Fitmodel User Manual

Fitmodel by
Igor Kulikov and Yannis Kalaidzidis
1998-2013

Manual by
Lars Opherden and Yury Bodrov
Last Update: January 7, 2014

Contents

1 Introduction

FitModel is a program designed to fit experimental data to a model. The model has to be provided as a system of ordinary differential equations which will be solved and then fit to the data.
FitModel was developed with the PLUK language. The first version was released in 1998 by Igor Kulikov under the supervision of Yannis Kalaidzidis who continued the development of this program until today.

2 User Interface

2.1 Overview

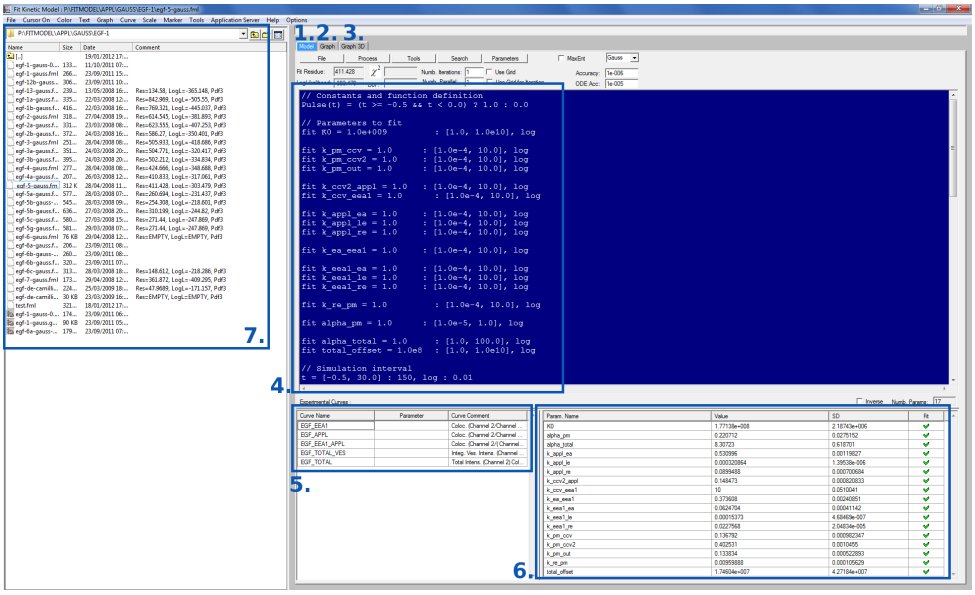


Figure 1: Screenshot of the FitModel program.

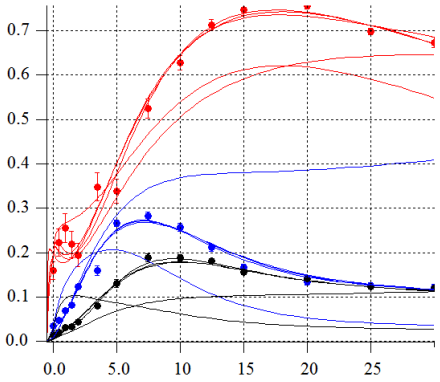
There are 3 tabs in FitModel as well as an conventional File Manager (7.) on the left side.

2.1.1 Model Tab (1)

This is the main tab. It is divided in 3 windows.

- The **Model Window (4)** contains all the code for the model. See section 3 for further reference.
- The **Experimental Data Window (5)** maps data in the model to real data sets.
- Finally the **Parameter Window (6)** contains a list of all parameters of the model and their current values and standard deviations. There is a check box to select if the parameter can be modified to fit the model.

This tab displays all graphs specified in the model. One example is shown in figure in the right.
The graphs provide the same functionality like scaling, re-naming, etc as MotionTracking graphs.
For further information please check the MotionTracking manual.



2.1.2 Graph Tab (2)

2.1.3 Graph 3D Tab (3)

This tab will display the 3D Graph of the model if selected.

2.2 Menus

There are two menus in Fitmodel. The top one is similar to MotionTracking's Graph Window. The lower one is specific to FitModel.

2.2.1 File

This submenu gives you access to the **Load**, **Save** and **Save As** functions to save or load the model. Models will be saved in *.fml* format. Additionally there is the option to **Export** the model as plaintext (*.txt*). This will copy all the code you see in the model window into the text file. Finally there is the option to print the model as text.

2.2.2 Process

- The **Fit Model** button will start the calculation. This may take a very long time. Note that if the process is cancelled the calculation will be stopped but the values which are already calculated will be available for further use.
- **Simulate Model** seems to have no function at the moment.
- **Draw Model** draws all selected Graphs in the Graph tab.
- **Parse Model** puts all the calculated parameter values into the Parameter Window.
- **Reset** resets all values in the Parameter Window and all calculations.

2.2.3 Tools

Tools → Show Program Source: This will open a new window which will show the system of equations as PLUK source code so it can be copied and pasted.

2.2.4 Search

This submenu contains the **Find**, **Next** and **Replace** buttons. They work similar to most other applications. **Find** can be opened by ALT+F3, **Next** can be triggered by F3.

2.2.5 Parameters

Thjis submenu allows you to choose the appearance of Fitmodel. The button **Font** opens a window where you can select the Font, Size etc and **Background** lets you choose the background color.

3 Creating Models and Syntax

3.1 Constants and function definition

Define all constants and functions which will be used in the model

- **Syntax:**

  ```
  CONSTANT1 = VALUE
  FUNCTION2(x) = (CONDITION1 && or || CONDITION2 ...) ? VALUE1 : VALUE2
  ```

  FUNCTION2 will be VALUE1 if the conditions are true and VALUE2 if they are false. VALUEs can be other functions as well, for example 3*x+1 or x**2.

- **Example:**

  ```
  Pulse(t) = (t >= -0.75 && t < 0.0) ? 1.0 : 0.0
  ```

  This is a rectangular function with the value 1 for $-0.75 < t < 0$ and 0 for the other cases.

3.2 Parameters to fit

List of all parameters which need to be fitted. PARAM can be any combination of letters, numbers and "_".

- **Syntax:**

  ```
  fit PARAM1 = STARTVALUE1 : [MINVALUE1, MAXVALUE1], SCALE1
  fit PARAM2 = STARTVALUE2 : [MINVALUE2, MAXVALUE2], SCALE2
  ...
  ```

- **Example:**

  ```
  fit k_pm_ccv = 1.0 : [1.0e-4, 10.0], log
  ```

3.3 Kinetical model (system of ODE)

List of all ordinary differential equations. NAME can be any combination of letters, numbers and "_". The system must be ordinary, there is only one independent variable (usuall time, t, which is used as example here).

- **Syntax:**

```
dNAME1/dt = FUNCTION1(NAME1,NAME2,...,PARAM1,PARAM2,...)
dNAME2/dt = FUNCTION2(NAME1,NAME2,...,PARAM1,PARAM2,...)
...
```

- **Example:**

```
dEEA1/dt = Pulse(t) * K0 - k_pm_ccv * PM
```

3.4 Simulation interval

Specifies the interval of the independant variable in which the fit of the model should be done.

- **Syntax:**

```
x = [MIN, MAX] : NUMBER_OF_STEPS
this will result in equidistant steps
x = [MIN, MAX] : NUMBER_OF_STEPS, log: SIZE_OF_FIRST_STEP
this will result in logarithmic steps
```

- **Example:**

```
t = [-0.75, 30.0] : 150, log : 0.01
```

3.5 Start condition of ODE

List of start conditions of differential equation system.

- **Syntax:**

```
NAME1 = VALUE1\\
NAME2 = VALUE1\\
...
```

- **Example:**

```
EEA1 = 0.0
```

3.6 Model curves

List of all curves that will be fitted to the data sets later.

- **Syntax:**

```
model MODEL1 = FUNCTION1(NAME1,NAME2,...,MODEL2,MODEL3,...)
```

- **Example:**

```
model mdl_EGF_APPL = (APPL + EEA1_APPL) / mdl_EGF_TOTAL_VES
```

3.7 Fit curves

Lists which data set should be fitted to which curve.

- **Syntax:**

```
compare DATASET1 : MODEL1
```

- **Example:**

```
compare EGF_EEA1 : mdl_EGF_EEA1
```

3.8 Draw result

List of all plots that should be done.

- **Syntax:**

```
draw[PlotID] MODEL1, MODEL2, ... NAME1, NAME2, ... : COLOR
```

- **Examples:**

```
draw[1] PM, CCV, APPL, EEA1_APPL, EEA1, LE, RE, REp, TotalOut : LIGHTGRAY
draw[2] mdl_EGF_TOTAL : LIGHTBLUE
```

4 Known Issues

4.1 Broken Buttons

Tools → Open Graph Model
Tools → Show Covariation Matrix
Tools → Show Correlation Matrix