (see page 2.4).

Simulate command:

Clicking on this button launches all simulations selected in the `Simulation items selection tree` of the `Variables Panel Mode dialog` (page 2.8).

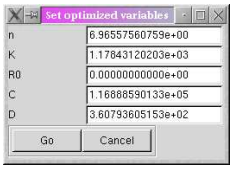
Plot sim command:

This command plots all curves selected in the `Simulation items selection tree` of the `Variables Panel Mode dialog` (page 2.8).

Optimize command:

Clicking on this button launches the optimization selected in the `Optimization items selection tree` of the `Variables Panel Mode dialog` (page 2.8).

At the end of the optimization, a window is popup showing the values found by the optimizer, and the `Go` button must be clicked if the user wants to update the database with the optimized values. Otherwise, `variables` are left unchanged.



Plot opt command:

This command plots all comparisons defined in the `optimization item` selected in the `Optimization items selection tree` of the `Variables Panel Mode dialog` (page 2.8).

Stop command:

This command can be used to stop a running optimization. The above `Set optimized variables dialog` is popup, showing the best values found by the optimizer before optimization was canceled. As in the previous case, the update of the database with the optimized values is only done if the `Go` button is clicked.

Cancel command:

This command closes the `Variables Panel dialog` without further modification to the database.

Chapter 3

Experimental results management

Experiments

The `Experiments command` allows some management of experimental data in order to put them in a form suitable for comparisons with simulation results prior to optimization. In the application, experimental data are manipulated through three different types of `Experiment items` :

- `base experiment items` are X-Y curves extracted directly from the raw experimental data by selecting columns in input ASCII files ;
- `derived experiment items` can be generated from the previous ones (or from other `derived experiment items`) by applying various transformation operations ;
- `composite experiment items` can be defined to regenerate multi-columns data from the results obtained after transformation.

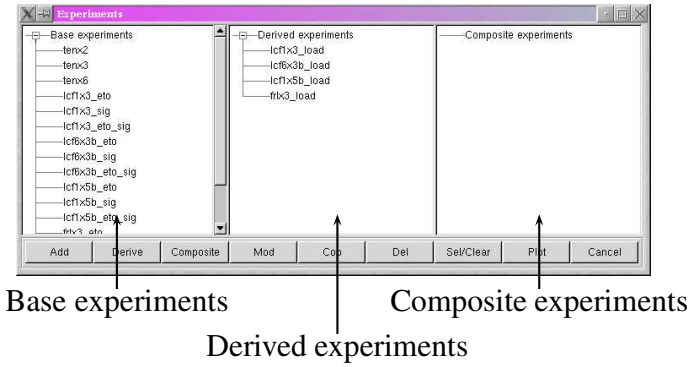
These `experiment items` are the basis of comparisons with simulation results in the definition of the optimization procedures, and can also be used to define the loading of simulations. The transformation operations supported in the definition of the `derived experiment items` are quite comprehensive, including :

- selection of regions of the input curve ;
- simple transformations of the X-Y coordinates using mathematical functions ;
- complex manipulations by means of a Zprogram script ;
- powerful data smoothing (LOESS algorithm).

Finally, `composite experiment items` are mainly used to define multiaxial (and/or anisothermal) loadings in simulations.

Experiments management dialogs

The main dialog is made of 3 selection boxes, one for each of the 3 types of `experiment items` described in the previous section, and various commands that can be applied on the selected items.



Add command:

This command allows to create `base experiments items`, ie. X-Y curves defined by selecting 2 columns of an input ASCII file containing the experimental data. By hitting the `Add command`, the user opens the `Add base experiment dialog` (page 3.5).

Derive command:

This command creates a new `derived experiment item` from the selected `experiment item`. Either `base` or `derived` types of items can be derived from. By hitting the `Derive command`, the user opens the `Add derived experiment dialog` (page 3.7).

Composite command:

This command creates a new `composite experiment item` that allows to regenerate multicolumns data by pasting together `base` or `derived experiment items`. By hitting the `Composite command`, the user opens the `Add composite experiment dialog` (page 3.17).

Mod command:

This command edits the selected `simulation item`, allowing to change its definition.

Cop command:

Using this command, the user can create a new item by copying the selected `base`, `derived` or `composite experiment item`. The copied item then appears at the end of the corresponding `items selection tree`.

Del command:

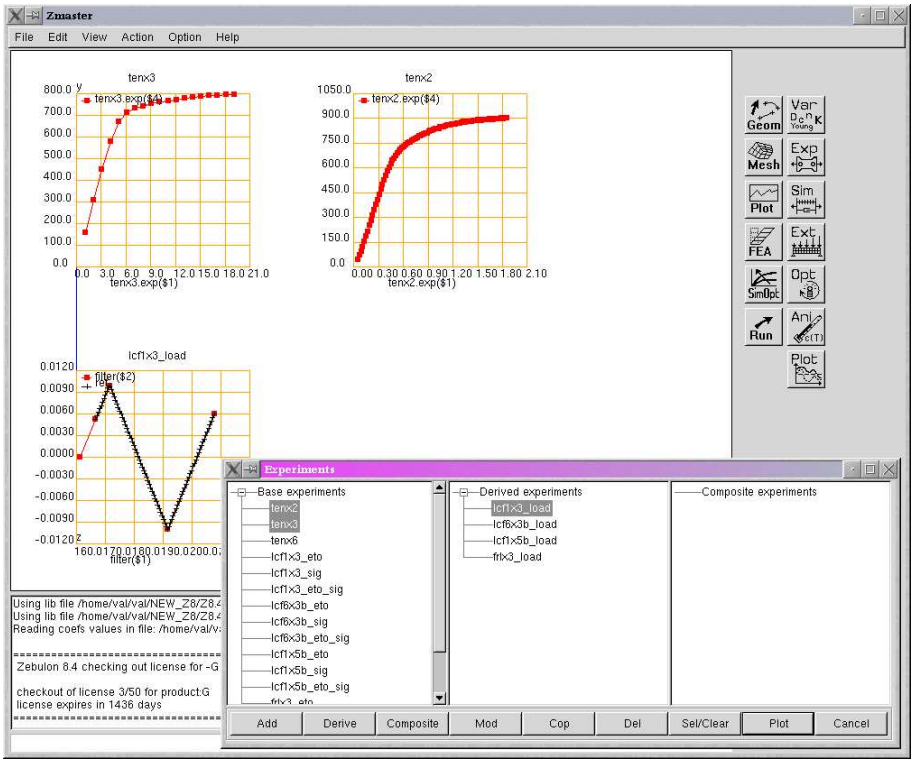
This command deletes from the database all the selected items.

Sel/Clear command:

This command clears all selections in the selection trees associated to the 3 types of `experiment items`.

Plot command:

This command draws all X-Y curves defined in the selected `simulation items`. Plots are tiled in the graphics area across the X-Y plane, as shown in the next figure.

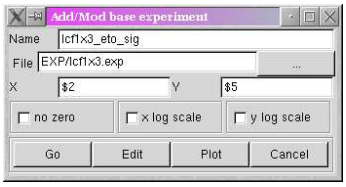


Cancel command:

This command closes the `Experiments management dialog` without further modification to the database.

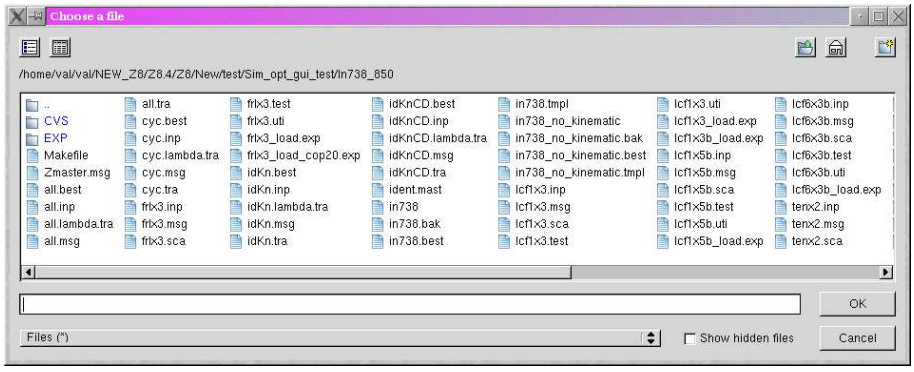
Add base experiment dialog

This command allows to create `base experiment items`, ie. X-Y curves defined by selecting two columns of an input ASCII file containing the experimental data. The graphics dialog opened when the command is activated is shown below.



The following sub-items are managed by the dialog.

- The name of the `experiment item` defined in the A default name is assigned at creation, that should be changed to something more meaningful to the user.
- The name of the file containing the experimental data, specified in the `File` textfield. The browse button available on the right of the textfield opens a file selection dialog that helps selecting an existing file.



- Column indexes in the input ASCII file that will define the `X` and `Y` values to store in the `experiment item`. Syntax is $\$nc$, where nc is the column index required.
- The `no zero` checkbox can be used to specify if the origin of the curve, when not available in the actual data, should be included when drawing the plot. Default is off.
- The `x log scale` checkbox can be used to plot the curve with a logarithmic abscissa. Default is off.
- The `y log scale` checkbox that can be used to plot the curve with a logarithmic ordinate. Default is off.

The following sub-commands may be activated from this dialog.

Go command:

This command updates the database with the informations included in the present dialog.

Edit command:

This command edits the input experimental file specified in the `File` textfield.

Plot command:

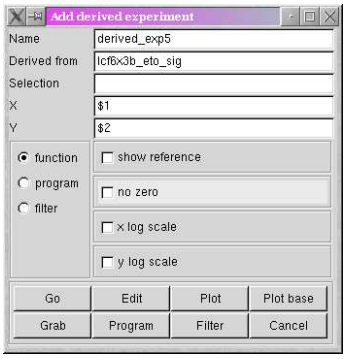
This command draws the X-Y curve stored in the current `base Experiment item`.

Cancel command:

This command closes the current dialog, without further modification to the database.

Add derived experiment dialog

This command is used to apply transformations on pre-existing `base` or `derived experiment items`. The dialog is shown in the next figure.



The name of the `experiment item` can be defined in the `Name` textfield.

The `Derived from` textfield contains the name of the `experiment item` from which new data are derived. At creation, this textfield is filled automatically with the name of the `experiment item` selected before activating the `Derive command` of the `Experiments dialog` (page 3.2).

If the `show reference` checkbox is toggled, the original curve (from which new data is derived) will also be shown when the `derived item` is plotted. Default is off.

The `no zero` checkbox that can be used to specify if the origin of the curve, when not available in the actual data, should be included when drawing the plot. Default is off.

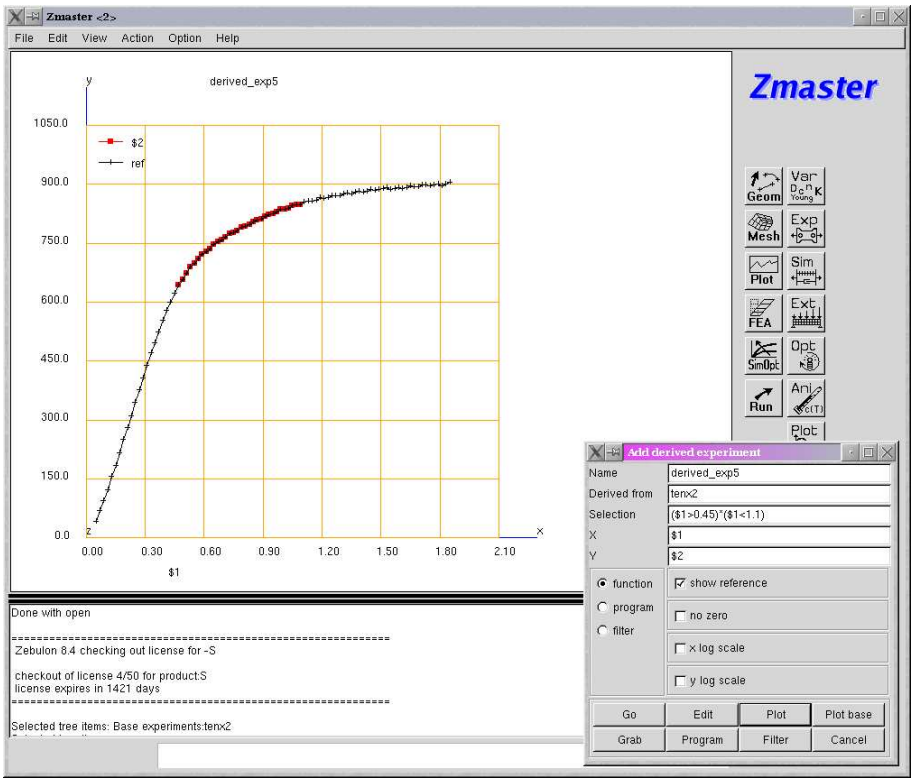
The `x log scale` and `y log scale` checkboxes that can be used to respectively plot the curve with a logarithmic abscissa and ordinate. Default is off.

The `Selection` textfield may be used to select the regions of the input curve that should be retained before applying transformation operations. Any function of the X (entered as "$1") or Y coordinate (entered as "$2") can be specified. Complex selection criteria may be built by using the "$*$" (logical .OR. operator) and "$+$" (logical .AND. operator) operators.

Example:
Entering the following function in the `Selection` textfield :

```
($1>0.45)*($1<1.1)
```

selects portion of the original curve with X values between 0.45 and 1.1, as illustrated in the next figure.



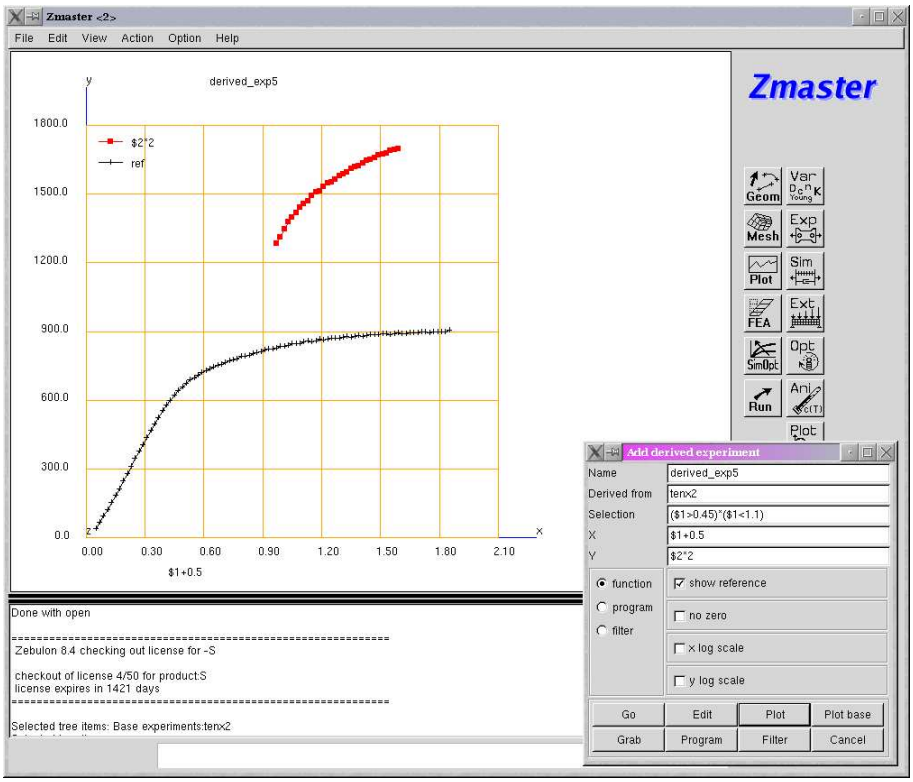
Three modes of transformation are allowed, the particular type chosen being specified by means of the checkbox included in the dialog.

- `Function` mode

  In this case the (X,Y) coordinates of the derived curve are defined by functions entered in the `X` and `Y` textfields. Arguments of these functions are typically the (X,Y) coordinates of the original curve, denoted as "$1" and "$2" in the function definitions. Most mathematical functions of the `libc` library are allowed in the definition.

  Example:

  In the following example the original curve is translated in the X direction by a value of 0.5 (function "$\$1 + 0.5$" entered in the `X` textfield), and Y values are multiplied by 2 (function "$\$2 * 2$" entered in the `Y` textfield). Note, that the same selection criterion as before (function "$(\$1 > 0.45) * (\$1 < 1.1)$" entered in the `Selection` textfield) has been applied, in order to restrict the transformation to a particular region of the input curve. The `show reference` checkbox is toggled on, such that both the original and derived curves are drawn on the same plot.



- `Program` mode

  In this case the transformation is defined by means of a Zprogram script, which may allow more complex transformations than in the previous mode. The user should refer to the Developer's manual for a complete reference on the Zprogram language syntax. The script to be applied can be edited by activating the `Program command`. A default script is provided that simply copies the original curve to the output :

```
void main()
{ int ip;
  for(ip=0;ip<!x;ip++) {
    xres[ip]=x[ip];
    yres[ip]=y[ip];
  }
}
```

  The original (X,Y) values are stored respectively in the "`x`" and "`y`" vectors, while coordinates of the result curve should be defined in the "`yres`" and "`xres`" vectors. By default the latter output vectors have the same size as the input ones. However they can be resized in the script, in which case the input and output curves will have a different number of points.

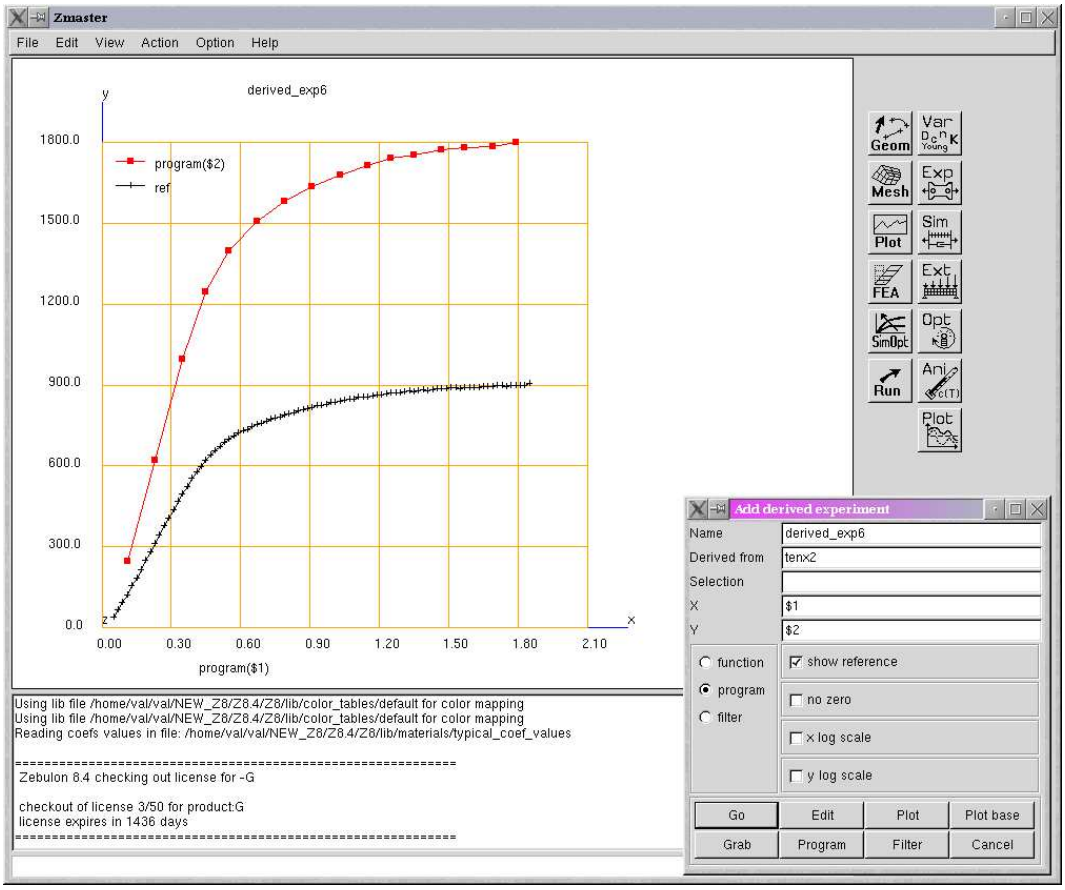
  Example:

  The following script selects one point every each increment of 0.1 along the X axis, and multiplies the Y value by 2.0.

```
void main()
{ int ip, count;
  double xinc, x0;

  xinc=0.1;
  count=0;
  x0=0.;
//Calculation of the number of points retained in the output curve
  for(ip=0;ip<!x;ip++) {
    if((x[ip]-x0)>xinc) {
      count=count+1;
      x0=x[ip];
    }
  }
//Resizing of the ouput vector
  xres.resize(count);
  yres.resize(count);
  count=0;
  x0=0.;
  for(ip=0;ip<!x;ip++) {
    if((x[ip]-x0)>xinc) {
      xres[count]=x[ip];
      yres[count]=2*y[ip];
      count=count+1;
      x0=x[ip];
    }
  }
}
```

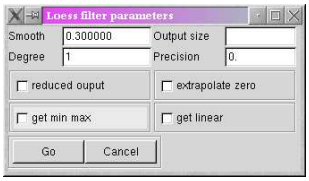
The results are shown in the following figure.



- `Filter` mode

  This mode allows to smooth experimental data by means of the LOESS algorithm LOESS (Cleveland W. S. and Devlin S. J. 1988). In this method the smoothed values are obtained at each point of the original curve, by performing a least-square fit on a subset of the curve centered at the current point. During the fit, the weights assigned to each point in this subset are given decreasing values when the distance to the current point increases. The functions used in the least-square fits may be polynomials of an arbitrary degree. However only linear or quadratic polynomials are used in practice.

  Clicking on the `Filter command` opens the dialog shown in the next figure that allows to specify the various smoothing parameters.



  Sub-items included in the dialog have the following meaning :

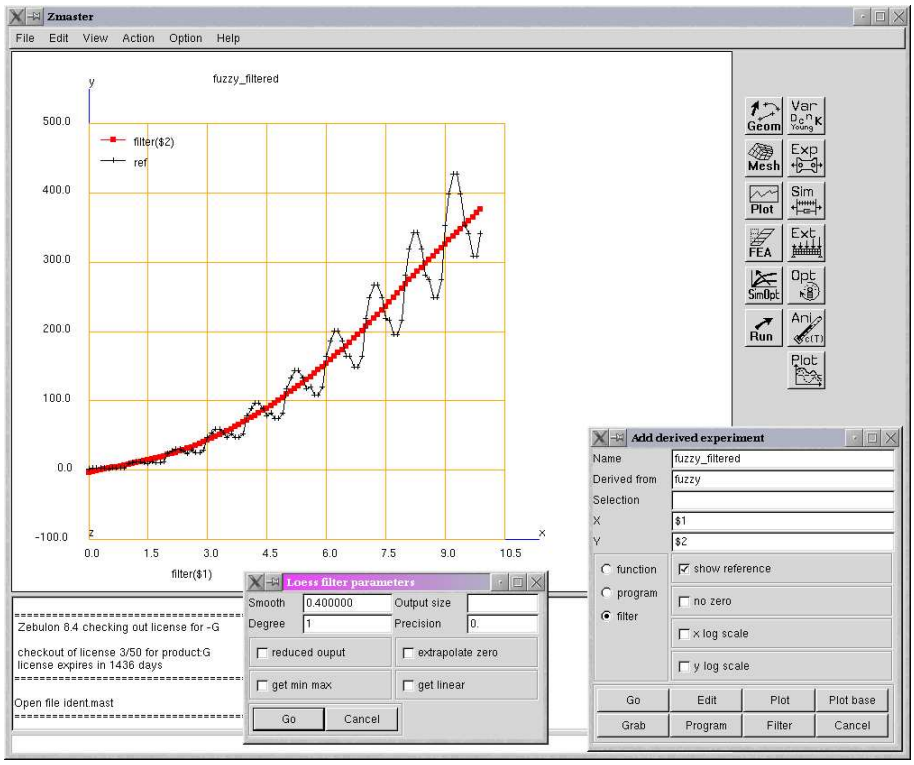
  – The `Smooth` textfield defines the size of the subsets on which least-square fits are performed. Values given should be less than 1.0, and correspond to a fraction of

the total number of points in the original curve. Increasing values of `Smooth` yields smoother curves, a value of 1 corresponding to a simple polynomial fit on the whole curve.

- The `Degree` textfield defines the degree of the polynomial used in the fits. Only values of 1 (linear fit) or 2 (quadratic fit) are allowed. Linear fits yield smoother results, but may miss critical points from the original curve.
- The `reduced output` option can be activated to specify the number of output points (default is to generate a smoothed value for each point in the original curve). In this case the number of output points required should be defined in the `Output size` textfield.
- If the `get min max` option is selected, min/max points of the original curve will be retained in the output, for each reversal of the Y value. The `Precision` parameter may then be used to define the precision involved in the detection of these reversal points. A fine tuning of this `Precision` parameter may be necessary to avoid non significant reversal points on fuzzy data. This option can be used in conjunction with the `reduced output` option. Note that in this case the `Smooth` parameter defines the size of subsets as a fraction of the number of points between 2 reversals. Similarly the `Output size` parameter is the number of output points required between 2 reversals.
- If the `extrapolate zero` option is activated, a point with a null Y coordinate will be calculated and included in the output curve. This option is particularly interesting when the `experiment item` generated is used to define a simulation loading (page 4.12).
- If the `get linear` option is activated, the algorithm attempts to generate a piecewise linear output, by detecting change in slopes of the smoothed curve. In this case, an output point will be generated for each change in slope. The `Precision` parameter can be tuned to control the precision involved in the detection. This option is exclusive and cannot be used in conjunction with the `get min max` and `reduced output` options.

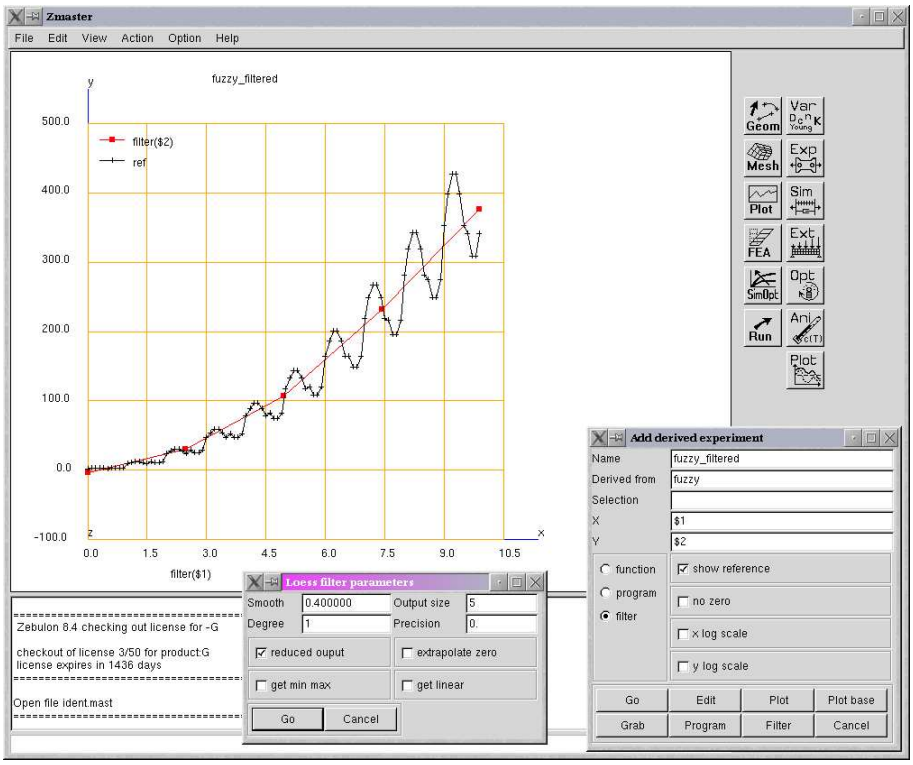
Example:

The following figure illustrates the effect of the smoothing algorithms implemented.



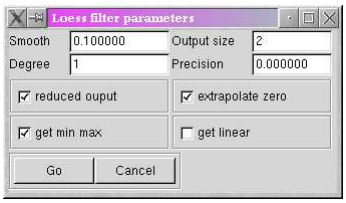
Example:

The following figure illustrates the use of the `reduced output` option on the same input data as before.

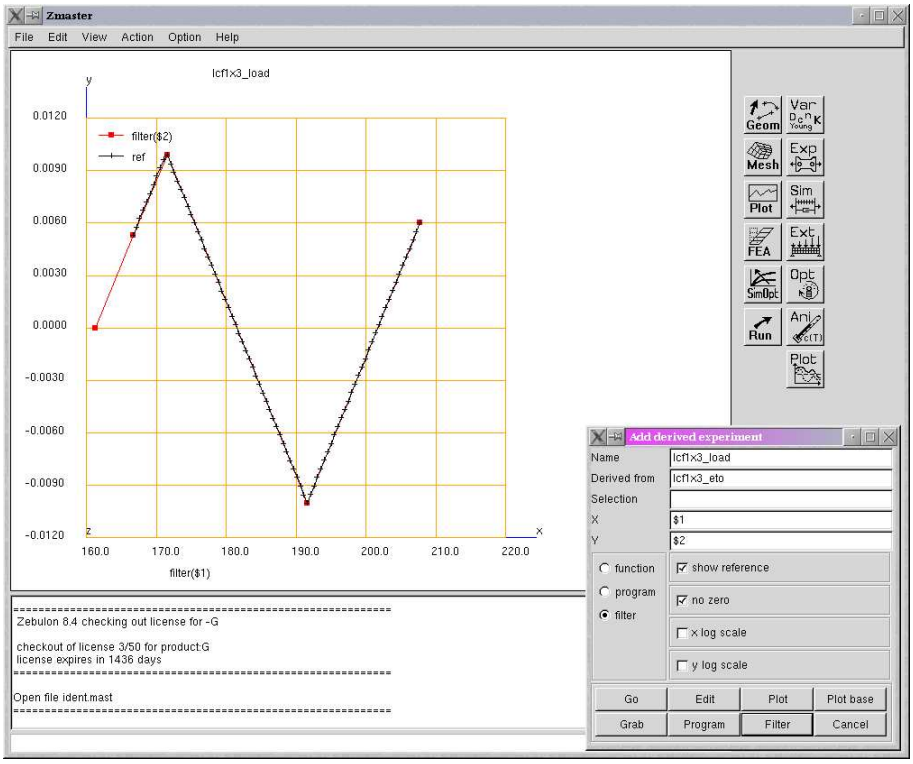


Example:

The following example illustrates the use of the `reduced output`, `get min max` and `extrapolate zero` options, as defined in the dialog shown in the next figure.



Results obtained are shown hereafter.



The various sub-commands included in the `Add derived experiment dialog` are summarized hereafter.

Go command:

This command updates the database with the current definition of the `derived experiment item`.

Edit command:

This command edits an output file containing the (X-Y) values obtained by applying the transformation operations selected in the dialog.

Plot command:

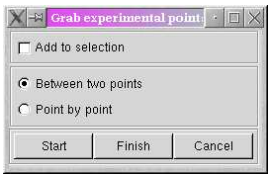
This command plots the (X,Y) curve stored in the `derived experiment item`.

Plot base command:

This command plots the original (X,Y) curve, that is stored in the `experiment item` from which new data are derived.

Grab command:

This command may be used to update the selection defined in the `Selection` textfield, by clicking on various regions of the original curve. The corresponding dialog is shown in the next figure. Selection of single points, or portions of the curve included between two selected points, are allowed. Also, options to either erase or add to the selection criterion already defined in the `Selection` textfield are available.



Program command:

This command edits the Zprogram script used to generate the derived results when the `program` option is selected.

Filter command:

This command opens the `Loess filter parameters dialog` that allows to set various parameters involved in the smoothing algorithm (page 3.11) applied when the `filter` option is selected.

Cancel command:

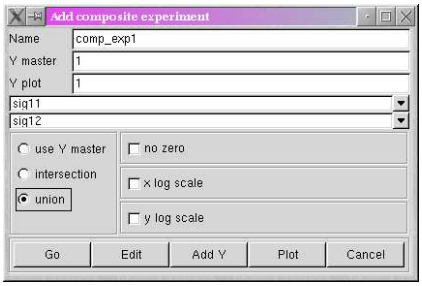
This command closes the dialog without further modification to the current `derived experiment item`.

Add composite experiment dialog

This type of `experiment item` is used to re-generate multi-columns data from (X-Y) curves stored in `base` or `derived experiment items`. The main usage of those is to define multiaxial and/or anisothermal simulation loadings as described at page 4.12.

`Composite items` store data of the form (X,Y1,...,Yn), where n is the number of non-composite `experiment items` that are pasted together. The Yi values are calculated from the Y values of the i^{th} non-composite item, and the X values are computed as a combination of available X values in the latter items.

Definition of `composite experiment items` uses the dialog shown in the next figure.



The `Name` textfield defines the name of the item in the database. A default name is assigned at creation, that should be changed to something more meaningful by the user.

By default, two non-composite items are used in the definition. The corresponding combo boxes are filled with the name of all non-composite items defined previously in the database, and a particular item in this list should be selected in both cases. The `Add Y command` allows to add an unlimited number of non-composite `experiment items` in the definition. Activation of this command adds a new combo box in the dialog, that allows to select the name of the new experiment added.

Three modes of generation are available :

- `use Y master mode`. If this mode is selected the master curve must be specified in the `Y master` textfield. The rank of the master curve within the list of non-composite items defining the `composite experiment item` is expected. In this case, the X values of the `composite item` generated will correspond to those of the master curve. Y values associated to slave curves will be computed by linear interpolation if no data is available at the current X coordinate.

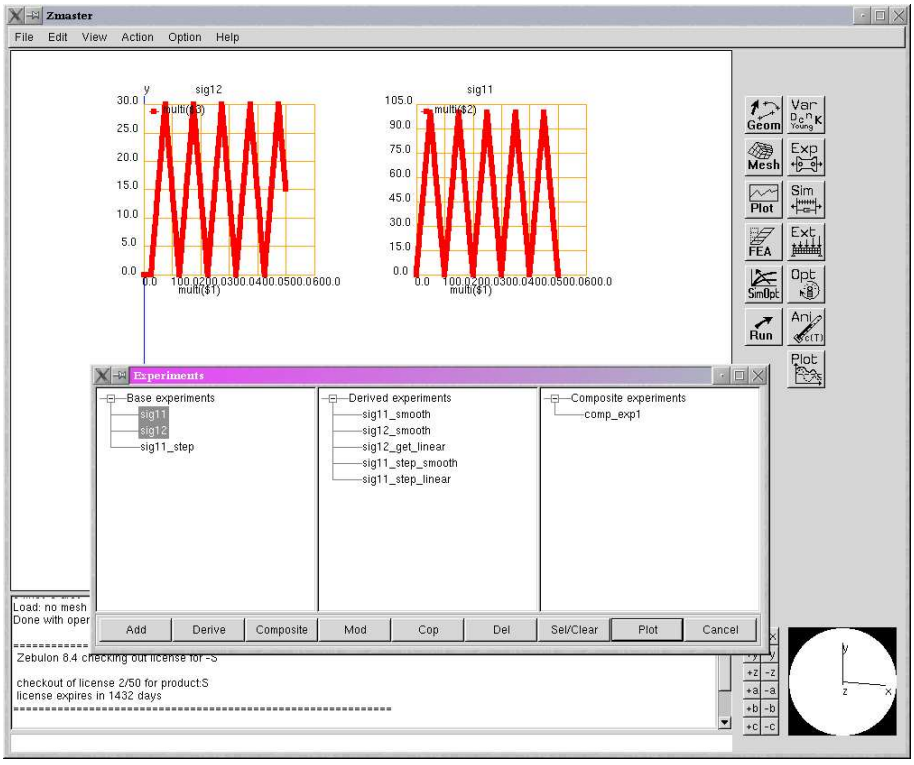
- `intersection mode`. Only points for which the X values correspond in the curves added in the `composite item` will be generated.

- `union mode`. The X values generated will be an union of the X values available in the curves added in the `composite item`. Linear interpolation is used to calculate missing Y values.

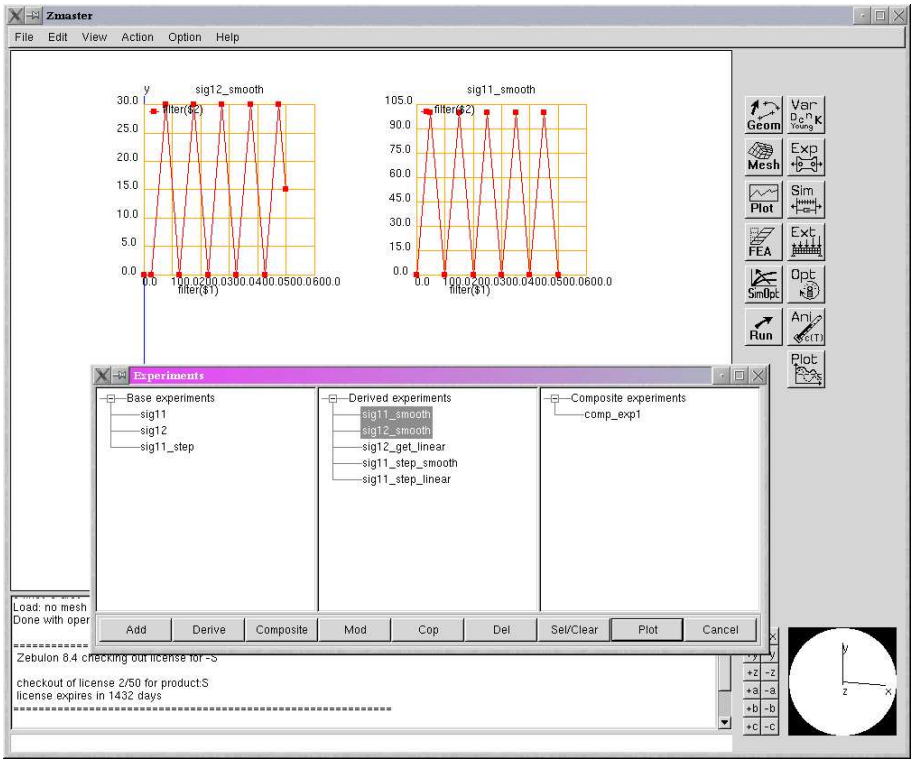
The `Y plot` textfield allows to specify which Y value will be plotted when the `Plot command` is hit. The `Edit command` edits an ASCII file with the generated results.

Example:

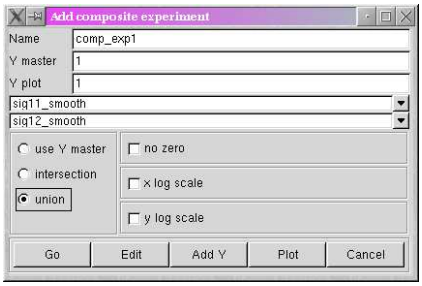
Let us assume that the experimental data define a multiaxial (sig11,sig12) loading. Two `base experiment items` are extracted from the experimental file, one for each component of the loading, as shown in the next figure. Note that both sig11 and sig12 are periodic, but the loading is out of phase.



`Derived experiment items` are then generated with the `filter` option (page 3.11) to extract only the loading extremum points.



Next a `composite item` is defined to store both components of the filtered multiaxial loading. The corresponding input is shown in the next figure. The `union` option has been selected.



The new curves obtained are shown hereafter. The output includes a point corresponding to each extremum point of the 2 out of phase loading components, and can be used to define a valid multiaxial simulation loading.


Zmaster
File Edit View Action Option Help
comp_exp2
sig12_smooth($2)
sig11_smooth($1)
comp_exp1
sig11_smooth($2)
sig11_smooth($1)
Load: no mesh available...
Done with open
Zebulon 8.4 checking out license for -S
checkout of license 2/50 for product:S
license expires in 1432 days
Geom
Var
Mesh
Exp
Plot
Sim
FEA
Ext
SimOpt
Opt
Run
Ani
Plot

Chapter 4

Simulation items management

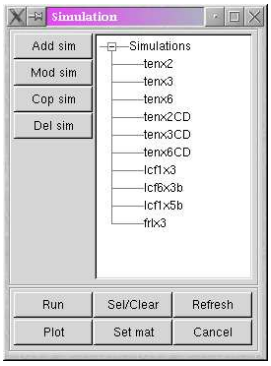
Simulations

The `Simulations command` allows to build interactively input files for the Zsim module (see the Zmat user manual for a complete reference on this program). The simulation results can then be used to define comparisons with experimental results to drive the optimization module.

Note that simulations can be launched either from this dialog, or from the `Variables Panel` (page 2.8).

Simulations management dialog

The main dialog is shown in the following figure. The selection tree on the right allows to select `simulation items` to apply appropriate commands on.



Add sim command:

This command may be used to create a new `simulation item` in the database by opening the `Add/Mod simulation dialog` (page 4.6).

Mod sim command:

This command edits the selected `simulation item`. By hitting the `Mod sim command`, the user opens the `Add/Mod simulation dialog` that allows to change the definition of the item (page 4.6).

Cop sim command:

This command creates a new `simulation item`, by copying the informations contained in the selected `simulation item`. The new item can then be edited by means of the `Mod sim command`.

Del sim command:

This command deletes from the database all the `simulations items` selected in the `Simulation items selection tree`.

Run command:

This command launches all the simulations selected in the `Simulation items selection tree` using the Zsim module.

Sel/Clear command:

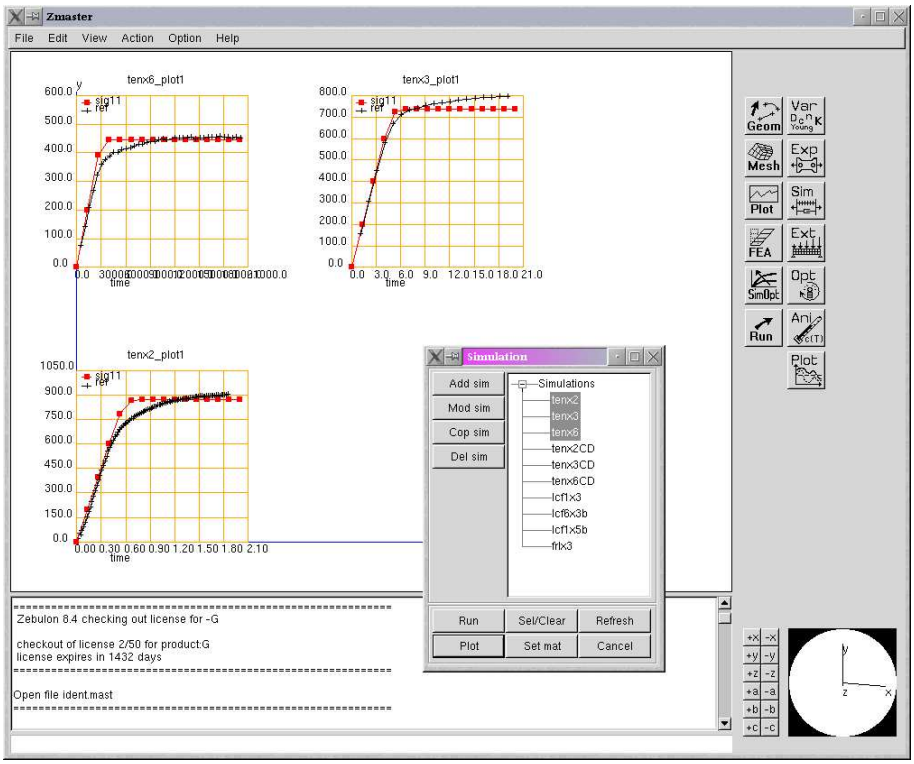
This command clears all selections in the `Simulation items selection tree`.

Refresh command:

This command updates the `Simulation items selection tree` with the informations contained in the `Update simulations dialog` that allows by means of the `Anisothermal variables dialog` to create new `simulation items`.

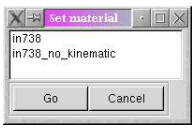
Plot command:

This command draws all output curves defined in the `simulation items` selected in the `Simulation items selection tree`. Plots are tiled in the graphics area across the X-Y plane, as shown in the following figure.



Set mat command:

This command may be used to set the material file of several simulations at the same time, by opening the `Set material dialog` shown in the next figure.



By hitting the `Go command`, the material file selected in the `Material file selection box` becomes the material file assigned to all the selected `simulation items`. The `Cancel` button closes the `Set material dialog`.

Cancel command:

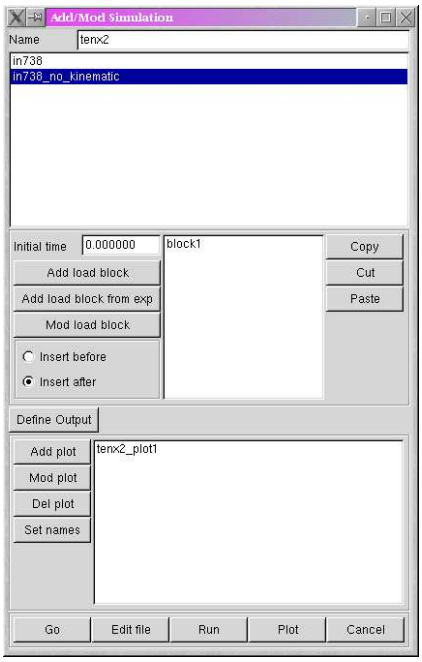
This command closes the `Simulations management dialog` without further modification to the database.

Add/Mod Simulation dialog

This dialog is used to create or modify `simulation items` that will be run using the Zsim module. Typically the following informations are needed to define valid simulations :

- The material behavior, defined in a material file containing values of material coefficients.
- The simulation loading, defined by setting the evolution of components of the strain and stress tensors and external parameters (eg. temperature for anisothermal tests) against time. Mixed strain/stress loadings are allowed by Zsim.
- The simulation output (stress/stress components, material internal variables, external parameters ...) that will be stored in ASCII files.
- Various plots that can be defined from the simulation output results to check the material response under the prescribed loading.

The dialog is shown in the next figure.



The name of the `simulation item` can be modified in the `Name` textfield. The default name assigned at creation should be changed to something more meaningful to the user. The selection box at the top of the dialog is filled with the names of all template files loaded in the application, and attached to design `variables` as described in the `Variables management` chapter (page 2.2). Assigning a material behavior to the `simulation item` is done by selecting a particular file in this list.

Those template files are the basis of the mechanism that links the `variables` and their current values to the `simulation items` : when a simulation is launched, the material file

selected in the `Material file selection box` is updated with the current values of the `variables` database.

The user can define the initial time of the simulation by filling the `Initial time` textfield command that may be used to define a non-zero *tbeg* starting time for the loading sequence, in which case the *time* values entered in the table of each load block will be incremented by the *tbeg* value entered. Note that the initial time is automatically filled from the first defined time of the first load block.

Concerning the load blocks, two modes of insertion are allowed, the particular type chosen being specified by means of the checkbox included in the dialog. If the `Insert before` mode is chosen, each created load block will be inserted before the selected load block. If the `Insert after` mode is chosen, each created load block will be inserted after the selected load block. Default is `Insert after` mode.

Various sub-commands available in this dialog are the following ones.

Add load block command:

This command opens the `Define load block dialog` that allows to specify manually the simulation loading (page 4.10).

Add load block from exp command:

This command opens the `Define load block from experiment dialog` that allows to use an `experiment item` available in the database to define the simulation loading (page 4.12).

Mod load block command:

This command opens the `Define load block dialog` corresponding to the selected load block, that allows to manually modify the simulation loading (page 4.10).

Copy command:

This command copies the selected load block (without suppressing it). Note that the user is then supposed to select an other load block, to choose the appropriate `Insert before` or `Insert after` mode, and to hit the `Paste command`.

Cut command:

This command cuts the selected load block. Note that the user can use this command to simply delete a load block, or to paste it.

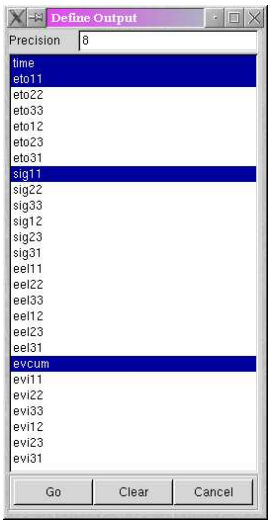
Paste command:

This command pastes a previously copied or cut load block. Note that the user is supposed to have chosen the appropriate `Insert before` or `Insert after` mode, and to have selected a load block before.

Define Output command:

This command opens a window containing a selection box filled with all the output material quantities supported by the behavior selected. A material file should have been selected

beforehand in the `Material file selection box`, otherwise the command returns an error message. The output generated in the Zsim results files is specified by selecting the required quantities as illustrated in the next figure.



The `Precision` textfield defines the number of digits printed for floating point values in the ASCII results files (default is 8). The `Clear` button clears the current selection. The `Go` button updates the database with the current selection, while the `Cancel` button closes the `Define Output dialog` without further modification to the database.

Add plot command:

This command adds a new `plot item` to the current `simulation item` by opening the `Add/Mod plot dialog` described at page 4.14.

Mod plot command:

This command edits the `plot item` selected in the `Plot items selection box` to allow some modifications. The `Add/Mod plot dialog` described at page 4.14 is opened with the current definition of the `plot item`.

Del plot command:

This command deletes from the `simulation item` definition all the `plot items` selected in the `Plot items selection box`.

Set names command:

This command assigns to all the defined plots a name constituted by the name of the `simulation item` and the rank of each plot in the `Plot items selection box`.

Go command:

This command updates the database with the definition of the `simulation item` currently edited.

Edit file command:

This command edits the Zsim input file generated automatically by the application from the informations defined in the present dialog. Some Zsim commands can be directly input in this file, and will be preserved each time the file is re-written by the application. In particular, the following commands are only accessed by means of the input file (see the Simulation Chapter in the Zmat manual for a description of the syntax) :

- `**solver` command
- `*integration` command in the `**model` section

Default values are provided for these commands however, such that direct access of the input file is seldom needed.

Run command:

This command launches the simulation currently edited using the Zsim module.

Plot command:

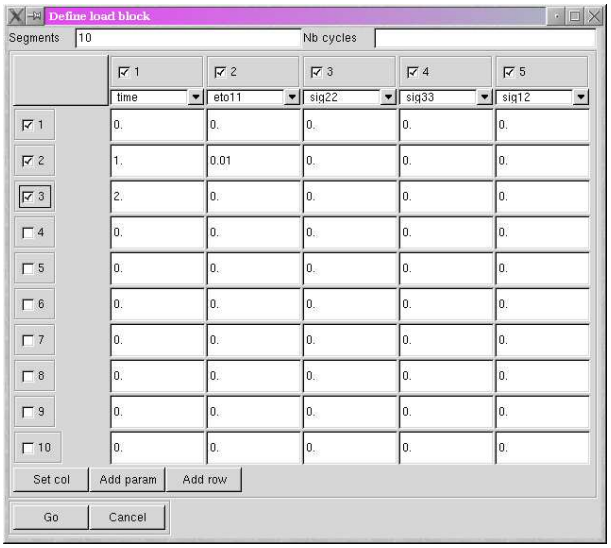
This command draws all curves stored in the `plot items` attached to the current simulation. Plots are tiled in the graphics area across the X-Y plane.

Cancel command:

This command closes the dialog without further modification to the database.

Define load block dialog

When this command is activated for the first time the dialog shown in the next figure is opened.



Textfields for entering numerical values that define the loading are organized in a tabular manner. In this table, the first column always defines the *time*, and other columns represent one particular component of the loading (eg. stress/strain components or external parameters). Rows in this table can be used to set the loading components to the value prescribed at the particular time defined in the first cell of the row. Note that the first *time* value is expected to be always 0.0, and that the `Initial time` textfield should be used to define a loading that does not start at 0.0.

The checkboxes at the head of each column allow to activate/deactivate a particular loading component. The default loading corresponds to a 2D simulation (ie. only 11, 22 33 and 12 components of the stress/strain tensors are activated). To allow a 3D simulation the 23 and 13 shear components should be activated in the corresponding checkboxes at the top of the table. Similarly, definition of an anisothermal loading requires activation the *param:temperature* component in the last column of the dialog. The combo boxes at the head of each column, may be used to choose between a strain (*etoij* components) or stress (*sigij* components) control for this particular component.

Checkboxes at the beginning of each line can be used to add or remove a *time* sequence within the loading definition (the default dialog includes ten possible *time* sequences, even if only the first two are activated).

Note that, once the loading has been completely defined by clicking on the `Go command`, the next time this dialog will be opened, only line/columns selected previously will be included in the table. Addition of new loading components or time sequences is still possible though, by means of the `Add param` or `Add row commands`.

Other textfields at the top of the dialog have the following meaning :

- the `Segments` textfield defines the number of output points generated in the results files between each values of the time sequence.
- the `Nb cycles` textfield may be set to a *ncyc* non-zero value, in which case the time sequence is expected to define a cycle that will be repeated *ncyc* times. Note that in this case, the *time* values used to define the sequence are local to the cycle.

Various sub-commands defined in this dialog are described hereafter.

Set col command:

This command sets all cells of the column specified to a given value, as illustrated in the following figure.

Add param command:

This command adds a new column in the loading table. The column is labeled *param* by default, and the definition should be changed to something like *param:name* where *name* is the name given to the external parameter in the simulation.

Add row command:

This command adds a new row in the loading table.

Go command:

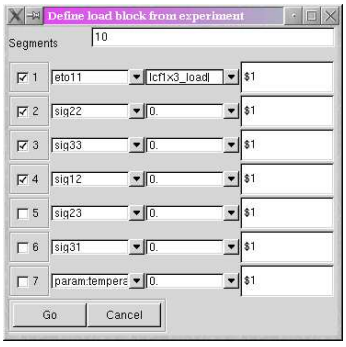
This command updates the loading definition for the current `simulation item`, using the values entered in the dialog.

Cancel command:

This command closes the dialog without updating the loading definition.

Define load block from experiment dialog

This command is used to define a simulation loading from an `experiment item` created beforehand by means of the `Experiments command` (page 3.2). The dialog is shown in the next figure.



The first column in this dialog is used to define the loading components (eg. stress/strain components or external parameters). The combo boxes can be used to choose between strain (*etoij* components) or stress (*sigij* components) control for this particular component. The checkboxes at the beginning of each row allow to activate/deactivate a particular loading component. The default loading corresponds to a 2D simulation (ie. only 11, 22 33 and 12 components of the stress/strain tensors are activated). To allow a 3D simulation the 23 and 13 shear components should be activated in the corresponding checkboxes at the beginning of table rows. Similarly, definition of an anisothermal loading requires activation the *param:temperature* component in the last row of the dialog.

The second column specifies the name of the `experiment item` that will define the evolution of the loading component during time. A combo box is filled with all `experiments items` defined beforehand by means of the `Experiments command`, and the item required must be selected from this list. Note that only one `experiment item` can be chosen in this list for the various loading components, but the same experiment may be assigned to several loading components.

The X value stored in the `experiment item` will be used to define the *time* sequence.

If a `base` or `derived item` type is selected (let us recall that those are used to store X-Y curves) the Y value stored in the item will define the actual loading component values.

Input in the third column is only needed if a `composite` type of `experiment item` is assigned to the loading components (this type of item can store multi-column data). In this case the column number in the `composite item` should be given in the last column of the dialog for each loading component attached to the experiment : syntax is *$nc* where *nc* is the rank of the Y value in the `composite item` definition.

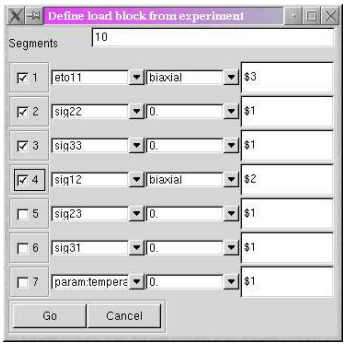
Example:

In the previous figure the *eto11* loading component is assigned by means of the `experiment item` named *lcf1x3_load*, ie. the time sequence and the *eto11* values are defined respectively

by the X and Y values stored in this `experiment item`. Other loading components (*sig22*, *sig33* ...) are assigned to a null constant value over the whole time sequence.

Example:

The next figure illustrates the use of a `composite experiment item` to define a multiaxial loading. Both the *eto11* and *eto12* loading components are defined by means of a `composite experiment item` named *biaxial*. Within this item, the X values will define the time sequence, while the `Y3` and `Y2` values (the third and second Y values stored in the item, specified as *$3* and *$2* in the last column of the dialog) will be used to define the *eto11* and *eto12* components respectively.



The `Segments` textfield defines the number of output points generated in the results files between each values of the time sequence stored in the X column of the `experiment item`.

The following sub-commands can be activated from this dialog.

Go command:

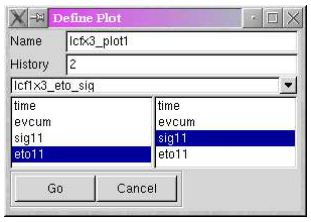
This command updates the loading definition for the current `simulation item`, using the values entered in the dialog.

Cancel command:

This command closes the dialog without updating the loading definition.

Define plot dialog

This dialog, used to create or modify a `plot item` attached to the current `simulation item`, is shown in the figure hereafter.

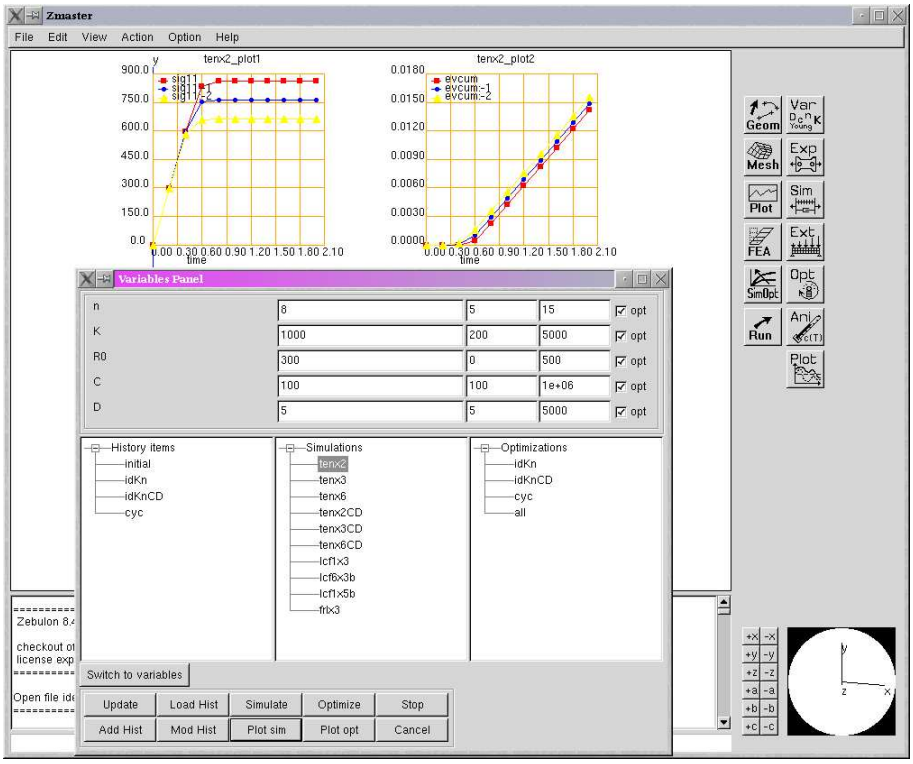


The following sub-items are accessed from this dialog.

- The name of the plot item defined in the `Name` textfield.
- The number *nc* of curves entered in the `History` textfield, corresponding to the *nc* more recent simulations, that will be drawn on the same plot. This type of plots is a convenient way to visualize for instance the influence of material coefficients on the simulated response.

  Example:

  In this example, three simulations have been performed for increasing values of the initial yield stress *R0*. Plots of *sig11* vs *eto11* and the cumulated plastic strain *evcum* vs *time* obtained by setting the `History` parameter to 3 are shown in the next figure.



- A combo box filled with the `experiment items` defined beforehand by means of the `Experiments command` (page 3.2), where a particular item can be selected to be used as a reference curve drawn on the same plot as the simulation curve.
- Two selection boxes filled with the simulation output components defined by the `Define Output command` described at page 4.7. A particular component should be selected to specify the X (left box) and Y (right box) parameters that will be drawn on the plot.

The following sub-commands can be activated from this dialog.

Go command:

This command updates the plot definition using the values entered in the dialog.

Cancel command:

This command closes the dialog without updating the plot definition.

Chapter 5

External simulation items management

External simulations

.

The `External simulations command` allows to define and launch interactively simulations performed with an arbitrary external script or program, and especially programs that can not be launched using `Simulations command`.

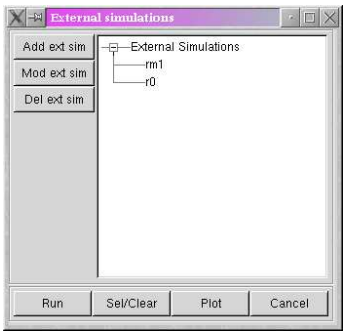
`Simulations command` is restricted to RVE calculations (using the Zsim module) that only apply to standard experimental tests where the response may be considered as homogeneous (eg. conventional tensile tests). For other experimental tests where a complete structural calculation may be needed (eg. bending tests), the `External simulations` mode can be used to drive a finite element code or any external program within Sim-Opt Plugin. An external simulation can be a list of several executions commands.

The external simulation results can then be used to define comparisons with experimental results to drive the optimization module.

Note that external simulations can be launched either from this dialog, or from the `Variables Panel` (page 2.8).

External simulations management dialog

The `External simulations dialog` is shown in the following figure. The selection tree allows to select `external simulation items` to apply appropriate commands on.



Add ext sim command:

This command may be used to create a new `external simulation item` in the database by opening the `Add/Mod external sim dialog` (page 5.5).

Mod ext sim command:

This command edits the selected `external simulation item`. By hitting the `Mod ext sim command`, the user opens the `Add/Mod external sim dialog` that allows to change the definition of the item (page 5.5).

Del ext sim command:

This command deletes from the database all the `external simulation items` selected in the `External simulation items selection tree`.

Run command:

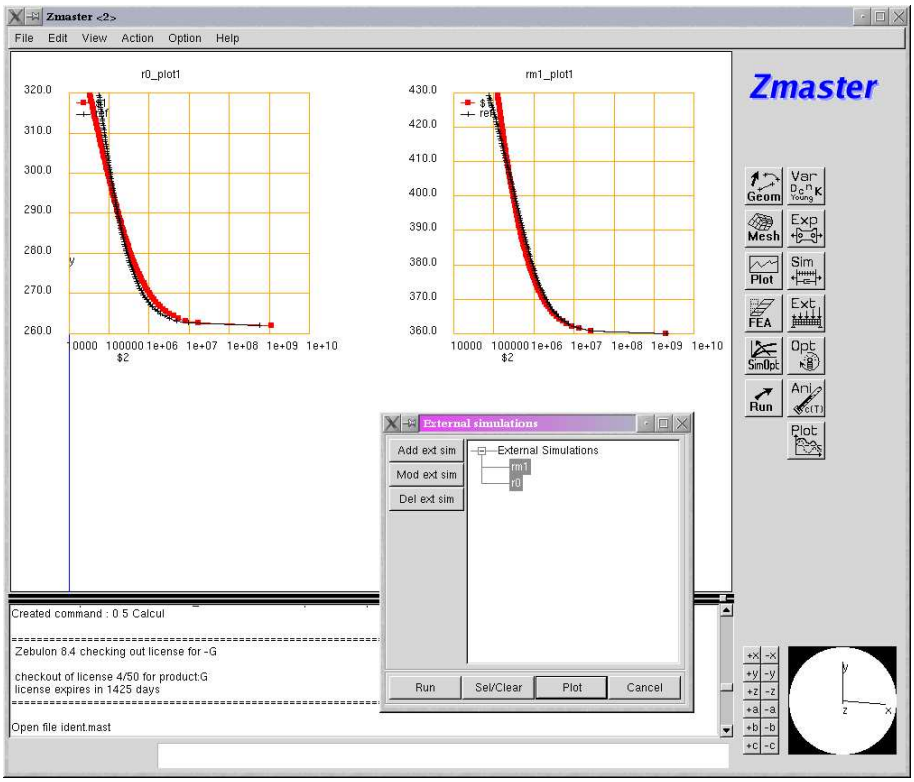
This command launches all the external simulations selected in the `External simulation items selection tree` using the appropriate module.

Sel/Clear command:

This command clears selections in the `External simulation items selection tree`.

Plot command:

This command draws all output curves defined in the `external simulation items` selected in the `External simulation items selection tree`. Plots are tiled in the graphics area across the X-Y plane, as shown in the following figure.

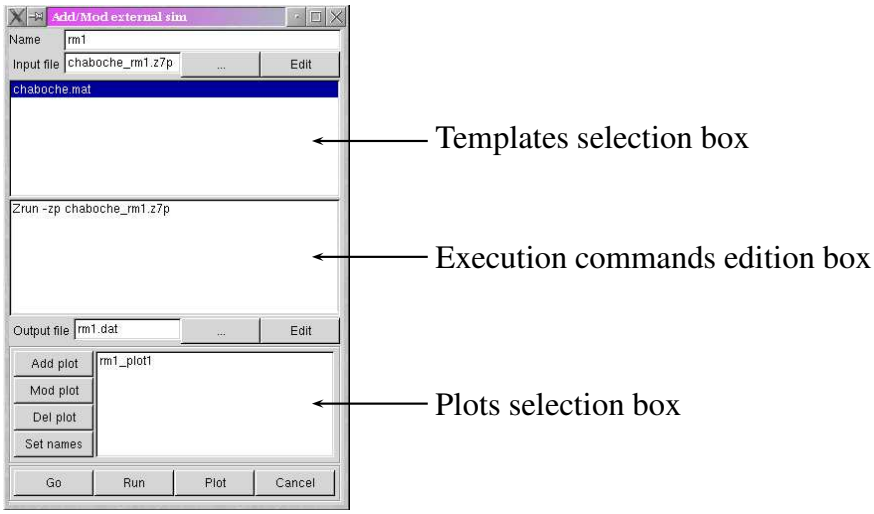


Cancel command:

This command closes the `External simulations management dialog` without further modification to the database.

Add/Mod external sim dialog

This dialog, used to create or modify `external simulation items` that will be run using the appropriate module, is shown in the next figure.

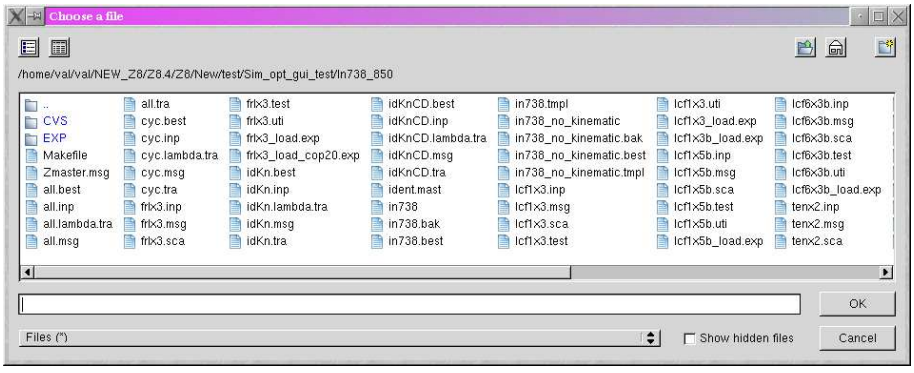


The proceedings of this dialog consists of :

- firstly, filling the upper part of the dialog, to define a name, to select an input file and a template file, and then to use or modify execution command(s) ;
- secondly, using the `External simulation` results by means of the lower part of the dialog, to define as many plots as needed with available variables.

The name of the `external simulation item` can be modified in the `Name` textfield. The default name assigned at creation should be changed to something more meaningful to the user.

The `Input file` textfield may be filled by the user or by means of the browse button available on the right of the textfield that opens a file selection dialog to help the selection of an existing file as shown in the next figure.



The first selection box at the top is filled with the names of all template files loaded in the application. Assigning a material file behavior to the `external simulation item` is done by selecting a particular file in this list.

The second box is used to define execution commands. The default execution commands written at creation should be adjusted to the particular commands needed to launch the external scripts or programs. Note that as many execution commands as needed may be specified, in that case they are separated by a carriage return in the `Execution commands edition box`.

The informations given in the `Input file` textfield is only indicative. Setting on actual value allows to edit the input file to check the data needed by the execution commands. A template should be selected in the `Template selection box` (page 5.5). This template file should correspond to an input file needed by the execution commands. Note that in the current implementation, there is only one template file allowed, such that all `variable` definitions that may affect the execution commands results should be located in the same file.

The `Output file` textfield allows to define the name of a file (generated by the execution commands) that may be used to draw plots. This output file should include ASCII data where the number of lines corresponds to the number of points, and columns (separated by a blank space character) correspond to the various components that can be plotted. If the output file contains the name of the components as headers, these headers (appearing in the two selection boxes of the `Define Plot dialog`) may be used to define the plot, otherwise the definition of the plot in the `Define Plot dialog` uses the number of the columns (for instance $1 or $2) to specify the X and Y parameters that will be drawn.

The following sub-commands may be activated from this dialog.

Edit command:

This command edits the file specified by means of the `Input file` or `Output file` textfield. Note that the `Edit input file dialog` allows to modify the selected file, using the `Save command`, while the `Edit output file dialog` does not allow to modify the selected file.

Add plot command:

This command adds a new `plot item` to the current `external simulation item` by opening the `Define Plot dialog` described at page 5.8.

Mod plot command:

This command edits the `plot item` selected in the `Plot items selection box` to allow some modifications. By hitting the `Mod ext sim command`, the user opens the `Define Plot dialog` (described at page 5.8) with the current definition of the `plot item`.

Del plot command:

This command deletes from the `external simulation item` definition all the `plot items` selected in the `Plot items selection box`.

Set names command:

This command assigns to all the defined plots a name constituted by the name of the `external simulation item` and the rank of each plot in the `Plot items selection box`.

Go command:

This command updates the database with the definition of the `external simulation item` currently edited.

Run command:

This command launches the `external simulation item` considered using the appropriate module.

Plot command:

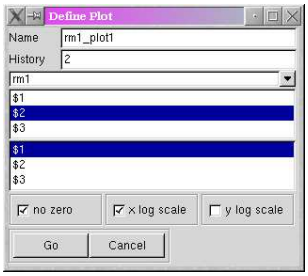
This command draws all output curves defined in the `external simulation item`. Note that plots are tiled in the graphics area across the X-Y plane.

Cancel command:

This command closes the `Add/Mod external sim dialog` without further modification to the database.

Define plot dialog

This dialog, used to create or modify a `plot item` attached to the current `external simulation item`, is shown in the figure hereafter.

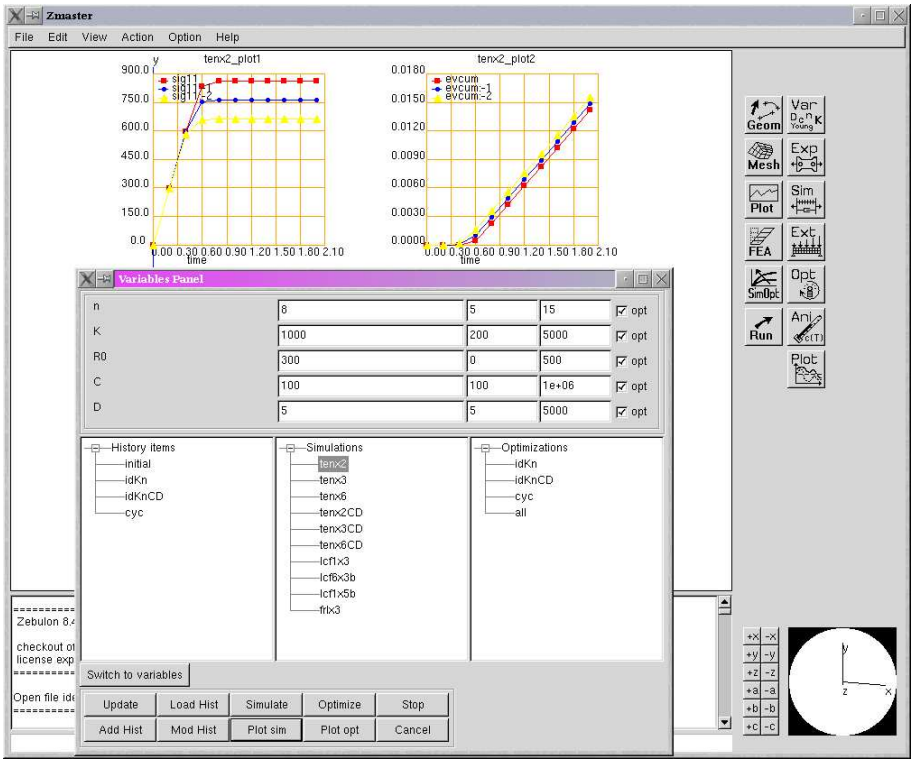


The name of the `plot item` can be modified in the `Name` textfield. The default name assigned at creation should be changed to something more meaningful to the user.

The `History` textfield allows to define the number *nc* of curves the user wants to display on this particular plot. The plotted *nc* curves will correspond to the *nc* more recent external simulations. This type of plots is a convenient way to visualize for instance the influence of material coefficients on the simulated response.

Example:

In this example, 3 simulations have been performed for increasing values of the initial yield stress *R0*. Plots of *sig11* vs *eto11* and the cumulated plastic strain *evcum* vs *time* obtained by setting the `History` parameter to 3 are shown in the next figure.



The combo box filled with the available `experiment items` defined beforehand may be used to select a particular item that will be drawn on the same plot as a reference curve.

The two selection boxes filled with the components of the external simulation output file selected in the `Output file` textfield of the `Add/Mod external sim dialog`, allows to respectively specify the X (top box) and Y (bottom box) parameters that will be drawn on the plot.

The `no zero` checkbox can be used to specify if the origin of the curve, when not available in the actual data, should be included when drawing the plot. Default is off.

The `x log scale` and `y log scale` checkboxes can be used to respectively plot the curve with a logarithmic abscissa and ordinate. Default is off.

The following sub-commands can be activated from this dialog.

Go command:

This command updates the definition of the current plot of the considered `external simulation item`, using the informations included in the dialog.

Cancel command:

This command closes the `Define Plot dialog` without updating the definition of the current plot.

Chapter 6

Optimization items management

Optimizations

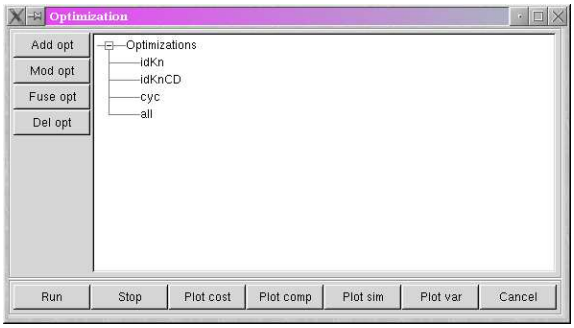
.

The `Optimizations command` allows to build interactively input files for the Zopt module (see the Zmat user manual for a complete reference on this program). An `optimization item` is based on a comparison between a `simulation item` and an `experiment item`, to fit the simulation to the experiment by adjusting simulation variables.

Note that optimization procedures defined can be launched either from this dialog, or from the `Variables Panel` (page 2.8).

Optimizations management dialog

The main dialog is shown in the following figure. The selection tree allows to select `Optimization items` to apply appropriate commands on.



Add opt command:

This command may be used to create a new `optimization item` in the database by opening the `Add/Mod optimization dialog` (page 6.7).

Mod opt command:

This command edits the selected `optimization item`. By hitting the `Mod opt command`, the user opens the `Add/Mod optimization dialog` that allows to change the definition of the item (page 6.7).

Fuse opt command:

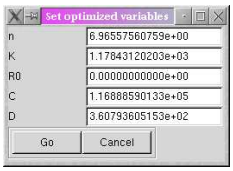
This command allows to paste the comparisons between experimental and simulation results that define the different `optimization items` selected. By fusing the selected `optimization items`, the user will create a new `optimization item` containing all the comparisons. Note that a simple copy is obtained if a single `optimization item` is selected.

Del opt command:

This command deletes from the database all the `optimization items` selected in the `Optimization items selection tree`.

Run command:

This command launches the optimization selected in the `Optimization items selection tree` using the Zopt module. The optimization runs in a separate execution thread and can be stopped at any moment using the `Stop command`. When the optimization terminates, either because of convergence or by activation of the `Stop command`, a window is opened showing the best values found by the optimizer, as shown in the next figure.



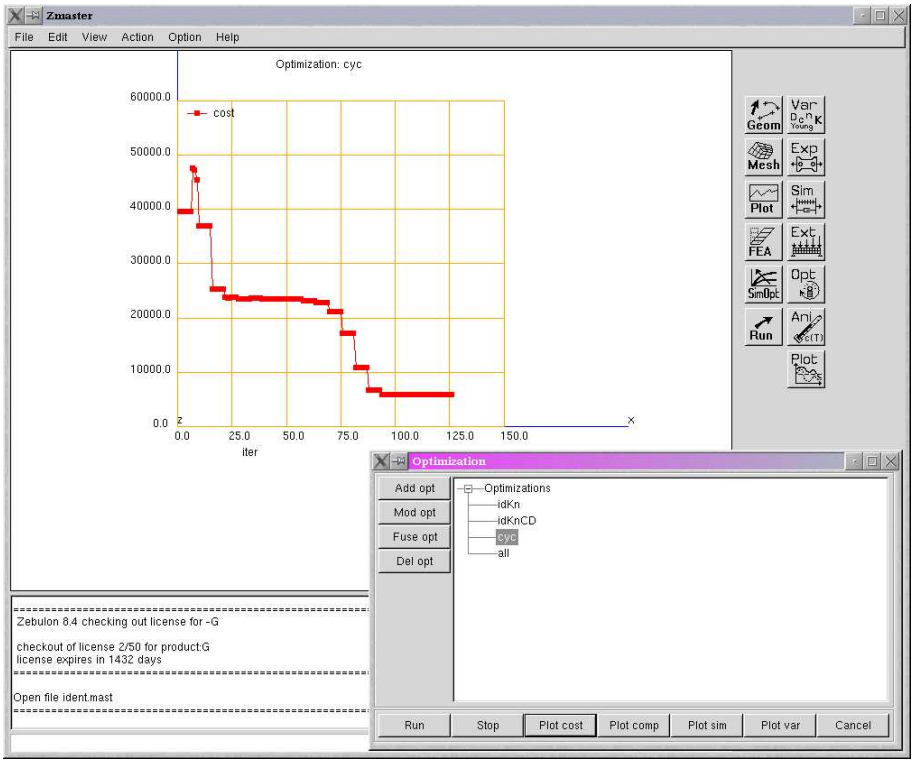
The `Go` button must be clicked if the user does want to update the `Variables` database with the optimized values. Otherwise, values are left unchanged.

Stop command:

This command stops the optimization that is currently running in a separate thread. As described above, a window is popup, that proposes to update the `Variables` database with the best values found until the `Stop command` was hit.

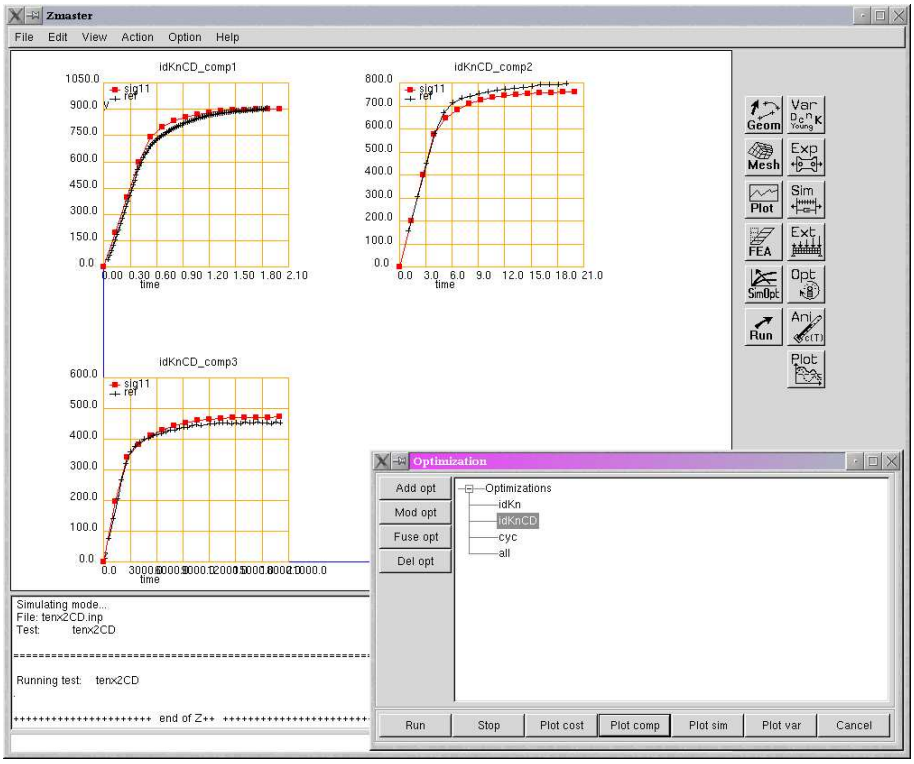
Plot cost command:

This command plots the evolution of the cost function associated with comparisons between experimental and simulation results during optimization iterations. The command can be activated to check the convergence while the optimizer is currently running.



Plot comp:

This command draws comparisons between experimental and simulation results that define the `optimization item` selected. The command can be activated when the optimizer is running to visualize the current distance between experimental and simulation curves.

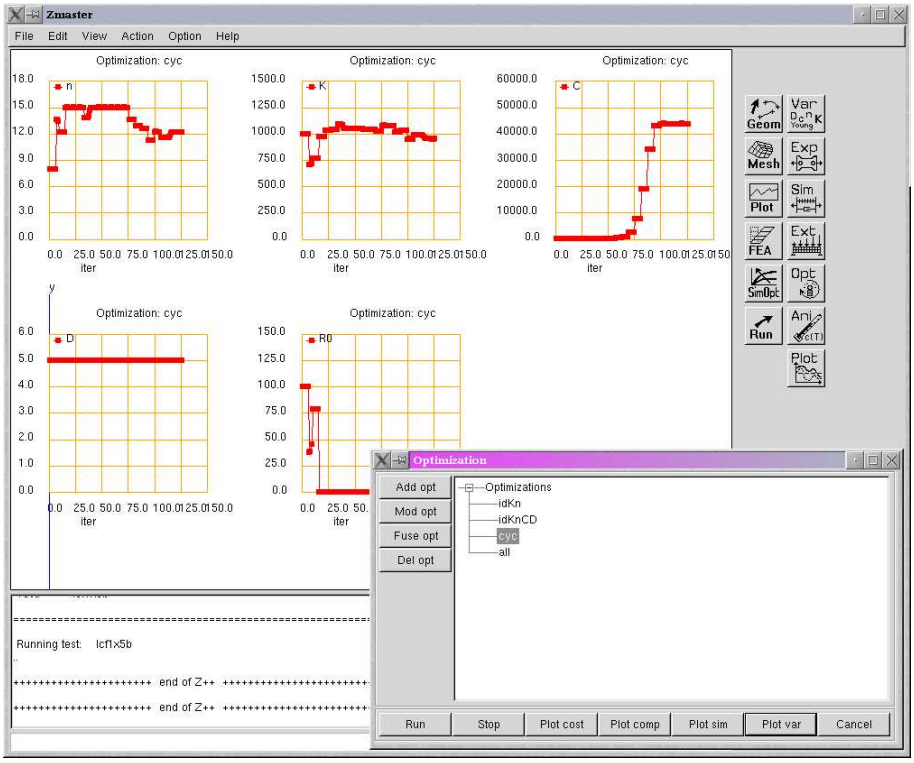


Plot sim:

This command draws all plots attached to the simulations involved in the selected optimization.

Plot var:

This command plots the evolution of the variables activated by the currently selected `optimization item`.

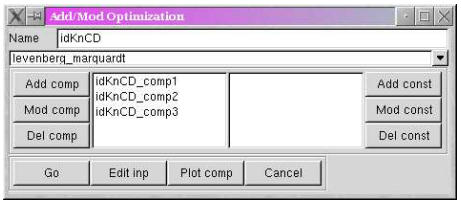


Cancel command:

This command closes the `Optimizations management dialog` without further modification to the database.

Add/Mod Optimization dialog

`Optimization items` are created or modified by means of the `Add/Mod optimization dialog` shown in the next figure.



The following sub-items can be modified :

- The name of the `Optimization item` defined in the `Name` textfield. A default name is given at creation, that should be modified to something more meaningful to the user.
- The optimization method that can be selected by means of the combo box at the top of the dialog. The following methods are available (see the Optimization Chapter of the Zmat manual for a complete reference) :
  - Levenberg-Marquardt : gradient-driven algorithm, particularly suitable to the minimization of least-squares type cost functions. This algorithm does not support optimization constraint.
  - Nelder-Mead : heuristic method that does not need gradient computation. The implementation supports optimization constraints and may be preferred to Levenberg-Marquardt when gradients calculated by perturbation are too imprecise or too costly to compute. Convergence may be slow however when the number of variables is increased.
  - Augmented-Lagrangian : this classic gradient-driven method should be used instead of Levenberg-Marquardt when optimization constraints are needed.
  - Evolution : a genetic algorithm implementation, that can be preferred to the previous methods to escape local optima. Note however that activating this method largely increases the number of iterations needed for convergence.
  - Single : a tool that performs a single optimization iteration, and is provided to verify the validity of input data.
- A list of `comparison items` (between experimental and simulation results), managed by means of the `Comparison items selection box`, and its associated commands.
- A list of optimization constraints managed by means of the `Constraint items selection box`, and its associated commands.

The following sub-commands can be activated from this dialog.

Add comp command:

This command opens the `Add/Mod comparison dialog` (page 6.10) to add a new `comparison item` in the definition of the optimization problem.

Mod comp command:

This command edits the `comparison item` selected in the `Comparison items selection box` and opens the `Add/Mod comparison dialog` (page 6.10) for modification.

Del comp command:

This command removes all the `comparison items` selected from the definition of the optimization problem.

Add const command:

This command opens the `Add/Mod constraint dialog` (page 6.14) to add a new `constraint item` in the definition of the optimization problem.

Mod const command:

This command edits the `constraint item` selected in the `Constraint items selection box` and opens the `Add/Mod constraint dialog` (page 6.14) for modification.

Del const command:

This command removes all the `constraint items` selected from the definition of the optimization problem.

Go command:

This command updates the database with the `optimization item` informations included in the dialog.

Edit inp command:

This command opens the Zopt input file generated automatically by the application from the informations defined in the dialog. The various convergence parameters allowed by the optimization method selected should be changed directly in the input file under the `***convergence section`. Default values are provided that may be adequate in the majority of cases, so this operation is seldom needed. The Optimization chapter of the Zmat manual describes the parameters available to finely tune the optimization algorithm. Note that other sections of the input file are automatically regenerated each time the `optimization item` is updated with the `Go command`, so there is no use to manually modify the commands included in those sections.

Plot comp command:

This command plots all the `comparison items` selected in the `Comparison items selection box`.

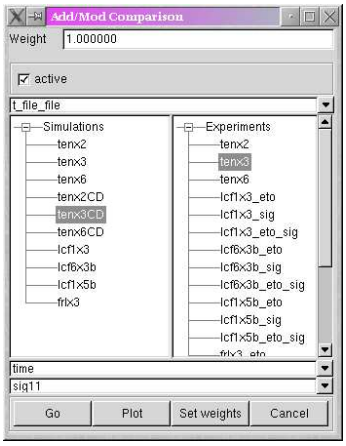
Cancel:

This command closes the `Add/Mod optimization dialog` without further modification to the database.

Add/Mod Comparison dialog

This dialog allows to add or modify a new `comparison item` in the definition of the optimization problem. A `comparison item` is built from two curves corresponding respectively to the experimental data and the simulation results. The least-square distance between those two curves is added to the cost function of the optimization problem.

The next figure shows the sub-items manipulated by this dialog.



The following sub-items can be modified :

- The `Weight` textfield may be used to modify the influence of the current comparison among all the comparisons of the selected optimization. Default value is 1. Filling the `Weight` textfield with a value greater than the default value will also increase the weight of the current comparison in the global cost function minimized by the optimizer.
- The `active` checkbox allows to deactivate the comparison without having to remove it from the definition of the optimization problem by the `Del comp command` (default value is *active*).
- The comparison method can be selected by means of the combo box. The following methods are available (see the Optimization Chapter of the Zmat manual for a complete reference) :
  - `t_file_file` : standard comparison method with a time-weighted averaging.
  - `g_file_file` : generalized-file-file method, based on point densities, where the user has to consider the distribution of comparison points in order to control the quality of error estimation.
  - `weight_file_file` : comparison method especially adapted to the `Set weights command`. This method takes into account the weights locally assigned to points by user, so `weight_file_file` must be selected to activate the `Set weights command`.
  - `scatter_file_file` : comparison method adapted to the particular cases for which experimental data have several points defined by the same abscissa.

The selection tree on the left is used to select the `simulation item` (within the ones defined beforehand by means of the `Simulations command` at page 4.2) used to generate the simulation curve. Selecting a `simulation item` has the effect to fill the two combo boxes at the bottom of the dialog with the names of the output components selected for this particular simulation (page 4.7). A particular component must then be chosen for the X and Y values by means of these combo boxes to complete the definition of the simulation curve.

The selection tree on the right is used to select the `experiment item` (within the ones defined beforehand by means of the `Experiments command` at page 3.2) that defines the experimental curve.

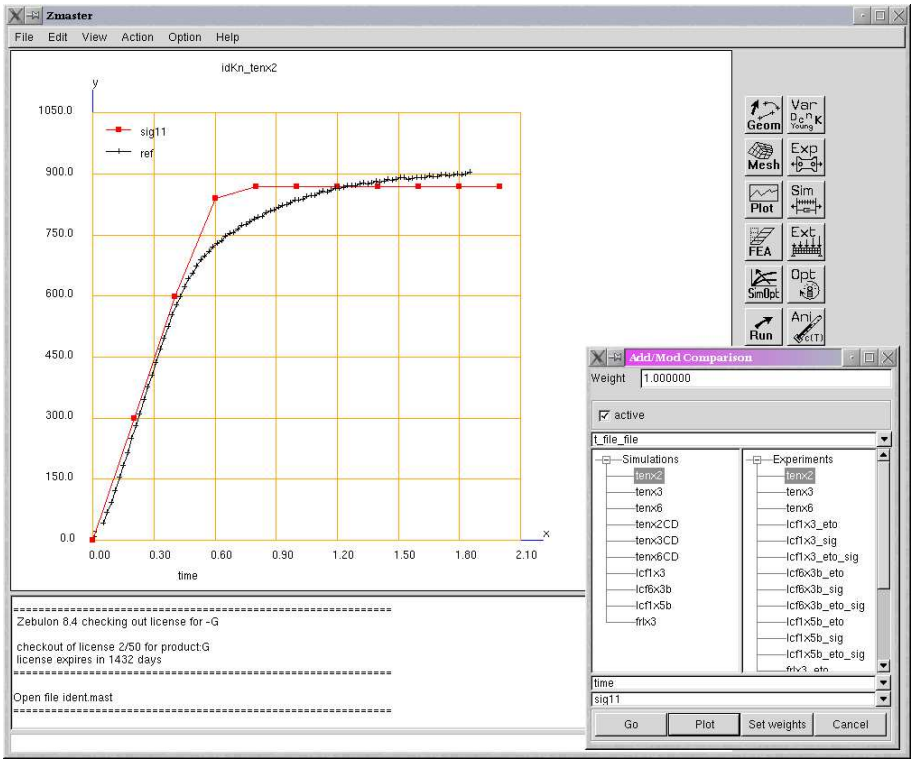
The following sub-commands may be activated from the dialog.

Go command:

This command updates the `comparison item` in the definition of the optimization problem with the informations included in the dialog.

Plot command:

This command draws on the same plot the experimental and simulation curves used to build the `comparison item`. The use of this command is strongly recommended to verify the validity of the comparison. Note however that the simulation should have been launched once beforehand for the corresponding simulation curve to be plotted.

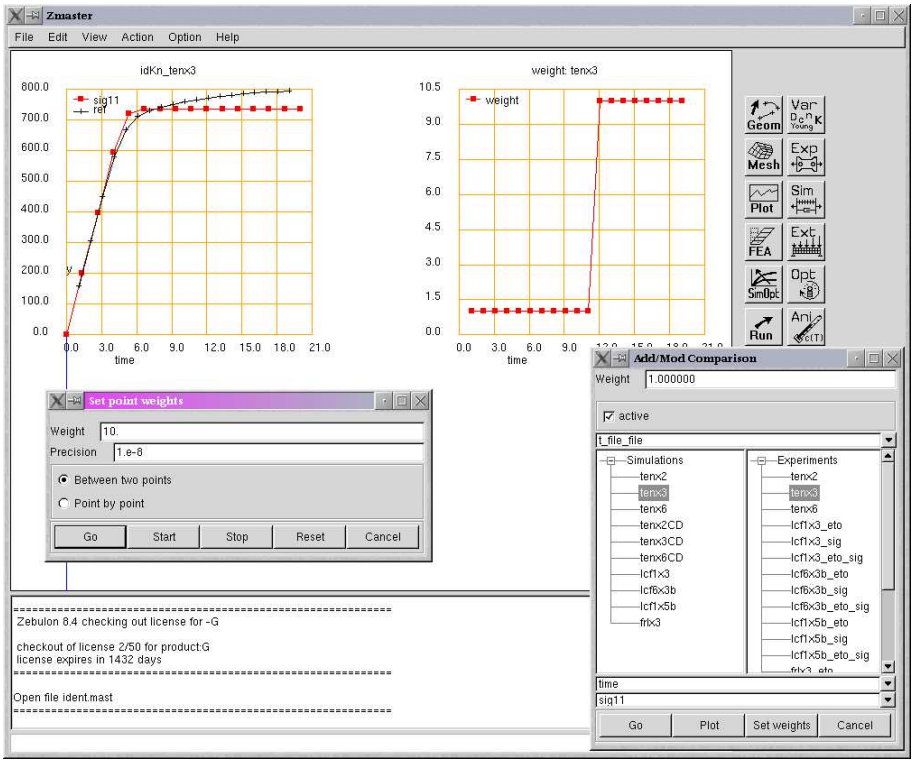


Set weights command:

This command opens the `Set point weights dialog` that allows to assign to each point

of the simulation curve a specific weight taken into account by the `weight_file_file` comparison method.

The next figure shows an example using the `Set point weights dialog`. The comparison curve is plotted on the left of the graphics area, and the weight curve on the right.



In the `Set points weights dialog` the following sub-items can be modified :

- The `Weight` textfield may be used to define the weight value assigned to the current point selection. Default value is 1.
- The `Precision` textfield defines the precision the points of the weight curve will be selected with. Default value is 10^{-8}.
- Two modes of selection are allowed, the particular type chosen being specified by means of the checkbox included in the dialog.
  - `Between two points` mode adds to the current point selection all the points located between the two selected points. This mode may be used to assign specific weight values to some regions of the comparison curve.
  - `Point by point` mode adds the particular selected point to the current point selection. This mode may be used to assign specific weight values to some critical points.

The following sub-commands may be activated from this dialog.

- The `Go` button updates the weight of the current point selection with the value given in the `weight` textfield.

- The `Start` button must be clicked to initialize the point selection, using the `Between two points` or `Point by point` selection mode. Points can then be added to the current point selection by clicking either on the comparison curve on the left or on the weight curve on the right of the graphics area.
- The `Stop` button must be clicked to close the point selection, using the `Between two points` or `Point by point` selection mode.
- The `Reset` button updates the considered `comparison item` in the definition of the optimization problem with the default values.
- The `Cancel` button closes the `Set point weights dialog` without further modification to the database.

Cancel command:

This command closes the `Add/Mod comparison dialog` without further modification to the database.

Add/Mod Constraint dialog

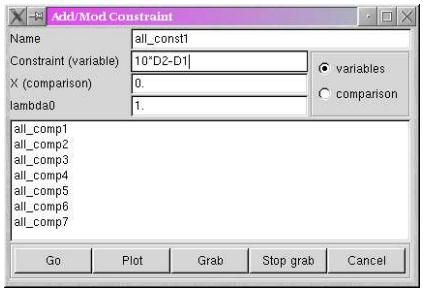
This dialog is used to create or modify optimization constraints. Constrained optimization problems have the following general form :

$$\begin{array}{ll} \text{Find } x_i^* \text{ such that :} & \\ F(x_i^*) = Min\ F(x_i)\ , & i = 1, ..., n_o \quad \text{the number of optimization variables} \\ g_j(x_i) \leq 0\ , & j = 1, ..., n_c \quad \text{the number of optimization constraints} \end{array}$$

Note that only inequality constraints $g_j(x_i) \leq 0$ are currently supported. In this framework, two types of constraints may be defined :

- `Variable Constraints` defined as functions of the optimization variables ;
- `Comparison Constraints` that can be used to specify that the simulation response should correspond exactly to the experimental curve at some critical points.

The dialog is illustrated in the following figure.



A name can be assigned to the `constraint item` in the `Name` textfield, while the `lambda0` textfield is used to specify the initial value of the Lagrange multiplier (or penalization parameter) assigned to the constraint (default value is 1.0).

Other parameters depend on the type of constraint selected, specified by means of the checkbox on the right of the dialog :

- `Variables Constraints`

  The function $g_j(x_i)$ of the optimization variables x_i should be defined in the `Constraint (variable)` textfield. Most mathematical functions of the `libc` library are allowed in the definition.

  Example:
  For a behavior with two nonlinear kinematic hardenings defined by parameters (C1,D1) and (C2,D2), one may use a constraint to specify that the saturation rate of the first mechanism (parameter D1) should be ten times higher than the second one (parameter D2). Two `Variable constraints` are necessary to enforce such an equality constraint :

```
10.0*D2 - D1
D1 - 10.0*D2
```

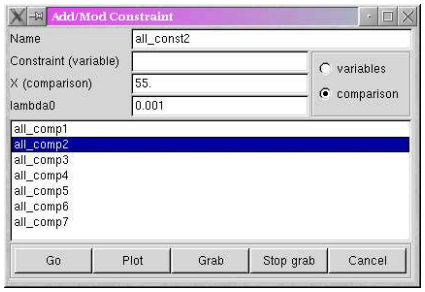
- Comparison Constraints

  Definition of such constraints is based upon comparison items described in the previous section (page 6.10). All comparisons added beforehand in the definition of the optimization problem are included in the dialog selection box, where the user can choose the particular item needed. The point for which the experimental and numerical response should be equal is specified by entering the corresponding coordinate along the X axis in the X (comparison) textfield. Note that deactivated comparisons (by means of the active checkfield as described at page 6.10) are allowed in the definition.

  The Grab button plots the comparison curves in the graphics area, where the constraint X coordinate can then be defined by a simple click on the region of interest. This mode can be closed by hitting the Stop grab button.

  Example:

  The dialog shown in the next figure illustrates the definition of such a comparison constraint.



The following sub-commands may be activated from this dialog.

Go command:

This command updates the constraint item in the definition of the optimization problem with the informations included in the dialog.

Plot command:

This command plots in the graphics area the curve of the selected comparison item.

Cancel command:

This command closes the Add/Mod Constraint dialog without further modification to the database.

Chapter 7

Anisothermal variables management

Anisothermal variables

The `Anisothermal variables command` represents the main contribution of this new 8.4 version.

In the proposed methodology, the material model is first calibrated on isothermal experiments using the previous Sim-Opt commands. In this context, an optimal set of coefficients is thus obtained for each temperature.

The first aim of the `Anisothermal variables command` is to provide tools allowing to verify the evolution of those coefficients on the whole range of temperature. Global functions, that allow to monitor the evolution with temperature of critical material properties (eg. R_{O2} yield stress) that may be expressed as a particular function of the material model coefficients can be used.

The second aim of the `Anisothermal variables command` is to simplify the anisothermal models identification method. The idea consists of creating anisothermal variables based on the current simulation variables, and then fitting those new variables to the experiments defined for different temperatures. For instance, if the user has launched five optimizations to fit eight simulations to eight experiments defined at five distinct temperatures, the user can use the `Anisothermal variables command` to transform the current simulation variables into temperature dependent simulation variables. Such an approach needs the creation of a template file including the temperature depend-ant simulation variables that will appear in the `Variables dialog`. The user can then launch one optimization on the temperature depend-ant simulation variables to fit at the same time the experiments defined at five distinct temperatures. One can consider that this optimization amounts to fit the temperature depend-ant simulation variables to the best values of the current simulation variables stored in the histories (typically one history per temperature value). The `Update sim command` allows finally to create, at any temperature chosen by the user, new simulations based on the temperature depend-ant simulation variables.

Such a proceedings may be decomposed into four steps :

- fit the set of optimization `variables` for each temperature value the solution of which is stored in an `history item` ;
- create `anisothermal variable items` from the coefficients optimized for each temperature value :
  - evolution of the coefficients versus temperature may be plotted ;
  - global functions definition from several `anisothermal variables` (for instance 0.2% yield stress) that allows to evaluate the model built from tabulated temperature values ;
  - for each `anisothermal variable`, definition of an analytical function the coefficients of which may be optimized versus the tabulated temperature values ;
- automatically add new items that contain the anisothermal behavior :
  - automatic creation of the template file corresponding to the chosen material file ;

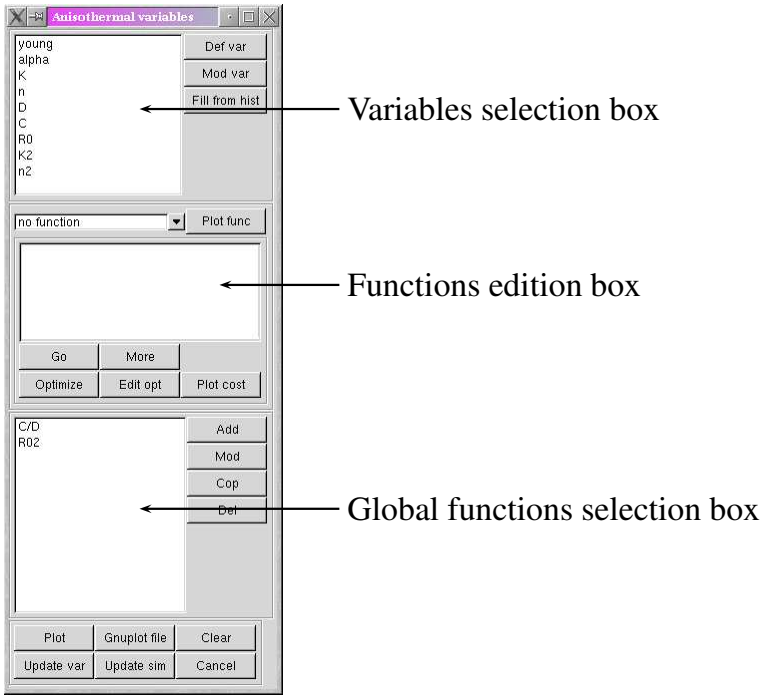
  - automatic creation of new optimization parameters that are the coefficients of the functions attached to the `anisothermal variables` ;
  - creation of isothermal `simulation items` that use the anisothermal material file ;

- execution of a whole optimization that implies the optimization parameters containing the anisothermal behavior.

Anisothermal variables management dialog

The dialog is represented hereafter. The dialog can be decomposed into four parts. The first frame includes the `anisothermal variables` and their definition based on experimental data, the second concerns the parametric functions used to work out the `anisothermal variables`, the third contains the global functions, built from several `anisothermal variables`, while the last includes the current command buttons.

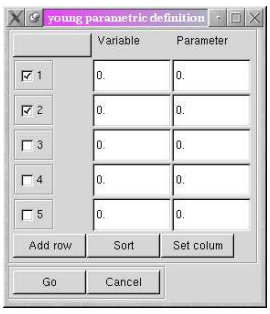
An `anisothermal variable` is created in the database from its corresponding `variable` when its definition versus the parameter values is done, its representative function is associated, and when the `Update var command` is clicked.



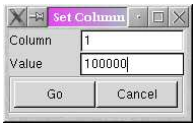
• At the top of this dialog, the first frame contains on the left a selection box that lists the `anisothermal variables` on which the sub-commands on the right can be activated, as described in the next pages. Note that the selection box is by default filled with all the `variables`.

Def var command:

This command allows to define the values of the `anisothermal variable` selected versus the parameter (temperature) values. By hitting the `Def var command`, the user opens a dialog shown in the next figure.



By default, `variable` and parameter values are stored into a table constituted by five rows, the first two being activated. Addition of new definition lines is still possible by means of the `Add row` button. Note that an added row is activated by default (see the checkbox at the beginning of each line). The `Sort` button allows to sort the lines according to the parameter column. If the user clicks on the `Go` button despite of a not sorted parameter column, an error message is printed. The `Set column` button may be used to set a whole column to a constant value, as shown in the next figure.



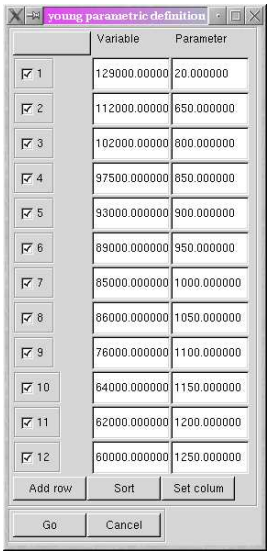
The `Go` button updates the definition of the selected `anisothermal variable` with the informations included in the dialog, while the `Cancel` button closes the `parametric definition dialog` without further modification to the definition of the selected `anisothermal variable`.

Mod var command:

This command allows to modify the definition of the `anisothermal variable` selected. If the user tries to modify an `anisothermal variable` not previously defined, an error message is printed. Note that this command opens the `parametric definition dialog` corresponding to the selected `anisothermal variable`.

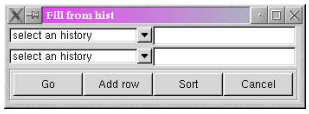
Example:

The dialog shown in the next figure illustrates the definition of such an `anisothermal variable`.



Fill from hist command:

The aim of the `Fill from hist command` is to define at the same time all the `variables` selected in the `Variables selection box`, by using the best values of the current simulation variables stored in the histories (typically one history per temperature value). When the user clicks on this command, a warning message is printed to recall that the use of the `Fill from hist dialog` will modify the parametric `variables` already defined. By hitting the `Fill from hist command`, the user opens a dialog shown in the next figure.



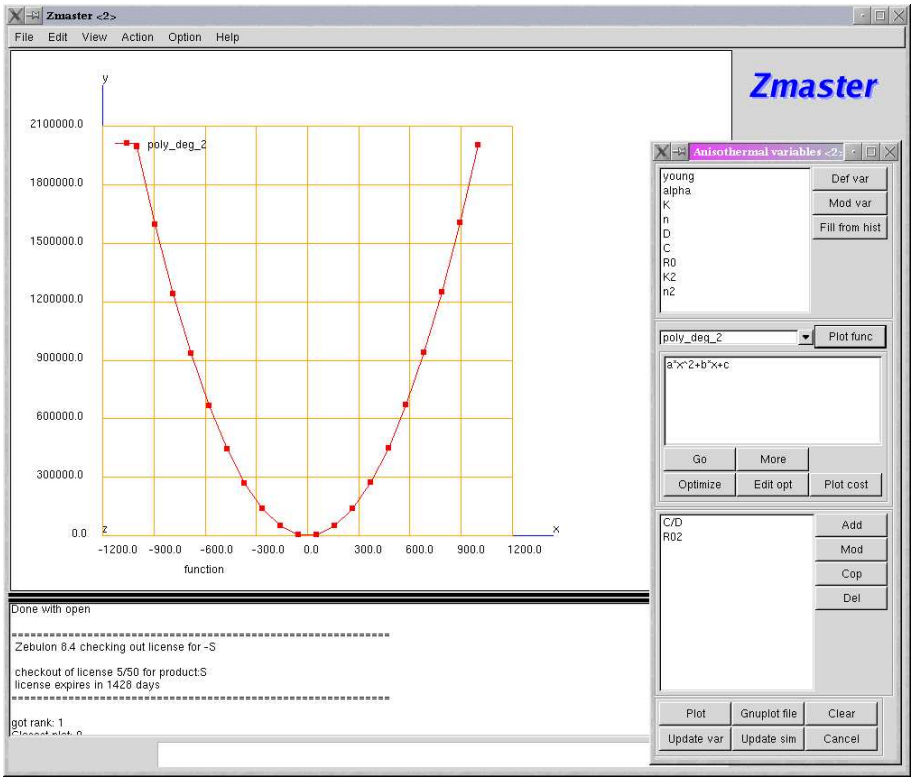
The combo boxes of the left column allows to select an history among the whole `history items` list. The user is then supposed to associate the selected `history item` to a parameter value in the right column. Note that all the `variables` implied by the selected `history item` will be associated to the parameter value, while the others (listed in the `anisothermal variables selection box` but not included in the selected `history item`) will be associated to the parameter value too, but with their own value reduced to null. The `Go` button updates the definition of all the `anisothermal variables` with the informations included in the dialog. Addition of new definition lines is still possible by means of the `Add row` button. The `Sort` button allows to sort the lines according to the parameter column. If the user clicks on the `Go` button despite of a not sorted parameter column, an error message is printed. The `Cancel` button closes the `Fill from hist dialog` without further modification to the definition of the `anisothermal variables`.

• The second frame allows to define a -or to use a predefined- function to fit the evolution of the selected `anisothermal variable` with temperature. At the top of the second frame, a combo box may be used to select a function among a list stored in the `functions_data_base` file. When the user selects a function, its expression appears in the textfield.

Plot func command:

This command allows to plot the function selected in the `functions_data_base` file. Note that only the functions of the `functions_data_base` file can be plotted by hitting the `Plot func command`. If the combo box displays the `no function` or the `user's function` option, an error message is printed if the user clicks on the `Plot func command`.

The dialog shown in the next figure illustrates the plot of a function stored in the `functions_data_base` file.



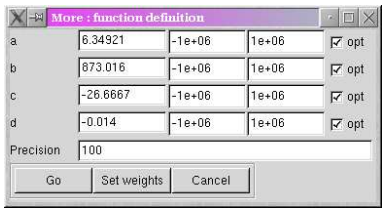
The textfield may be used to build user's function not obviously based on functions included in the `functions_data_base` file.

Go command:

This command assigns to the (previously defined) `anisothermal variable` selected the function the expression of which is written in the textfield.

More command:

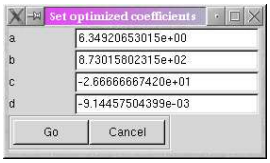
This command allows to define or modify the current and min/max values of the parameters of the parametric function assigned to the selected `anisothermal variable`. By hitting the `More command`, the user opens the `More : function definition dialog` shown in the next figure.



This dialog is based on a table that lists the parameters of the parametric function assigned to the selected `anisothermal variable`, with their current and min/max values, and a checkbox to activate or deactivate their optimization. The `Precision` textfield corresponds to the number of points the parametric function curve will be constituted of. Note that the parametric function curve will besides include all the parameter values contained in the definition of the selected `anisothermal variable`. The `Go` button updates the definition of the parametric function of the selected `anisothermal variable` with the informations included in the dialog. The `Set weights` button opens the `Set points weight dialog` described in a previous section (page 6.11). The `Cancel` button closes the `More : function definition dialog` without further modification to the definition of the parametric function of the selected `anisothermal variables`.

Optimize command:

This command launches the Zopt module on the checked parameters to fit the parametric function with the `anisothermal variable` experimental data. When the optimization terminates because of convergence, a window is opened showing the best values found by the optimizer, as shown in the next figure.



The `Go` button must be clicked if the user does want to update the checked parameters values in the definition of the parametric function with the optimized values. Otherwise, the checked parameters values are left unchanged.

Edit opt command:

This command opens the Zopt input file automatically generated by the application from the informations defined in the dialog. The various convergence parameters allowed by the optimization method selected should be changed directly in the input file under the `***convergence section`. Default values are provided that may be adequate in the majority of cases, so this operation is seldom needed. The Optimization chapter of the Zmat manual describes the parameters available to finely tune the optimization algorithm. Note that modifications of the automatically generated input file will be taken into account if the user hits the `Save command`.

Plot cost command:

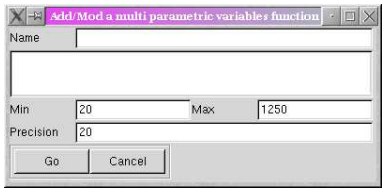
This command plots the evolution of the cost function associated with comparisons between the parametric function and the `anisothermal variable` experimental data during

optimization iterations. The command can be activated to check the convergence while the optimizer is currently running (page 6.4).

• On the left of the third frame, a selection box lists the global functions, built from several `anisothermal variables`, on which the sub-commands on the right can be activated, as described in the next pages.

Add command:

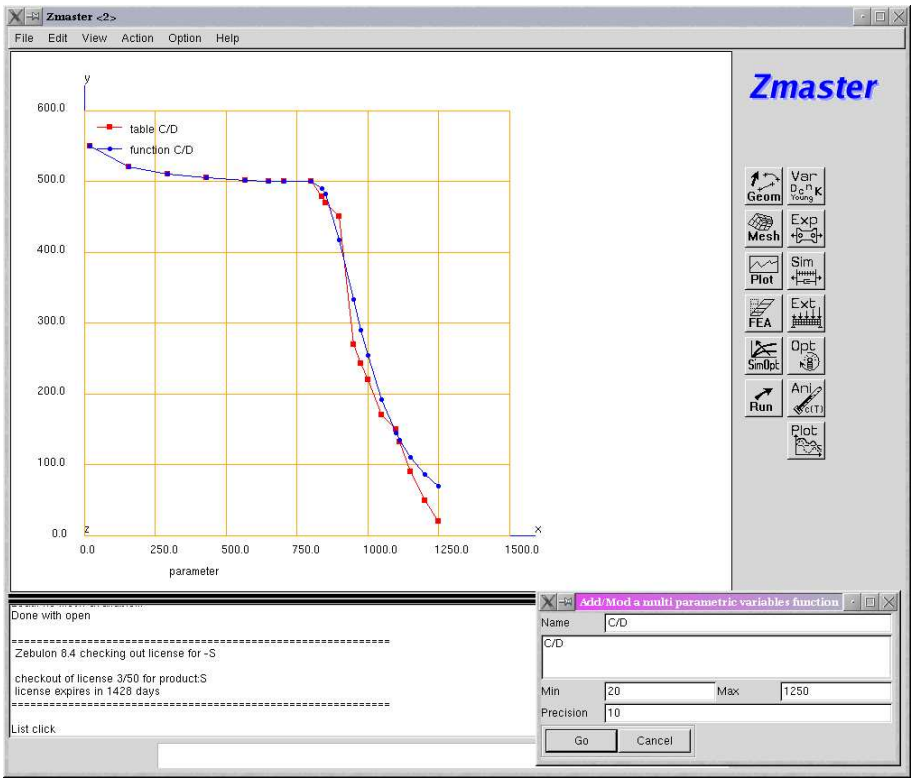
This command may be used to add a new global function. By hitting this command, the user opens the `Add/Mod a multi parametric variables function dialog` shown in the next figure.



The `Name` textfield allows to define the name of the global function. Note that no default name is given at creation. The edition textfield is used to write the expression of the global function. The `Min` and `Max` textfields define the min/max X values of the global function curve. Note that the `Min` and `Max` textfields are respectively filled by default with the min/max values of the parameter in the definition of the `anisothermal variables` implied in the global function. The `Precision` textfield corresponds to the number of points the global function curve will be constituted of. Note that the global function curve will besides include all the parameter values contained in the definition of the `anisothermal variables` implied in its definition. The `Go` button updates the definition of the global function with the informations included in the dialog, while the `Cancel` button closes the dialog without further modification to the definition of the global function.

Example:

The next figure illustrates the definition of such a global function.



Mod command:

This command edits the global function selected. The `Add/Mod a multi parametric variables function dialog` is opened allowing to change the definition of the selected global function (page 7.9).

Cop command:

This command creates a new global function in the `Global function selection box`, by copying the informations contained in the selected global function. The new global function can then be edited by means of the `Mod command`.

Del command:

This command deletes from the database the global functions selected in the `Global function selection box`.

• The fourth frame contains the following sub-commands.

Plot command:

This command plots at the same time in the graphics area the curves representing the `anisothermal variables` (with or without defined parametric function) selected in the `Anisothermal variable selection box`, and the ones corresponding to the global functions selected in the `Global function selection box`.

Gnuplot file command:

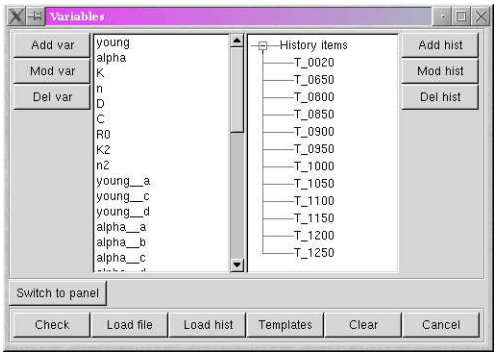
This command creates the gnuplot file corresponding to the curve associated to the `anisothermal variables` (with or without defined parametric function) selected in the `Anisothermal variable selection box`, or to the global functions selected in the `Global function selection box`. Note that an error message is printed if the user clicks on the `Gnuplot file command` while several items are selected. The created gnuplot file, named `aniso.gnuplot`, is located in the current directory.

Clear command:

This command clears at the same time the selections in the `Anisothermal variable selection box` and in the `Global function selection box`.

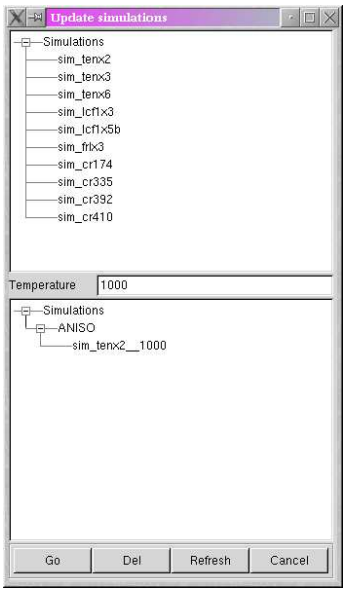
Update var command:

This command updates the `variable` database by adding as new `variables` all the coefficients of the parametric functions attached to the `anisothermal variables` defined in the `Anisothermal variables dialog`. By hitting the `Update var command`, the `Variables Creation mode dialog` (for instance) will display the next time it will be opened the anisothermal parameters in the `Variables selection box`, as shown as example in the next figure.



Update sim command:

This command may be used to create new `simulation items`. By hitting this command, the user opens the `Update simulations dialog` shown in the next figure.



This dialog contains at the top a `Simulation items selection tree` that lists all the `simulation items` created by means of the `Simulation dialog`. In the middle of the dialog, the `temperature` textfield allows to define a temperature value for which the user wants to create a new `simulation item` copied from the one selected in the upper `Simulation items selection tree`. The just created `simulation item` is then displayed in the `Simulation items selection tree` at the bottom of the dialog. Note that this second `Simulation items selection tree` lists only the `simulation items` created by means of this dialog, located in the `ANISO` directory itself located in the current directory. The `Go` button creates the new `simulation item` from the one selected in the upper `Simulation items selection tree`, conjugated with the value written in the `temperature` textfield, and updates the database. The `Del` button deletes from the database the `simulation items` selected in the second `Simulation items selection tree`. The `Refresh` button updates the two `Simulation items selection trees` with the informations contained in the `Simulation dialog`, while the `Cancel` button closes the `Update simulations dialog` without further modification to the database.

Cancel command:

This command closes the `Anisothermal variables dialog` without further modification to the database.

Chapter 8

Multiplot management

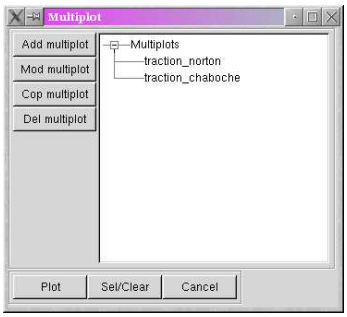
Multiplot

In our definition, a multiplot corresponds to a list of several curves. The `Multiplot command` allows to create, delete and modify `multiplot items`. If each module of the SimOpt Plugin proposes to plot results in a default format, `Multiplot command` represents for user a more complete tool that allows for example to plot several curves on the same graphic, to configure the frame parameters, or to save several curves to be used in a Gnuplot file.

Note that the `Multiplot dialog` provides graphic tools built to facilitate the plot treatments.

Multiplot management dialog

The main dialog is shown in the following figure. The selection tree allows to select `multiplot items` to apply appropriate commands on.



Add multiplot command:

This command may be used to create a new `multiplot item` in the database by opening the `Add/Mod Multiplot dialog` (page 8.4).

Mod multiplot command:

This command edits the `multiplot item` selected. The `Add/Mod Multiplot dialog` is opened allowing to change the definition of the item (page 8.4).

Cop multiplot command:

This command creates a new `multiplot item`, by copying the informations contained in the selected `multiplot item`. The new item can then be edited by means of the `Mod multiplot command`.

Del multiplot command:

This command deletes from the database all the `multiplot items` selected in the `Multiplot items selection tree`.

Plot command:

This command plots at the same time in the graphics area the curves representing the `multiplot items` selected in the `Multiplot items selection tree`.

Sel/Clear command:

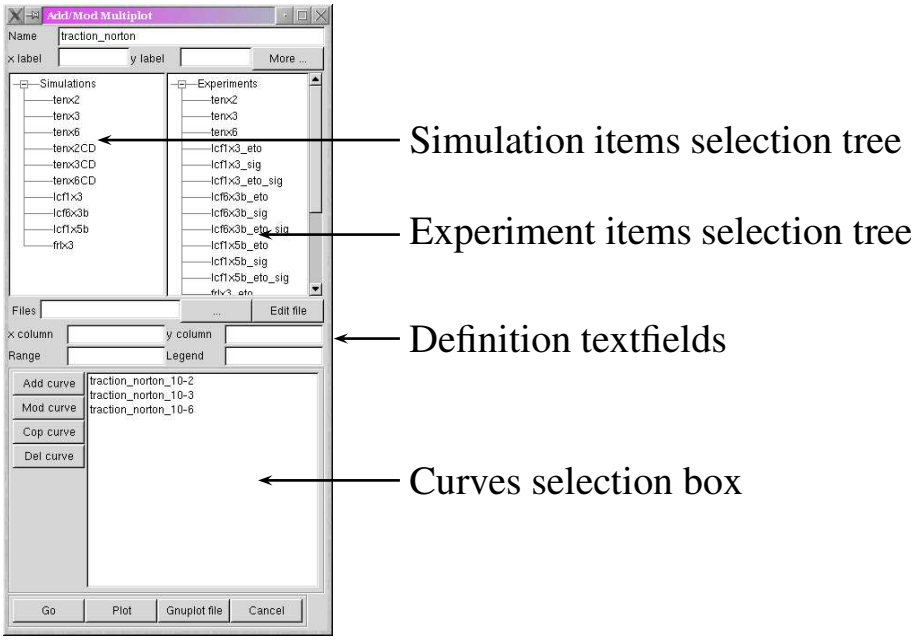
This command clears selections in the `Multiplot items selection tree`.

Cancel command:

This command closes the `Multiplot dialog` without further modification to the database.

Add/Mod Multiplot dialog

This dialog, used to create or modify `multiplot items` that will be plotted, is shown in the next figure.



The proceedings of this dialog consists essentially of :

- firstly, filling the `Files` textfield by selecting a file (containing simulation output or experimental data), not obviously in the `Simulations` or in the `Experiments` selection trees ;
- secondly, defining the number of the columns of the selected file, plotted in X and Y axis and respectively written in the `x column` and `y column` textfields ;
- finally, clicking on `Add curve` button to create a new curve of the current `multiplot item`.

If user selects a file in the `Simulations` or in the `Experiments` selection trees in the upper part of the dialog, the `Files` textfield in the middle part of the dialog will be automatically filled.

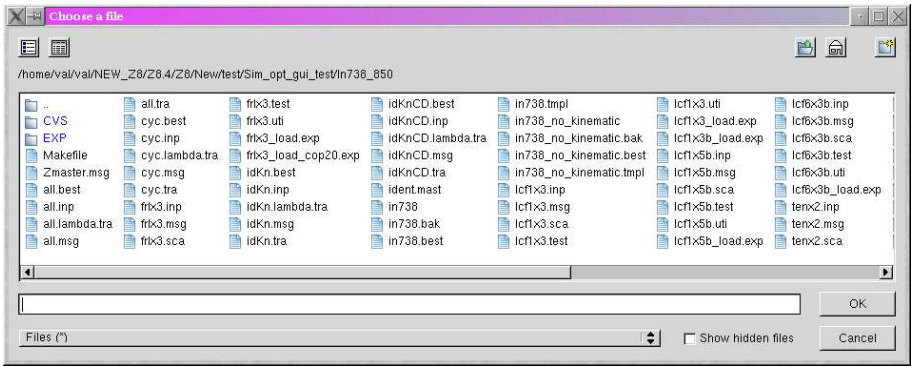
If user wants to modify a defined curve, user has to select the curve in the `Curves selection box` (the selected curve parameters then appear in the textfields in the middle part of the dialog), to modify parameters, and to click on the `Mod curve` button.

The name of the `multiplot item` can be modified in the `Name` textfield. The default name assigned at creation should be changed to something more meaningful to the user. In the graphics area, the name will appeared as the title of the plot.

The `x label` and `y label` textfields allows to respectively define the abscissa and ordinate labels.

The `Simulation items selection tree` on the left of the dialog and the `Experiment items selection tree` (which gathers the `base`, the `derived` and the `composite experiment items`) on the right allows to select the item to apply appropriate commands on.

The `Files` textfield must be filled by the user to select the file that contains the informations to plot. Three different ways allow to select the file. The first one consists of clicking on an item in the upper selection trees ; the name of the item then appears in the `Files` textfield. The user may directly write the name of the file. Otherwise, the browse button available on the right of the textfield opens a file selection dialog that helps selecting an existing file as shown in the next figure.



The `x column` and `y column` textfields must be filled by the user to indicate the number of the columns of the selected file that will be respectively used for the plot in abscissa and ordinate. Note that syntax is for instance *$1*.

The `Range` textfield allows to define a function the `variables` of which are the columns selected in the `x column` and `y column` textfields, to restrict the plot range.

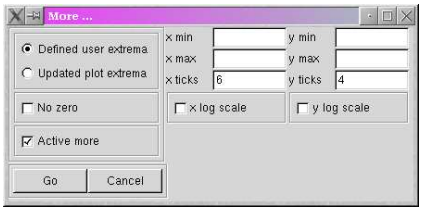
The `Legend` textfield may be used to define the legend of each curve included by the considered `multiplot item`.

At the bottom on the right in the dialog, the `Curves selection box` allows to select the `curve item` to apply appropriate sub-commands on. Note that when created, a new `curve item` appears in the `Curves selection box`.

The following sub-commands may be activated from this dialog.

More command:

This command can be used to adjust the graphical display parameters to user's needs. By activating the `More command`, the user opens the `More dialog` shown in the next figure.



Two modes of extrema management are allowed, the particular type chosen being specified by means of the checkbox included at the top on the left in the dialog. If the `Define user`

extrema mode is activated, the plot extrema that will be taken into account are those defined by the user in the x min, x max, y min and y max textfields. Otherwise, if the Update plot extrema mode is activated, the plot extrema that will be taken into account are those read in the database for the selected curves.

The x ticks and y ticks can be used to define the number of ticks the user wants respectively on the abscissa and ordinate axis. Default values are respectively 6 and 4.

The No zero checkbox can be used to specify if the origin of the curve, when not available in the actual data, should be included when drawing the plot. Default is off.

The x log scale and y log scale checkboxes can be used to respectively plot the curve with a logarithmic abscissa and ordinate. Default is off.

The Active more checkbox can be used to activate all the informations included in the More dialog. Default is on.

The Go button updates the graphical display parameters of the multiplot item with the informations included in the More dialog.

The Cancel button closes the dialog without further modification to the graphical display parameters of the multiplot item.

Edit file command:

This command edits the selected file specified in the Files textfield. Note that the Edit file dialog does not allow to modify the selected file, but it may be used to know how to fill the x column and y column textfields.

Add curve command:

When an item is selected in the Simulation or Experiment items selection tree, and the previously described textfields are filled, the Add curve command allows to create a new curve item with the informations included in the dialog.

Mod curve command:

This command may be used to modify a curve item already created. The proceedings consists of selecting a curve item in the Curve items selection box, modifying its parameters in the previously described textfields, and then hitting the Mod curve command.

Cop curve command:

This command creates a new curve item, by copying the informations contained in the selected curve item. To edit the new curve item, the user has only to select it since its parameters then appear in the previously described textfields.

Del curve command:

This command deletes from the database the curve item selected in the Curve items selection tree.

Go command:

This command updates the database with the multiplot item informations included in the dialog.

Plot command:

This command plots in the graphics area the curves representing the considered `multiplot item`.

Gnuplot file command:

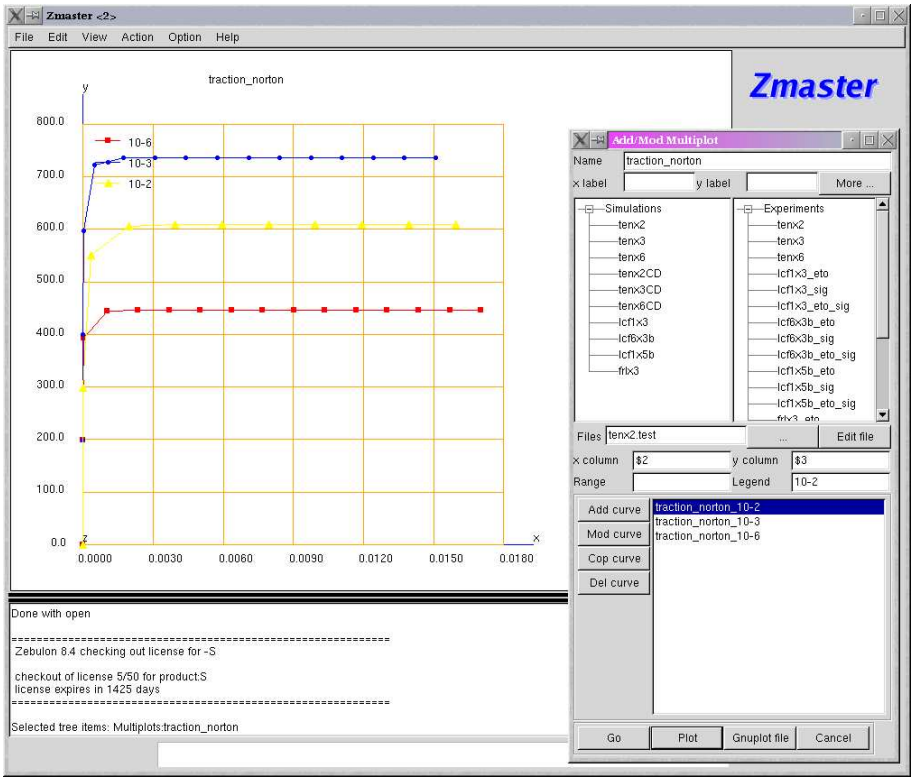
This command creates the gnuplot file corresponding to the curves representing the considered `multiplot item`. Note that the created gnuplot file is located in the current directory.

Cancel command:

This command closes the `Add/Mod Multiplot dialog` without further modification to the database.

Example:

The next figure illustrates the definition of such a `multiplot item`.



Chapter 9

Index

Index