

Contents

1. Annotation
2. Components
3. Start
 - 3.1. Function arguments
 - 3.2. Function call
 - 3.3. Function workspace
 - 3.4. Prepare input file
4. Graphical user interface (GUI)
5. Tips
6. Known Problems and possible Solutions

1. Annotation

The program Maler is developed for comfortable visualization and studying the multi-parameter one- and two-dimensional functions in the Matlab environment. The functions may be specified with or without using Matlab Symbolic Math toolbox.

2. Components

The Maler working directory includes:

- i. The subdirectory <auxiliary>, containing gif-images for GUI button icons, the file <maler_callspec_25.txt> with RGB color specifiers for multi-line plots, the file <maler_defaults.in> with default settings, the template script <maler_userbutt_callbacks.m> for user-defined button callbacks, and this manual <maler_user_guide.pdf>.
- ii. The subdirectory <ex> with three examples of the input file: <example_1d.m>, <example_2d.m> and <example_txt.in>. It contains also three corresponding output subdirectories.
- iii. The default output subdirectory <out>.
- iv. The Matlab function *maler* in file <maler.m>

3. Start

3.1 Function arguments

The function *maler* allows three optional arguments:

- The input file name (may include path), string, default = ''.
- The name of output directory (may include path), string, default = 'out'.

- The symbolic calculations switch, logical. Default value is 1, if symbolic toolbox is installed, or 0, in opposite case.

3.2 Function call

The function *maler* may be called in two ways:

- 3.2.1. Call without arguments, *maler()*. In this case, *maler* tries to export base workspace functions, parameters and variables or suggests input dialog boxes.
- 3.2.2. Call with input file, *maler('input_file_name', ['output directory'], [switch])*. Three types of the input file are allowed: text files, Matlab .m-files, and Matlab .mat-files.

3.3 Function workspace

The function *maler* reserves two names of the workspace variables.

The variable <WORKSPACE> (structure) is created in the base workspace (if not exist). It contains basic problem settings: names of variables, parameters and functions, function definitions, current values of parameters, and computational grid settings.

The variable <MALER> (structure) is created in the function *maler* workspace. It contains GUI settings, numerical grid, and current function values. When GUI window is closed, variable <MALER> is transferred to the base workspace.

3.4 Prepare input file

To avoid keyboard input, make sure that your input file contains the following four entries:

- WORKSPACE.VARIABLES = { 'var_name_1' , ['var_name_2'] };
- WORKSPACE.GRIDPARAM = { [var1_min_value , var1_max_value , var1_length] ,
[the same for 2nd variable if any] };
- WORKSPACE.PARAMETERS = { 'param_name_1' , 'param_name_2' , ... };
- WORKSPACE.PARSAMPVAL = { param_1_sample_value , param_2_sample_value , ... };

The cell array <WORKSPACE.PARAMETERS> may be empty. In this case the fourth entry <WORKSPACE.PARSAMPVAL > may be omitted.

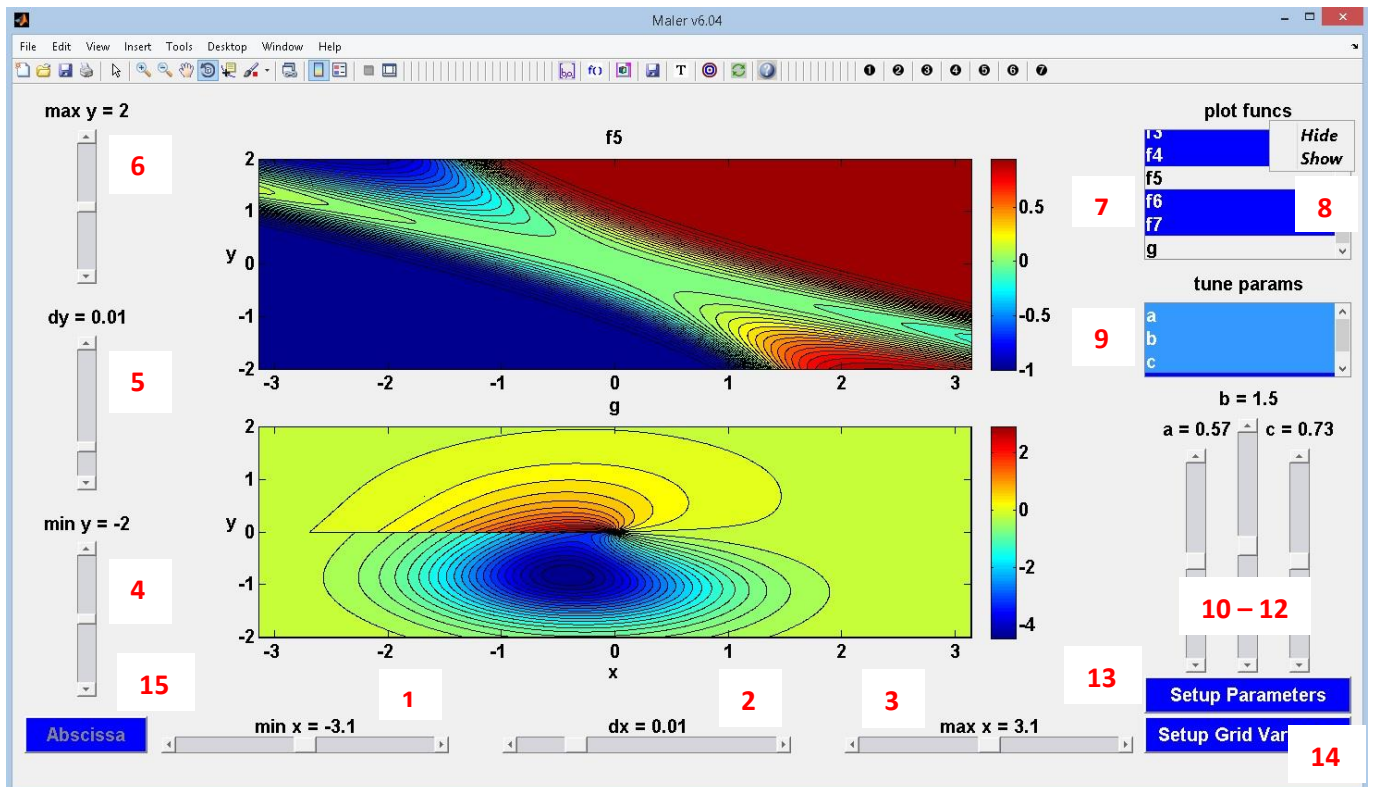
If cell array <WORKSPACE.VARIABLES> is empty, the program execution is terminated. If it contains more than two elements, excessive variables are treated as parameters.

If some of these four definitions are not provided in the input file or if they are not self-consistent, input dialog boxes appear to fix a mess.

Keyboard input is also requested if input file does not contain any function definition.





The examples of input files are found in the directory <ex>.

4. GUI



Graphical user interface represents the standard Matlab figure, complemented with a number of extra interactive elements.


- **Sliders 1, 2, 3** in the bottom control the minimal value of the first independent variable, grid step, and maximal value, respectively. The analogous **sliders 4 – 6** at the left are available for the second variable if any.
- **Listbox 7** <plot funcs> on the right allows selecting function(s) to be plotted. For multiple selections, hold <Ctrl> or <Shift> keyboard buttons.
- **Text label 8** <plot funcs> over the listbox 7 is interactive. The right-click opens context menu { 'Hide' , 'Show' }, which allows to hide some functions from the listbox 7, or return them back in a scope.
- **Listbox 9** <tune params> allows selecting the parameters (up to 3), for which tuning sliders are created.
- **Sliders 10 – 12** for parameters, selected in listbox 9.
- **Pushbutton 13** <Setup Parameters> calls input dialog box for setting parameters' values and corresponding slider ranges.
- **Pushbutton 14** <Setup Grid Variables> calls input dialog box for setting grid parameters and corresponding slider ranges.




- **Pushbutton 15 <Abscissa>** is enabled for one-dimensional problems. It allows changing the plot abscissa to any workspace functions. This button is also enabled for two-dimensional problems, when some one-dimensional cross-section is under consideration (see below).
-  **Figure Menu button <Add/Remove Parameter>** calls input dialog box to enter new parameter. To remove existing parameter, enter its name.
-  **Figure Menu button <Add/Change/Remove Function>** calls cyclically the input dialog box to enter new function, or redefine existing one. To remove existing function 'f', enter 'f=' with empty right-hand side. To break input cycle, provide an empty input by pressing <OK> or <CANCEL>.
-  **Figure Menu button <Plot Type>** (enabled for two-dimensional problems only) calls context menu, allowing selection of one of the following plot functions: {*contour*, *contourf*, *mesh*, *surf*}. When more than one function is figured, selections {*contour*, *contourf*, *mesh*} produce individual subplots for each function, while *surf* shows them all in one plot. The selection <cross-section> calls input dialog box to specify the cross-section in the following form: 'a = f(b)', where 'a' is the name of an independent variable, 'b' is the name of another variable, and 'f' is any function.
-  **Figure Menu button <Save>** calls context menu with three items: {'Figure', 'Problem', 'Workspace'}.

Item **'Figure'** calls another menu with two options: {'Save', 'Specify format'}. Figures may be saved in a number of formats, such as {'fig', 'eps', 'jpg', 'png'} and others (see Matlab help on the *print* function). To specify several formats, separate them by comma, as above. Saving in the 'fig' format is performed in any case independently on the user choice. Figures are saved in the output directory, passed as a second argument of the function *maler*.

Item **'Problem'** creates the input file in the output directory. If symbolic calculations are enabled (third argument of *maler*), the .m-file is created. In opposite case, .txt-file is written.

Item **'Workspace'** saves the *maler* workspace in the .mat-file.

- **T Figure Menu button <Reset Font Size>** allows resetting GUI font size to any integer value in the range [6, 32].
-  **Figure Menu button <Reset Sliders Display Accuracy>** allows resetting the number of valuable digits displayed in slider labels in the range [1, 12]. It does not affect the computational accuracy. For a high display accuracy reduction of the GUI font size may be required.

-  **Figure Menu button <Reset Default Settings>** calls procedure, writing the current settings {Font Size, Slider Display Accuracy, Save Figure Formats, and Figure Position} in the file <auxiliary/maler_defaults.in> to make them defaults.
-  **Figure Menu button <Open User Guide>** opens this manual on Windows platforms.
-  **User-defined Figure Menu buttons <1 – 7>**. Callbacks for these buttons are to be specified in the template file <auxiliary/maler_userbutt_callbacks.m>. The number of these buttons (in the range [0, 7]) and the buttons' tooltip strings are to be specified in <auxiliary/maler_defaults.in>.

5. Tips

1. For the first use, it is recommended to start with running the provided examples, such as:
<clear all, maler('ex/example_1d.m')>, and play around with GUI elements to get accustomed with interface. Adjust fonts, figure position and figure saving formats, then press the button <Reset Default Settings> to make your settings default.
2. To get access to the *maler* workspace from the base workspace, declare global variable <MALER> there.
3. The GUI 'Save Figure' command creates a copy of the current figure without GUI elements. After saving, the duplicate figure is closed. Empty entry in the <Save> → <Figure> → <Specify Format> input dialog box enforces saving the figure in a .fig-format without closing the duplicate figure.
4. For Windows operation system, the figure saving format 'cpbd' is available, enforcing copy to the clipboard. This copy may be pasted in other application, such as Microsoft Word, by pressing Ctrl+V. Format 'pdf' is also available at Windows platforms.
5. To get small-size picture in a screen resolution, specify 'jpg' format. For all other formats printer is set to use the 300 dpi resolution.
6. When multiple-subplot figure is saved, subplot labels in the saved pictures may overlap each other. In this case reduce GUI font size and repeat saving.

6. Known problems and possible Solutions

1. If two or more copies of the function *maler* are called, the handles of all preceding *maler* figures are overwritten. This makes impossible to close these figures in usual way. To close them, type in Matlab command window *<close all force>*.
2. When multi-subplot figure is drawing, the *colorbar* command may return error, or color bars in some subplots may be lost. Possible solution: get the current Matlab release by printing in Matlab command window *<version -release>*. Open file *<maler.m>* and go to line 151. Add your release in the cell array *<MALER.uset.specifReleases>*. Then save file and restart *maler*. If it does not help, contact developer.
3. If some auxiliary files are corrupted, delete them. They will be automatically generated on the next call of the function *maler* (except this guide).

Copyright: Daniil Korovinskiy, 2017.

E-Mail: daniil.korovinskiy@gmail.com