The art of model fitting to experimental results (Supplementary Information)

Pedro J. Sebastião

Physics Department, Instituto Superior Técnico, University of Lisbon, Av. Rovisco Pais, 1049-001 Lisboa, Portugal; Condensed Matter Physics Center - (IST site), Complexo Interdisciplinar, Av. Rovisco Pais, 1049-001 Lisboa, Portugal

E-mail: `pedro.jose.sebastiao@ist.utl.pt`

Contents

1. Notes on the use of *fitteia* interfaces

fitteia (Fitting Environment Interfaces for All) provides a set of graphic user interfaces accessible through the internet using any web browser (e.g. IE, Opera, Firefox, Mozilla, Safari, Chrome). fitteia is presently available on the site http://fitteia.org.
fitteia platform allows for:

1 The graphical representation of experimental results and/or mathematical functions;

2 The numerical fit of arbitrary user defined linear and/or non-linear functions dependent on M fitting parameters and n-experimental variables to experimental data

3 The data treatment using an integrated programmable calculator

4 The use of a integrated Matlab-like calculator

5 The writing of professional-quality reports using LaTeX

6 The production of documents in PDF, PNG, JPG, EPS formats using the fit results

7 The user registration through a email/password security process to assure the user's privacy

8 The sharing of working areas with other select users

9 The sharing of user defined fitting functions/fitting models, documents/projects with other select users

10 The possibility to use three different expertise levels (e.g. Basic, Advanced, Expert)

11 The saving/recovery of important results

12 The download/upload of the configuration/setup files

13 The assignment of two levels of user privileges

1.1. Registration

The *fitteia* login webpage (http://fitteia.org) presents some basic information about the service and presents a "login" link that users must follow to login the system. New users must first register accessing the "*sign up*" page. A valid email address is required to "*sign up*" *fitteia*. The user receives an email with a new password after answering a security question. This password is required to proceed to the second step of the registration process. The user is invited to change the password and to introduce his name in the system.

Registered users may reset their passwords following the same procedure.

1.2. Logging in

Logging in the system requires typing the email used for the registration and the password. When logging in for the first time the user is directed to a webpage as presented in the print screen of Fig. 1. Some links will conduct to actions that are more

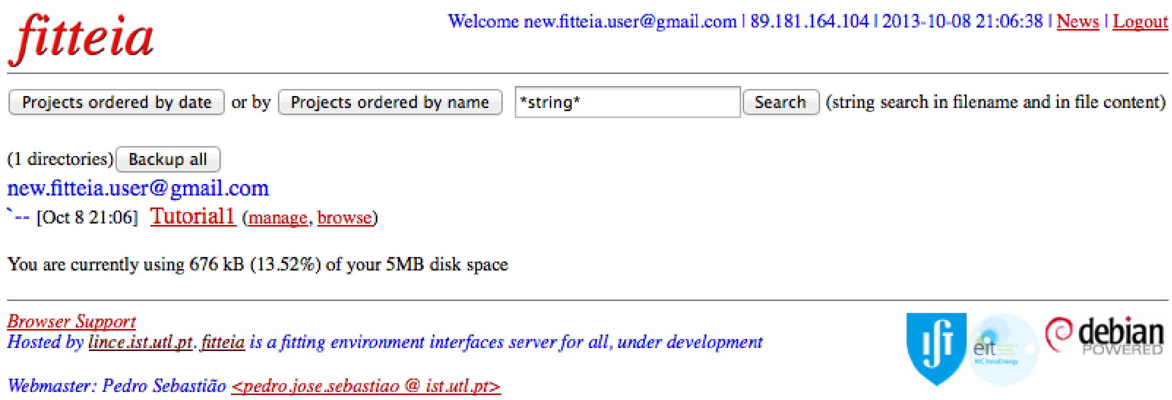


Figure 1. *fitteia* webpage after logging in.

or less self-explaining. The user should check the information concerning the "Browser Support".

"Tutorial1" is the name of a default folder that contains some self-explaining fit and plot examples.

"browser" allows for a vista to the users home folder tree using the web browser.

"manage" links to a webpage where the user can delete files or include emails of other users with whom the folder will be shared provided that the folder's name is not a duplicate of a pre-existing folder of the guest user.

"Tutorial1" links to *fitteia*'s *fitter* module that opens with some pre-recorded data loaded, as partially presented in Fig. 2.

1.3. The fitter module

1.3.1. Header

Figure 2. *fitteia*'s *fitter* webpage header.

Besides the self-evident links the following links deserve some additional comments.

"News" links to a page where new *fitteia* features are presented after being included in the system.

"Basic", "Advance", and "Expert" link to *fitter* module interface with different layouts according to the level of expertise required by the system to tackle the user's data analysis problem.

"Calculator" links to a programable calculator that receives the fitting parameters of the last fit performed and helps the user to perform additional calculus.

"Plotter" links to *fitteia*'s *plotter* module.

Report Editor" links to a web form filled with the last fit results and can be used to prepare LaTeX reports

"Upload" links to a upload web form that allows the user to upload a file to the system. usually a previous downloaded *fitter* or *plotter* environment file.

"MyFits" links to the list of user folders (included folder shared with him).

In Fig. 2 a pair of text boxes can be found between buttons "Overwrite" and "Archive". The text box filled with "Tutorial1" shows the folder's name where the user is working. Additional folders can be created by changing this name. Next to it a text box filled with "file_name" shows the environment name given to file where all the web page information will be saved.

"Overwrite" saves the web form in a file named "file_name.sav" in folder "Tutorial1" This file can be overwritten repeatedly.

"Archive" saves the web form in a file named "file_name-YYYY-MM-DD_hh:mm:ss.sav" in folder "Tutorial1". This file can not be overwritten and can only be deleted in "MyFits->Folder_name->manage".

On the bottom right side of Fig. 2 it is possible to see a button to select a previous saved environment fit data file to "Recover" or "Download". The list of files can be filtered using the "list" button.

1.3.2. Data, Plot Settings and Fitting Function

The core of the *fitter* module webpage is composed by the sections with following characteristics:

"1°-Data" The main text box allows the user to type (or copy and paste) values of x, y, and ϵ_y, where x is the independent variable, y is the dependent variable, and ϵ_y is expected to be the uncertainty of y. The numbers must be written in three columns and separated by white spaces. The values x, y, and ϵ_y form column vectors, named "c1", "c2",...,"cn", where 1,...,n is the order of the column

Below the main "Data" text box a second box allows users to perform arithmetic operations with the column vectors of the "Data" text box. The syntax for the arithmetic calculations is that of PERL but it is quite simple to write expressions to perform basic calculations. Both "ˆ"and "**" can be used in power expressions (e.g. c1ˆp or c1**p).

abs(c1) – absolute values of column vector c1;

sqrt(c1) – square root function of column vector c1;

exp(c1) – exponential function, e^{c1};

sin(c1), cos(c1), tan(c1), asin(c1), acos(c1), atan(c1) – trigonometric functions of column vector c1;

sinh(c1), cosh(c1), tanh(c1), asinh(c1), acosh(c1), atanh(c1) – hyperbolic functions of column vector c1;

Button "Calculate" when be used evaluates these expressions in sequence and generates a new Data table.

Button "Data Plot" generates a new layout of the *fitter* webpage where a graph with a data plot is presented according to the "Plot Parameters" introduced.

"2°-Plot Parameters" Axis names, limits, tick names, etc.. can be defined in this section. Both x and y autoscaling is possible.

"3°-Function and Parameters" The function to be fitted to the experimental results must be written in language C. The names of the independent and dependent variables must be introduced and they will have the corresponding values of the first and second columns of the "Data" table when performing the least-squares' minimization. For the large majority of cases the only language C specific mathematical functions that users must be aware of are:

$fabs(x)$ – absolute value of a floating point number x;

$pow(x, y)$ – x^y;

$sqrt(x)$ – square root function, $\sqrt{x}$;

$exp(x)$ – exponential function, e^x;

$sin(x)$, $cos(x)$, $tan(x)$, $asin(x)$, $acos(x)$, $atan(x)$ – trigonometric functions;

$sinh(x)$, $cosh(x)$, $tanh(x)$, $asinh(x)$, $acosh(x)$, $atanh(x)$ – hyperbolic functions;

For conditional calculations the syntax "(condition) ? if_true_this : if_false_this" can be used. When calculating ratios between integers, like in the case of "1/2" it is required to write "1.0/2.0" to obtain the expected value 0.5 as the result, otherwise an integer devision is performed and the result is 0.

If the fitting function is composed of different terms, *ftteia*'s *fitter* specific syntax character "\+" can be used to allow for the independent plotting of each one of the terms of the fitting function. The "Plot Parameters" section layout is changed according to the number of terms of the fitting function to allow for the setting of lines types, colors and labels of each curve.

Button "Compile" will compile the function and link it with the *fitteia*'s fitter kernel. A link to a log file might help to debug the function in case compilation errors occur. The *fitter* webpage layout changes in the case of a successful compilation and below the page header a graph and the function parameters' table are shown next to each other.

In the Data tex box and in the text boxes containing the plot settings addition arithmetic expressions can be included and will evaluated when the web form is submitted to the *fitteia* server.

1.3.3. Recovering a previous fit

After selecting the name and recovering the fit the *fitter* webpage layout shows the graph with the data and the function parameters' table as illustrated in Fig. 3 for the fit file "Basic-EJP-Resonance-const-err-2013-08-09_19-13-54".

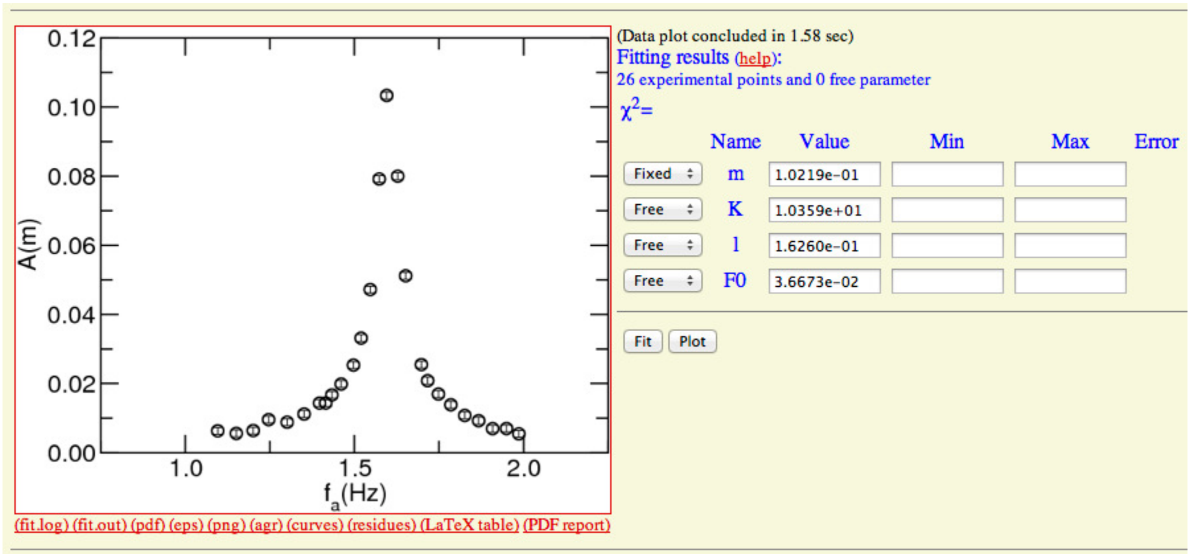


Figure 3. *fitteia*'s *fitter* graph and function parameters' table after "Recover" a previously saved fit.

If the the user has still storage space in his account buttons "Fit" and "Plot" are seen below the function parameters' table. They allow for two different actions:

"Fit" starts the fitting procedure using the values of the fitting parameters in the table as initial values. If a fitting parameter status is "Free" the minimization procedure tries to find the minimum of χ^2 in the parameter's space defined by the user. If "Min" and "Max" have finite values for a given parameter the minimization is restricted to that particular region of the parameter's space, otherwise the region extends $[-\infty; \infty]$. The graph is updated with both data points and the fitting curves calculated using the fitting output values.

"Plot" generates a graph with the data points and the curves calculated using the values of the parameters without performing a fitting procedure. It can be used to generate curves calculated with different parameters.

When the fitting ends an additional column is added to the function parameters' table with either the error estimates of the "Free" parameters, or the word "Fixed" stating the status of that parameter during the minimization procedure, as illustrated in Fig. 4

Below the graph several links become handy when exporting the graph in different file file formats, getting an automatic PDF report of the fit, obtaining a table with the values of the theoretical curves, or a LaTeX table with the fitting results.

"fit.log" deserves a comment as it contains a log report of the MINUIT actions during the fitting procedure plus some information from the *fitteia*'s *fitter* module.

"PDF Report" links to the "Report Editor" previously referred and returns a PDF report that contains the graph and fitting results obtained (check below some examples).

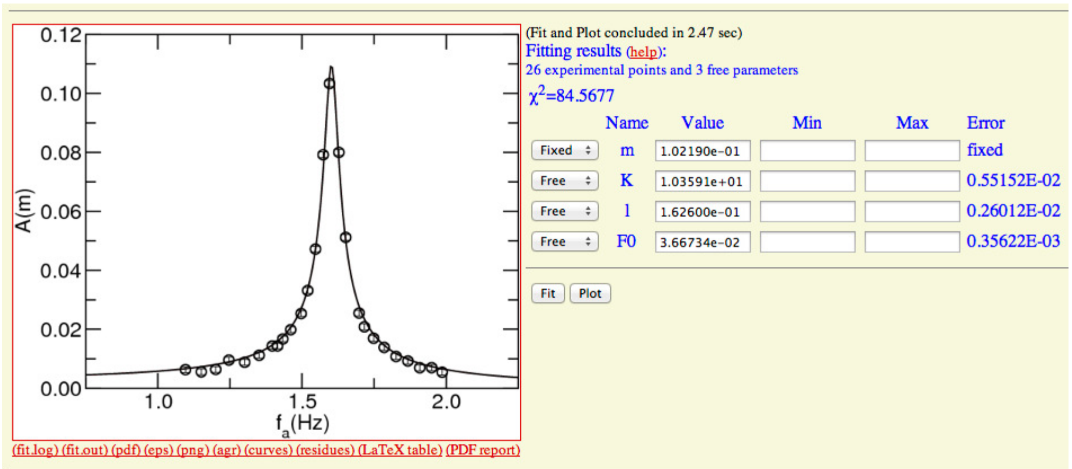


Figure 4. *fitteia*'s *fitter* graph and function parameters' table after a fitting procedure.

1.4. Using the Calculator module

After a fit the function parameters obtained are exported to the *Calculator* module. The *Calculator* web form layout shows a text box area where the user can write equations and expressions and perform different types of calculations including tables and/or vector operations. By default *Calculator* starts with a list of examples that can be studied as a tutorial text. In Fig. 5 part of *Calculator* webpage is illustrated after opening the "example" operations' set previously saved. The fitting parameters obtained in the fit of Fig. 4 were loaded and simple calculations using those parameters are presented.

1.5. Plotter module

Producing more elaborated graphs can be achieved using *fitteia*'s *Plotter* module. When opened it presents a basic layout. After "recovering" a previous save example its layout is modified as illustrated in Fig. 6.

The particular example illustrated in Fig. 6 presents 7 data sets where one data set is shown with black circles and the reaming data sets are represented by lines. The settings of each data set are of the type shown in Fig. 7 for Data set 1.

1.6. Advanced and Expert modes

Using the notes presented above for the *fitteia*'s Basic user mode the behavior of the *fitter* module for the remaining user modes can be followed using the examples illustrated in the "Tutorial1" folder.

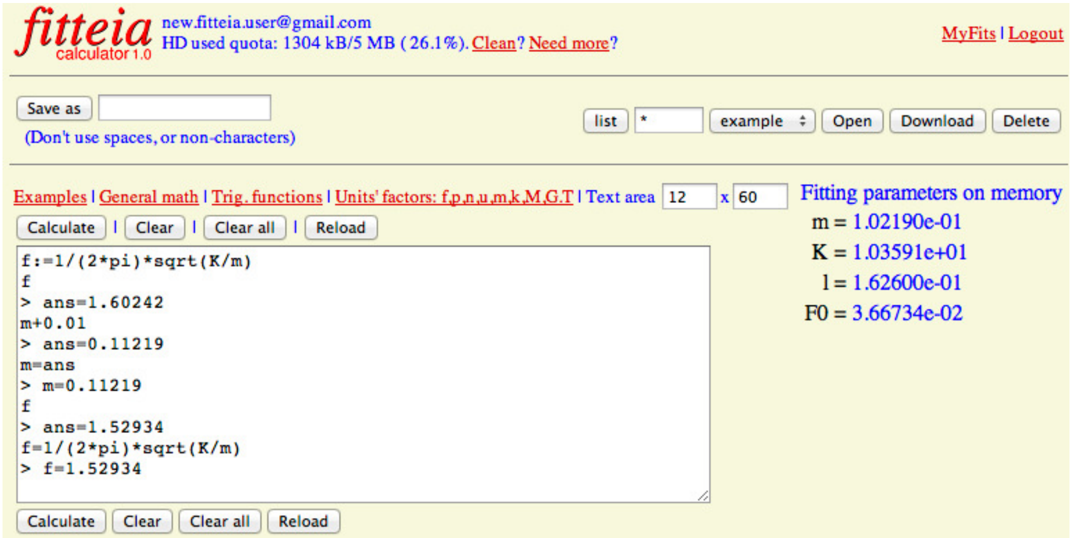


Figure 5. *fitteia*'s *Calculator* layout example.

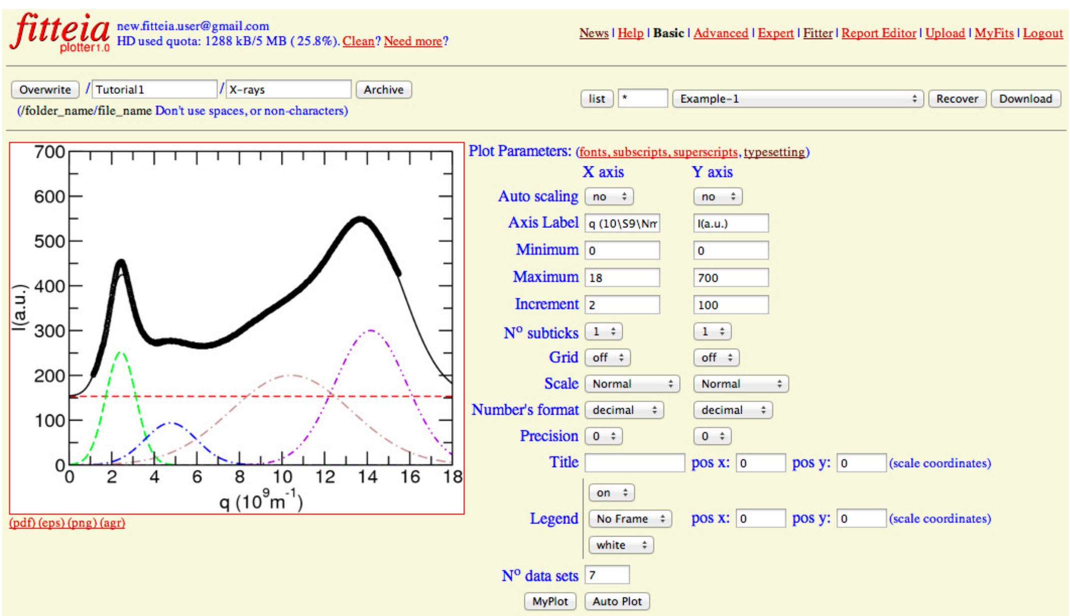


Figure 6. *fitteia*'s *Plotter* layout example.

1.7. Use of fitteia with increased privileges

For advanced model fitting the basic functions writing is, some times, not enough as new complex functions might be required. If they are not included in the system a user can ask for additional privileges in order to be able to develop more elaborated functions and libraries. The only difference noticed by the users is the layout of the *fitter* module as it includes additional text boxes for the additional user's C code. In Fig. 7 is presented one example of the *fitter* web age layout of section "Function and

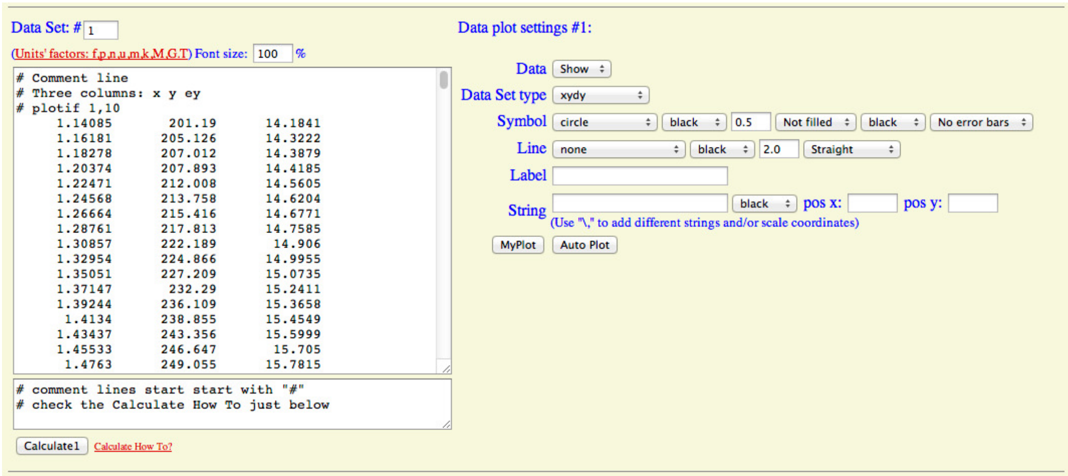


Figure 7. *fitteia*'s *Plotter* layout example of plot setting for one data set.

Parameters".

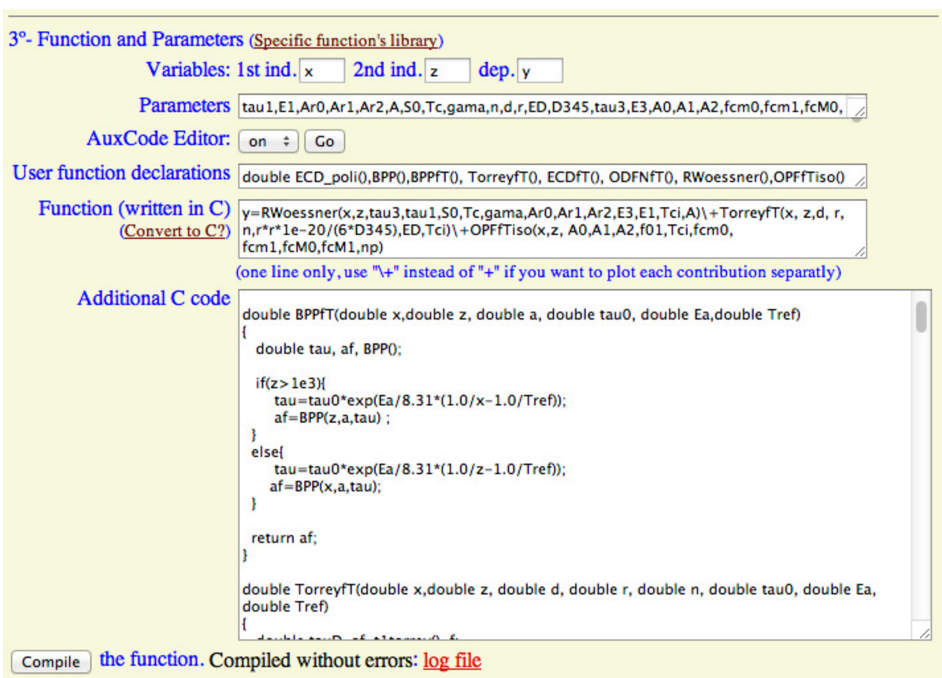


Figure 8. *fitteia*'s *fitter* layout example of section "Function and Parameters" for a privileged user.

2. Notes on the NMR relaxation models

2.1. Translational self-diffusion

The relaxation model used to fit spin-lattice relaxation results of a liquid crystal compound is in many cases that developed by H. C. Torrey [1] and can be expressed as

$$\left(T_1^{-1}\right)_{\mathrm{SD}} = \frac{9}{8}\gamma^4\hbar^2\left(\frac{\mu_0}{4\pi}\right)^2\frac{n\tau_D}{d^3}\left[\mathcal{T}(\alpha,\omega\tau_D) \;+\; 4\mathcal{T}(\alpha,2\omega\tau_D)\right], \tag{1}$$

with $\mathcal{T}(\alpha,x)$ given by

$$\mathcal{T}(\alpha,\,x) = \frac{32\pi\alpha}{15x^2}\left[\left(u_+\left(1+\frac{1}{u_-^2+u_+^2}\right)+2\right)e^{-2u_+}\cos u_- \right.$$
$$\left. +\; u_+\left(1-\frac{1}{u_-^2+u_+^2}\right) \;+\; u_-\left(1-\frac{1}{u_-^2+u_+^2}\right)e^{-2u_+}\sin 2u_-\right] \tag{2}$$

where

$$u_\pm = \frac{1}{2}\left(\frac{q(1\pm q)}{\alpha}\right)^{1/2} \tag{3}$$

$$q \;=\; \frac{x}{(4+x^2)^{1/2}} \tag{4}$$

and $\alpha = \langle r^2\rangle/(12d^2)$. In this model n is the density of spins, d is distance between neighborhood molecules, $\langle r^2\rangle$ is the mean square jump distance, and τ_D is the average time between molecular translational jumps.

2.2. Order parameter fluctuations

In the isotropic phase of a liquid crystal compound, close to the phase transition into the nematic phase, fluctuations of the order parameters are some times observed [2] when analyzing spin-lattice relaxation results. The relaxation mechanism can be expressed bt

$$\left(T_1^{-1}\right)_{OPF} = \frac{A_{OPF}}{\omega^{1/2}}\int_{\omega_{Cl}/\omega}^{\omega_{Ch}/\omega}\frac{\sqrt{x}}{1+(x+\omega_0/\omega)^2}dx \tag{5}$$

with $A_{OF} = (9/8)(\mu_0/4\pi)^2\gamma^4\hbar^2 k_B T\eta_{iso}^{1/2}/(4\pi^2K_{iso}^{3/2}a_{eff}^6)$. ω_{Cl} and ω_{Ch} are the low and high cut-off frequencies that correspond to the largest and smallest wave length modes, respectively. $\omega_0 \sim K_{iso}/\eta\xi^2$ depends on the coherence length associated with the size of nematic cybothatic domains in the isotropic phase and acts also as a cut-off frequency [2].

References

[1] H. C. Torrey. Nuclear spin relaxation by translational diffusion. *Physical Review*, 92(4):962–969, 1953.

[2] R.Y. Dong. *Nuclear magnetic resonance of liquid crystals.* Springer New York, 1997.

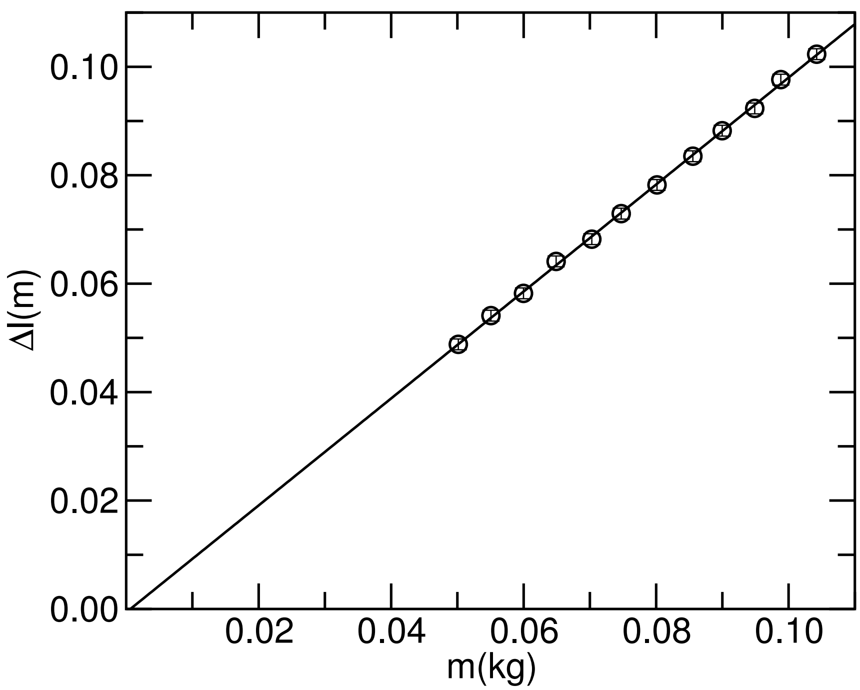
3. ***fitteia***'s ***fitter*** reports corresponding to the examples presented in the paper

fitteia Report
(internet based fitter service)

Subject	Tutorial1, Basic-EJP-Elastic-Constant
Date	Friday 9$^{\text{th}}$ August, 2013, 20:30
Affiliation	pj.sebastiao@gmail.com 77.54.10.14
Abstract	Fit report produced with the fit results of function: y= g/K*x + d0 to the 12 experimental points, considering 2 free parameters.

$g = 9.8$ (fixed) $K = 9.9356 \pm 0.17085$	$d0 = -0.00062389 \pm 0.001344$

$\chi^2[1] = 2.22217 \qquad \chi^2_t = 2.22217$



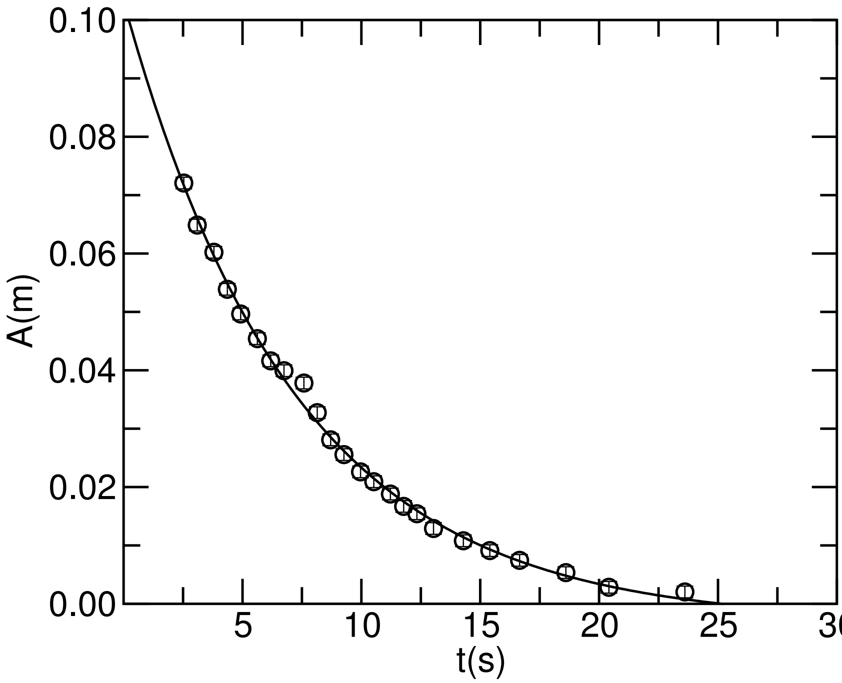
recta.pdf

fitteia Report
(internet based fitter service)

Subject	Tutorial1, Basic-EJP-Damped-oscilator
Date	Friday 9th August, 2013, 20:30
Affiliation	pj.sebastiao@gmail.com 77.54.10.14
Abstract	Fit report produced with the fit results of function: Z=(Zt0-Zinf)*exp(-l*t) + Zinf to the 24 experimental points, considering 3 free parameters.

$l = 0.13878 \pm 0.0044065$	$Zinf = -0.0032395 \pm 0.00093744$
$Zt0 = 0.10319 \pm 0.0015956$	

$\chi^2[1] = 30.1509 \qquad \chi_t^2 = 30.1509$



lambda.pdf

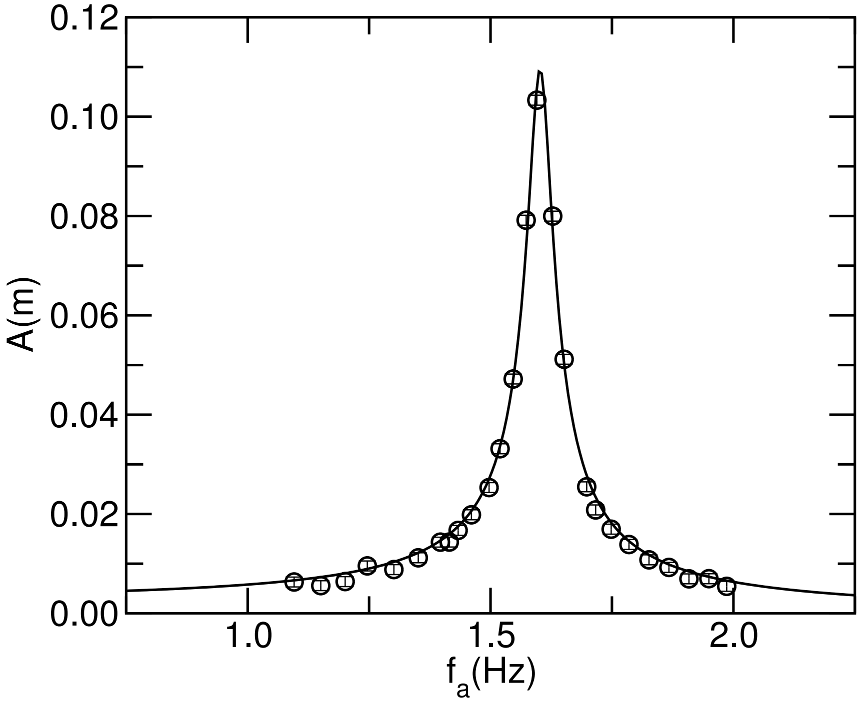
fitteia Report

(internet based fitter service)

Subject	Tutorial1, Basic-EJP-Resonance-const-err
Date	Friday 9$^{\text{th}}$ August, 2013, 20:31
Affiliation	pj.sebastiao@gmail.com 77.54.10.14
Abstract	Fit report produced with the fit results of function: A=F0/(m*sqrt(pow(K/m-4*pi*pi*fa*fa, 2.0) + 16*pi*pi*l*l*fa*fa)) to the 26 experimental points, considering 3 free parameters.

$m = 0.10219$ (fixed)	$l = 0.1626 \pm 0.0026012$
$K = 10.359 \pm 0.0055152$	$F0 = 0.036673 \pm 0.00035622$

$\chi^2[1] = 84.5677 \qquad \chi^2_t = 84.5677$



A.pdf

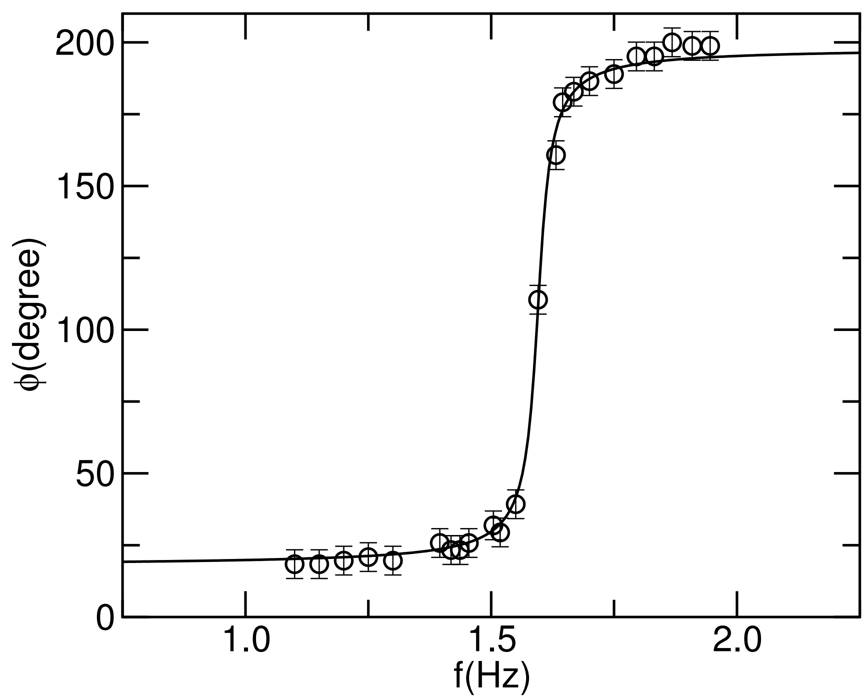
fitteia Report

(internet based fitter service)

Subject	Tutorial1, Basic-EJP-Phase-const-err
Date	Friday 9th August, 2013, 20:40
Affiliation	pj.sebastiao@gmail.com 93.108.109.5
Abstract	Fit report produced with the fit results of function: y=(x>sqrt(K/m)/(2*pi)) ? (atan(2*l*2*pi*x/(K/m-4*pi*pi*x*x)) + pi + b)*180/pi : (atan(2*l*2*pi*x/(K/m-4*pi*pi*x*x)) + b)*180/pi to the 23 experimental points, considering 3 free parameters.

$m = 0.10219$ (fixed)	$l = 0.12442 \pm 0.012487$
$K = 10.269 \pm 0.022925$	$b = 0.32068 \pm 0.019616$

$\chi^2[1] = 8.5947 \qquad \chi^2_t = 8.5947$



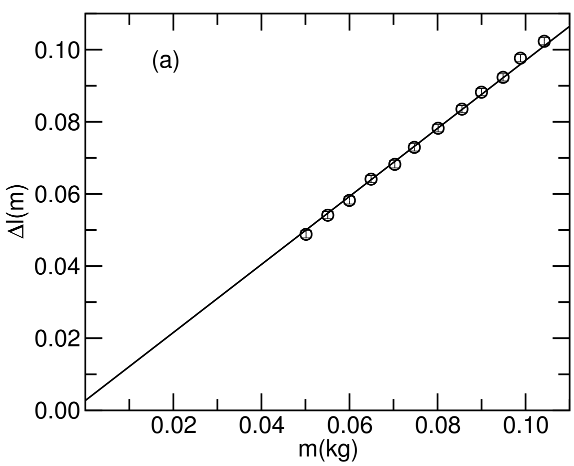
recta.pdf

fitteia Report
(internet based fitter service)

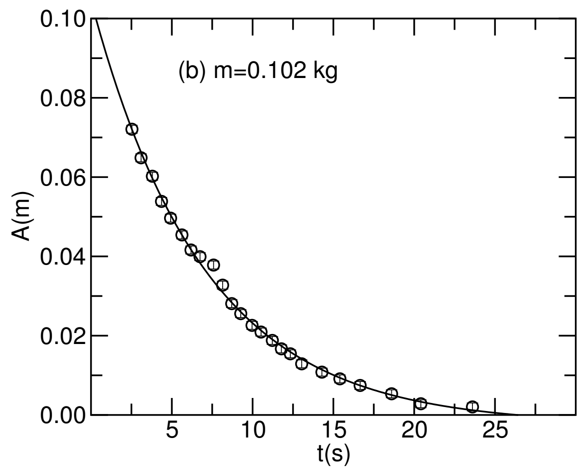
Subject	Tutorial1, Expert-EJP-spring-mass
Date	Friday 9th August, 2013, 20:24
Affiliation	pj.sebastiao@gmail.com 77.54.10.14
Abstract	Fit report produced with the fit results of function: y=(flag==1) ? g/K*x + d0 : (flag==2) ? (Zt0-Zinf)*exp(-l*x) + Zinf : (flag==3) ? F0/m/sqrt(pow(K/m-4*pi*pi*x*x, 2.0) + 16*pi*pi*l*l*x*x) : (x>sqrt(K/m)/(2*pi)) ? (atan(2*l*2*pi*x/(K/m-4*pi*pi*x*x)) + pi + a0)*180/pi : (atan(2*l*2*pi*x/(K/m-4*pi*pi*x*x)) + a0)*180/pi to the 85 experimental points, considering 8 free parameters.

$g = 9.8$ (fixed)	$A = 0.64014 \pm 0.09051$
$m = 0.10219$ (fixed)	$Zt0 = 0.1045 \pm 0.00058394$
$K = 10.392 \pm 0.0076501$	$Zinf = -0.0024616 \pm 0.0002795$
$d0 = 0.0027261 \pm 0.00028868$	$a0 = 0.33976 \pm 0.019833$
$l = 0.14291 \pm 0.00077868$	$F0 = 0.032986 \pm 0.00029406$

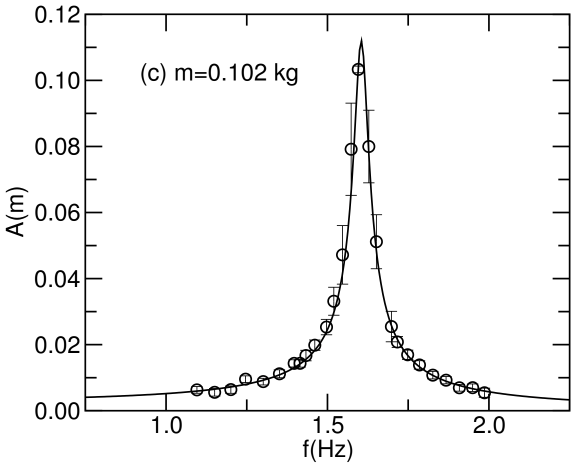
$\chi^2[2] = 31.0108$ $\chi^2[1] = 8.73329$
$\chi^2[4] = 10.327$ $\chi^2[3] = 10.8119$
$\chi^2_t = 60.8831$



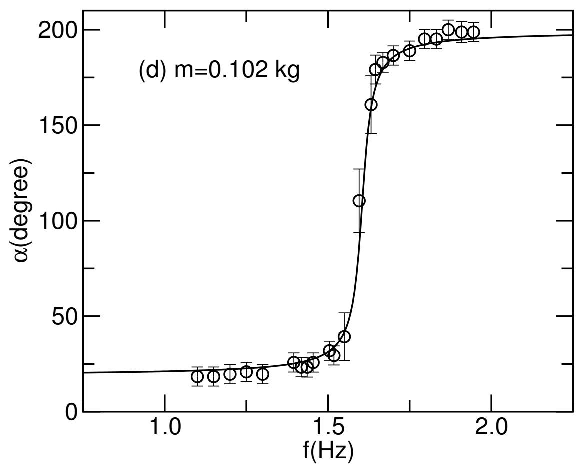
Spring-Mass-1.pdf



Spring-Mass-2.pdf



Spring-Mass-3.pdf



Spring-Mass-4.pdf

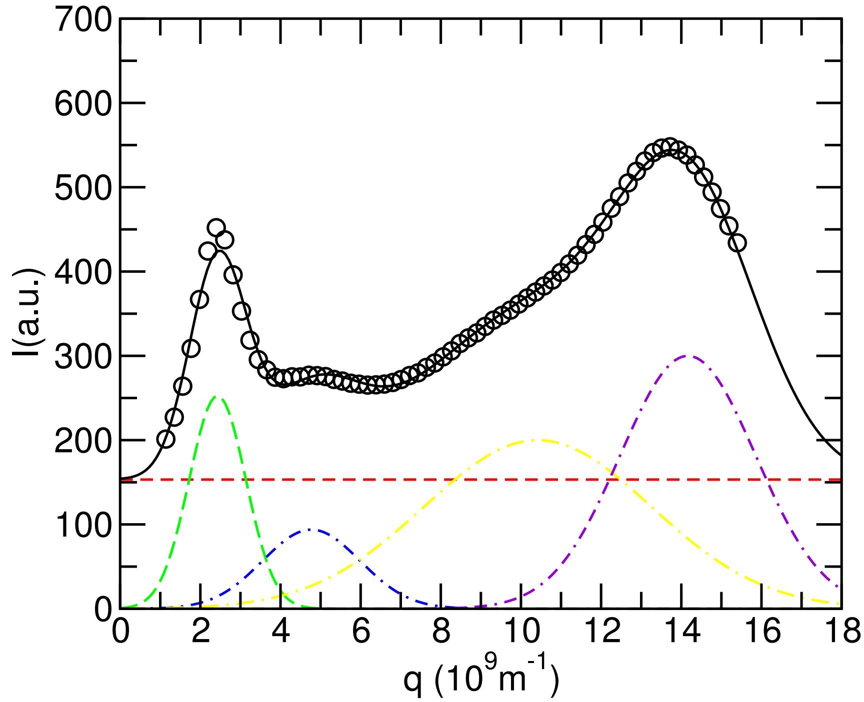
fitteia Report

(internet based fitter service)

Subject	Tutorial1, Basic-EJP-x-rays
Date	Friday 9$^{\text{th}}$ August, 2013, 20:21
Affiliation	pj.sebastiao@gmail.com 77.54.10.14
Abstract	Fit report produced with the fit results of function: y=a + a0*exp(-pow((q-q0)/b0, 2.0)) + a1*exp(-pow((q-q1)/b1, 2.0)) + a2*exp(-pow((q-q2)/b2, 2.0)) + a3*exp(-pow((q-q3)/b3, 2.0)) to the 69 experimental points, considering 13 free parameters.

$a = 153.21 \pm 5.4086$	$a2 = 200 \pm 29.723$
$a0 = 252.52 \pm 15.052$	$b2 = 4 \pm 0.12931$
$b0 = 1 \pm 0.014206$	$q2 = 10.419 \pm 0.12791$
$q0 = 2.4294 \pm 0.040112$	$a3 = 300 \pm 9.6683$
$a1 = 93.836 \pm 7.8405$	$b3 = 2.3904 \pm 0.14258$
$b1 = 1.7183 \pm 0.29467$	$q3 = 14.164 \pm 0.080498$
$q1 = 4.7695 \pm 0.14836$	

$\chi^2[1] = 8.78204 \qquad \chi_t^2 = 8.78204$



spectrum.pdf

fitteia Report

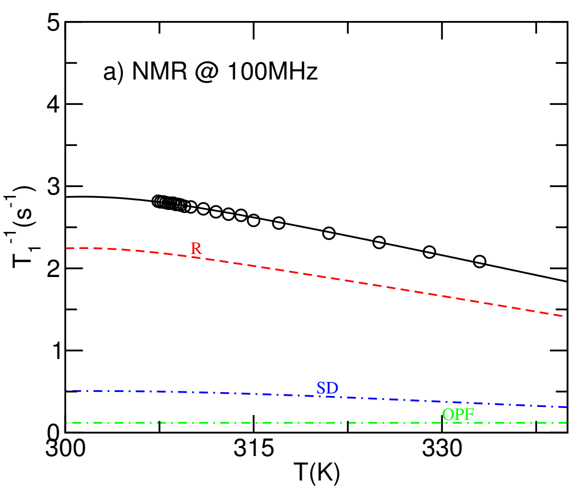
(internet based fitter service)

Subject	Tutorial1, Expert-EJP-5CB_Iso
Date	Friday 9th August, 2013, 20:26
Affiliation	pj.sebastiao@gmail.com 77.54.10.14
Abstract	Fit report produced with the fit results of function: z=((y>500.0) ? BPP(y, Arot, tauL*exp(EL/8.31*(1.0/x-1.0/Tref))) : BPP(x, Arot, tauL*exp(EL/8.31*(1.0/y-1.0/Tref)))) + ((y>500.0) ? BPP(y, Arot, tauS*exp(ES/8.31*(1.0/x-1.0/Tref))) : BPP(x, Arot, tauS*exp(ES/8.31*(1.0/y-1.0/Tref)))) + ((y>500.0) ? Torrey1(y, d, r, n, r*r*1e-20/(6*Dref*exp(-ED/8.31*(1.0/x-1.0/Tref)))) : Torrey1(x, d, r, n, r*r*1e-20/(6*Dref*exp(-ED/8.31*(1.0/y-1.0/Tref))))) + ((y>500.0) ? OPF(y, Aopf, f0, fcm, fcM, np) : OPF(x, Aopf, f0, fcm, fcM, np)) to the 44 experimental points, considering 4 free parameters.

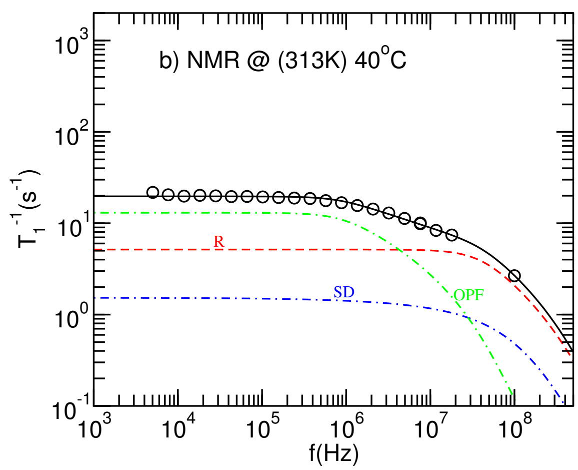
$Arot = 5.7688 \times 10^{+08} \pm 8.45 \times 10^{+06}$	$Dref = 5.44 \times 10^{-11}$ (fixed)
$tauS = 1.45 \times 10^{-09}$ (fixed)	$ED = 32800$ (fixed)
$ES = 29412$ (fixed)	$Tref = 313$ (fixed)
$tauL = 3.36 \times 10^{-10}$ (fixed)	$Aopf = 17438 \pm 1017.9$
$EL = 46828 \pm 6860.6$	$f0 = 1 \times 10^{+05}$ (fixed)
$d = 5$ (fixed)	$fcM = 4.25 \times 10^{+07}$ (fixed)
$n = 4.59 \times 10^{+22}$ (fixed)	$fcm = 6.838 \times 10^{+05} \pm 99853$
$r = 4$ (fixed)	$np = 10$ (fixed)

$\chi^2[313] = 8.81902 \qquad \chi^2[1 \times 10^{+08}] = 0.184233$

$\chi^2_t = 9.00325$



recta-1.pdf



recta-2.pdf