FITTER. The package for fitting a chosen theoretical multi-parameter function through a set of data points. Application to experimental data of the YuMO spectrometer

A G Soloviev¹, T N Murugova²³, A H Islamov³ and A I Kuklin²³

¹ Laboratory of Information Technologies, JINR, Joliot-Curie-6, Dubna 141980 Russia
² Frank Laboratory of Neutron Physics, JINR, Joliot-Curie-6, Dubna 141980 Russia
³ Bionanocenter of Moscow Institute for Physics and Technology, Dolgoprudny, Russia

E-mail: solovjev@jinr.ru

Abstract. Fitter is a C++ program aimed to fit a chosen theoretical multi-parameter function through a set of data points. The method of fitting is chi-square minimization. Moreover, the robust fitting method can be applied in Fitter. Fitter was designed to be used for a small-angle neutron scattering data analysis. Respective theoretical models are implemented in it. Some common used models (Gaussian and polynomials) are also implemented for wider applicability. The program includes the option to resolve the spectrometer. The investigation has been performed at the Laboratory of Information Technologies and at the Frank Laboratory of Neutron Physics, JINR. Fitter can be downloaded from [1].

1. Introduction

Fitter, the package for fitting a chosen theoretical multi-parameter function through a set of data points, has been successfully used in small-angle neutron scattering data analysis since 2003 [2; 3]. The several examples of researches using Fitter are given below. This handbook gives a long write-up description of the new version of Fitter. The method of small-angle neutron scattering (SANS) is frequently used for research of fundamental and applied problems in condensed matter physics, biology and chemistry. This method is often an only way to obtain direct structural information on systems with disordered and partly ordered arrangement of the nuclear density with characteristic sizes of about 20-1000 angstroms.

The small-angle neutron scattering (SANS) method has long history. So, the earliest references to SANS can be traced back to 1948 by D.J.Huges et al. [4]. Starting with classical work of A.Guinier, who suggested the so called "Guinier approximation" to powerful package of D.Svergun team [5-11], who developed and brought SANS method up to present moment. At the same time, in spite of the wide SANS...
method application and already more than 50-years history of its development, the problem of interpreting
the data obtained on spectrometers, still remain open. It is related, first of all, to the ill-posedness of the
solved problem and, on the other hand, to specifics of existing SANS spectrometers. Practically each
spectrometer is provided by not only software on initial processing experimental data, but also by a
complex of programs for fitting the curves measured in coordinates scattering intensity/q, where
\[ q = \frac{4\pi}{\lambda}\sin\left(\frac{\theta}{2}\right). \]

A modernization of the time-of-flight SANS YuMO spectrometer at the high flux pulse IBR-2 reactor
(JINR, Dubna) [12] was launched in 2000. One of the goals of the improvement was to increase the
dynamic Q-range of the instrument. It was realized by using a two-detector system with central holes at the
YuMO instrument. For this purpose "old" and "new" detectors were used. The main view of the setup
of the modernized YuMO instrument as well as an examples of a SANS experiments are presented in [13].
The use of the two detectors YuMO set up for the users have clearly demonstrated its high efficiency [14-30].
That are only several examples. In 2006 the position sensitive detector started at the YuMO
spectrometer [31]. For mono-disperse, homogeneous, globular particles, general intensity of scattering can
be expressed by a number of particles N, structural factor S(q) and form-factor F(q). However, for many
practically important applications, influence of the structural factor on experimental results is
insignificant. It is, for example, for strongly diluted not charged samples or even for the charged samples
but with good charge shield [32]. For more complicated model it is necessary the new programm
for segment [33] or a long helical model was created [34]. Neverethelss, easy development and quickly
programm necessary to have during modernization of spectrometer.

The new Fitter version is better suit requirements of new detector setup. The main changes are:

- fitting in a given range;
- improving the robust approach;
- choosing the MINUIT strategy;
- function in STD models;
- new visualization and GUI;

This paper is a long write-up description of the current version of program for fitting a chosen
theoretical multi-parameter function through a set of data points. Examples of application to experiments
on the small-angle spectrometer YuMO are considered.

2. Long Write-Up
The Fitter is designed to be used for SANS data processing first of all. Thus, SANS theoretical models are
implemented in it. Moreover, some standard mathematical models are added for wider applicability.
Besides the implemented theoretical models, Fitter has a minimization module. It provides a safe call of
MINUIT [35] procedures in the current version. The important feature of Fitter's design is its
expandability: both new models and new minimizing algorithms can easily be added to the existing ones.

2.1. Theoretical Models
Fitter's model module is designed as follows (figure 1). Abstract base class Model provides a common
interface, used by a minimization module. Thus, any theoretical model class inherits from it. Model
classes currently implemented in the Fitter are: STDModel, SANSModel, YuMOModel. Each concrete
class provides several theoretical functions. All of them are described below.
2.1.1. Standard Models. Standard mathematical models implemented in the Fitter are:

- **Gaussian**

  \[ G(x) = A \exp \left( -\frac{1}{2} \left( \frac{x - x_0}{\sigma} \right)^2 \right) + B \]

- **Exponential**

  \[ E(x) = A \exp \{mx + b\} + B \]

- **Polynomials** up to the eighth order

  \[ P_n(x) = \sum_{k=0}^{n} p_k x^k \]

2.1.2. **SANS (Small-Angle Neutron Scattering) Models.** Determination of invariants for small-angle scattering curves allows one to analyze the structure of a particle under study. Upon the first step of this analysis the particle form is approximated by simple geometrical bodies - ellipsoids, cylinders, prisms.

   Thus, SANS models implemented in the Fitter are:

- **Ball** of the radius \( R \)
\[ I(Q) = A\Phi^2(QR) + B \]

\[ \Phi(t) = 3\frac{\sin t - t\cos t}{t^3} \]

- **Ellipsoid of revolution** with half-axes \( a \) and \( av \)

\[ I(Q) = A\int_0^1 \Phi^2 \left[ Qa\sqrt{1 + x^2(v^2 - 1)} \right] dx + B \]

\[ \Phi(t) = 3\frac{\sin t - t\cos t}{t^3} \]

- **Cylinder** of the radius \( R \) and the length \( H \)

\[ I(Q) = A\int_0^1 \Lambda_i^2 \left( QR\sqrt{1 - x^2} \right) S^2(QHx / 2) dx + B \]

\[ \Lambda_i(t) = 2J_i(t) / t \]

\[ S(t) = \sin t / t \]

- **Elliptical cylinder** with half-axes \( a, av \) and the length \( H \)

\[ I(Q) = A\int_0^1 \Psi_{ec}^2(Q,a\sqrt{1-x^2}) S^2(QHx / 2) dx + B \]

\[ \Psi_{ec}^2(Q,a) = \frac{1}{\pi} \int_0^\frac{\pi}{2} \Lambda_i^2 \left( Qa\sqrt{\frac{1 + v^2}{2}} + \frac{1-v^2}{2}\cos y \right) dy \]

\[ \Lambda_i(t) = 2J_i(t) / t \]

\[ S(t) = \sin t / t \]

- **Parallelepiped** with the ribs \( a, b, c \)

\[ I(Q) = A\int_0^1 \Psi_p^2 \left( Q,b\sqrt{1-x^2},a \right) S^2(Qbcx / 2) dx + B \]
\[ \Psi_p(Q,b,a) = \frac{2}{\pi} \int_0^{\pi/2} S^2(Qa \sin y / 2) S^2(Qb \cos y / 2) dy \]

\[ S(t) = \sin t / t \]

- **Spherical shell** of the outer radius \( R_1 \) and the inner radius \( R_2 \)

\[ I(Q) = A \left[ \Phi(QR_1) - \left( \frac{R_2}{R_1} \right)^3 \Phi(QR_2) \right]^2 + B \]

\[ \Phi(t) = 3 \frac{\sin t - t \cos t}{t^3} \]

- **Triaxial ellipsoid** with half-axes \( a, b, c \):

\[ I(Q) = A \int_0^1 \int_0^1 \Phi(Q \sqrt{(a^2 \cos \frac{\pi}{2} x + b^2 \sin \frac{\pi}{2} x)(1 - y^2) + c^2 y^2}) dx dy + B \]

\[ \Phi(t) = 3 \frac{\sin t - t \cos t}{t^3} \]

- **Two axis ellipsiodal shell** with half-axes \( a, a, \nu \) and thickness \( t \):

\[ I(Q) = A \int_0^1 \int_0^1 V_c \Phi(Q \sqrt{(a + t)^2(1 - x^2) + x^2(\nu a + t)^2}) - V_c \Phi(Qa \sqrt{1 + x^2(y^2 - 1)})] dx + B \]

\[ V_c = \frac{4}{3} \pi abc \quad \text{(Volume of core)} \]

\[ V_c = \frac{4}{3} \pi (a+t)^2(\nu a + t) \quad \text{(Volume of core with shell)} \]

- **Triaxial ellipsoidal shell** model

\[ I(Q) = A \int_0^1 \int_0^1 V_c \Phi(Q \sqrt{(a+t)^2 \cos \frac{\pi}{2} x + (b+t)^2 \sin \frac{\pi}{2} x)(1 - y^2) + (c+t)^2 y^2}) \]

\[ -V_c \Phi(Q \sqrt{(a^2 \cos \frac{\pi}{2} x + b^2 \sin \frac{\pi}{2} x)(1 - y^2) + c^2 y^2})] dx dy + B \]

\[ \Phi(t) = 3 \frac{\sin t - t \cos t}{t^3} \]

\[ V_c = \frac{4}{3} \pi abc \quad \text{(Volume of core)} \]
\[ V_i = \frac{4}{3} \pi (a + t)(b + t)(c + t) \] (Volume of core with shell)

More detailed information about SANS models is available in [36].

2.1.3. SANS Models with YuMO spectrometer resolution. These models are implemented to fit data measured on the YuMO spectrometer [37] operated on the 4-th channel of the fast pulsed reactor IBR-2 [12]. YuMO models are the same as SANS, but they take into account the spectrometer resolution.

- **SANS Model with resolution**

\[
I(Q) = \frac{1}{\sqrt{2\pi\sigma^3}} \int_{Q_{3\sigma}}^{Q_{3\sigma}} \left( I_{\text{SANS}}(q) \exp \left\{ -\frac{1}{2} \left( \frac{q - Q}{\sigma} \right)^2 \right\} \right) dq
\]

\[ \sigma = \sqrt{2\Delta Q} \]

2.2. Minimization

To find theoretical model parameters, one should minimize a functional, which is a measure of deviation between a theoretical curve and experimental data. In a common case of a least-squares fit, the functional under minimization is a chi-square:

\[
\chi^2 = \frac{1}{N-N_{\text{params}}} \sum_{i=1}^{N} \left( \frac{f(x_i) - y_i}{\Delta y_i} \right)^2
\]

2.2.1. Minuit. We are using ROOT::TMinuit class to perform a minimization. This package was originally written in Fortran by Fred James and part of PACKLIB (patch D506) and has been converted to a C++ class by R.Brun. The current implementation in C++ is a straightforward conversion of the original Fortran version. The main changes are:

- The variables in the various Minuit labeled common blocks have been changed to the TMinuit class data members.
- The internal arrays with a maximum dimension depending on the maximum number of parameters are now data members arrays with a dynamic dimension such that one can fit very large problems by simply initializing the TMinuit constructor with the maximum number of parameters.
- The include file Minuit.h has been commented as much as possible using existing comments in the code or the printed documentation.
- The original Minuit subroutines are now member functions.
- Constructors and destructor have been added.
- Instead of passing the FCN function in the argument list, the addresses of this function is stored as pointer in the data members of the class. This is by far more elegant and flexible in an interactive environment. The member function SetFCN can be used to define this pointer.
- The derived class TMinuitOld contains obsolete routines from the Fortran based version.
Additional modifications were made to separate TMinuit class from the ROOT package (Fitter is ROOT independent indeed).

MINUIT offers the user a choice of several minimization algorithms. The MIGRAD algorithm is, in general, the best minimizer for nearly all functions. It is a variable-metric method with inexact line search, a stable metric updating scheme, and checks for positive-definiteness. Its main weakness is that it depends heavily on knowledge of the first derivatives, and fails miserably if they are very inaccurate.

For further details see MINUIT documentation [35].

2.2.2. Robust Fitting. The least-squares fitting involves the minimization of the sum of the squared residuals. There are two instances where this minimization produces less than satisfactory fit:

1. The significant outliers are present in data. In this case, the square of the residuals of these out-lier points may, within a given region, significantly shift the fitted curve away from the bulk of the data.

2. The the Y-data span more than several orders of magnitude. The squared residuals of the largest valued Y-points can overwhelm the influence of the squared residuals of the smallest Y-valued points, causing the smallest Y-value points to either be poorly fitted or not fitted at all. Data that require a logarithmic Y-scale to see all of the points may be a good candidate for robust fitting, especially if four or more major log divisions take place.

Robust estimates designed to be successful in such cases. The essence of robust fitting is to use a minimization that is less influenced by outliers and the dynamic range of the Y-variable. Each data measured point took into account with it own weight, which indicate influence of given point.

It is based on so-called M-estimates which follow from maximum like-hood approach, M-estimates are usually the most relevant for model fitting. Robust approach uses gross-error model (Hubert). Probability distribution function of measurement errors \( d_i \) suggested as a superposition of two distributions: basic \( g(d_i) \) and distribution of big errors \( h(d_i) \)

\[
D(d_i) = (1 - \varepsilon)g(d_i) + \varepsilon h(d_i), \quad \varepsilon \in [0, 1]
\]

\[
g(d_i) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{d_i^2}{2\sigma^2}\right)
\]

\[
h(d_i) = \alpha \exp\left(-\beta |d_i|\right)
\]

In practice, robustness works also in cases of other distributions \( h(d_i) \), for example, normal one like \( g(d_i) \) is, but with bigger RMS.

Application of maximum like-hood approach and few simplifications and the fact that RMS of the distribution \( g(d_i) \) can be approximated by chi-square of data points, give us well known Tukey bi-square weights, which is suitable in most cases.

\[
w_i = \begin{cases} 
1 - \left(\frac{f(x_i) - y_i}{c_\tau \chi^2}\right)^2, & |f(x_i) - y_i| \leq c_\tau \chi^2 \\
0, & |f(x_i) - y_i| > c_\tau \chi^2 
\end{cases}
\] (1)

In chi-square function data values are multiplied by their weights:
\[ \chi^2 = \frac{1}{\sum_{i=1}^{N} w_i - N_{\text{params}}} \sum_{i=1}^{N} w_i \left( \frac{f(x_i) - y_i}{\Delta y_i} \right)^2 \]  

(2)

The initial values of the weights are equal to 1. In the following iterations, weights are recalculated after each procedure of minimization and calculation of the new value of chi-square. Iterations are repeated until convergence (until chi-square value is stabilized within a predefined accuracy).

In general, the robust approach looks like:

- setting \( w_i = 1 \), calculating \( \chi^2 \) by (2), setting \( c_r = 5 \);
- while \( c_r > 1 \) doing the loop:
  - calculating \( w_i \) by (1) using current values of \( \chi \) and \( c_r \);
  - minimizing \( \chi^2 \) given by (2);
  - reducing \( c_r \) by \( c_r / = 2 \).

For detailed explanation see [38-40].

3. Installation

3.1. Installation on UNIX systems

In order to compile and install Fitter on your system, type the following in the base directory of the Fitter distribution:

```
% ./configure
% make
% make install
```

To build Fitter with GUI you should install FLTK [41] first. Second, you should install Gluplot plotting package [42], since it is used for visualization. Finally, enable using FLTK libraries while Fitter configuring:

```
% ./configure –enable-fltk
```

Since Fitter uses autoconf you should have not trouble compiling it. Should you run into problems please report them to the the authors Alexei Soloviev and Alexei Stadnik.

3.2. Installation on Windows

Fitter provides an user-friendly setup procedure for Windows users. Just run setup and follow the instructions.

4. Usage

4.1. General Usage

*Fitter* is a C++ program aimed to fit a chosen theoretical multi-parameter function through a set of data points. The method of fitting is chi-square minimization. Moreover, the robust fitting method can be
applied in Fitter. Fitter was designed to be used for a small-angle neutron scattering data analysis. Respective theoretical models are implemented in it. Some commonly used models (Gaussian and polynomials) are also implemented for wider applicability.

4.1.1. Synopsis.

```
fitter -- { std | sans | yumo } model [ -s name ] [ -r ] [ -R x1:x2 ] [ -v file ] [ -o file ] [ -H file ] [ --rc-read ] [ --rc-write ] file
fitter [ -h ] [ -V ]
```

4.1.2. Input. Input data file (ASCII) for fitter should have at least three columns: the first one would be interpreted by fitter as "X"-samples, second one - as "Y"-samples and the third one - as "Y"-errors. For YuMO models (see below) should be one more column, which is interpreted as "X"-errors.

4.1.3. Options.

MODEL OPTIONS

```
--std model Standard models. One of the following (see `--std help' for list):
  g Gaussian
  e Exponential
  p0 Polynomial of the order 0
  p1 Polynomial of the order 1
  p2 Polynomial of the order 2
  p3 Polynomial of the order 3
  p4 Polynomial of the order 4
  p5 Polynomial of the order 5
  p6 Polynomial of the order 6
  p7 Polynomial of the order 7
  p8 Polynomial of the order 8
```

```
--sans model SANS models. One of the following (see `--sans help' for list):
  b Ball
  c Cylinder
  c2 Elliptical cylinder
  e2 Two axis ellipsoid
  e2s Two axis ellipsiodal shell
  e3 Three axial ellipsoid
  e3s Triaxial ellipsiodal shell
  p Parallelepiped
  ss Spherical shell
```

```
--yumo model SANS models for YuMO spectrometer. One of the following (see `--yumo help' for list):
  b Ball with resolution
  c Cylinder with resolution
```
Elliptical cylinder with resolution $e_2$

Two axis ellipsoid with resolution $e_2s$

Two axis ellipsiodal shell with resolution $e_3s$

Three axial ellipsoid with resolution $e_3$

Triaxial ellipsiodal shell with resolution $p$

Parallelepiped with resolution $ss$

Spherical shell with resolution $s$

**PROCESSING OPTIONS**

- `-s, --strategy name` Change MINUIT strategy. MINIMIZE is a default one.
- `-r, --robust` Robust fitting. Useful for highly noised data.
- `-R, --range x1:x2` Fitting in range from $x_1$ to $x_2$. Note that no fit is applied if a range appears to be empty.
- `-v, --verbose [file]` Output MINUIT processing information. No output is produced by default. Repeating this option determines how much output will be produced:
  (a) minimum MINUIT output
  (b) normal MINUIT output
  (c) additional output giving intermediate results
  (d) maximum output, showing progress of minimizations

**OUTPUT OPTIONS**

- `-o, --output file` Output data file with the resulting theoretical curve. No file is created by default.
- `-H, --header file` Output resulting parameters and chi-square value to file (standard output by default). File may be the same as for theoretical curve, parameters and chi-square will be put to its beginning in this case.
- `-rc-read` Read rc-file. If specified, this option cause reading parameters starting values from rc-file. Some times this can highly speed-up minimization. Re-file .fitter should exist in $HOME$ (in C: on Windows).
- `-rc-write` Write rc-file. If specified, this option cause writing resulting parameters to rc-file. Useful for creating/repairing rc-file. Re-file .fitter is placed in $HOME$ (in C: on Windows).

**HELP OPTIONS**

- `-h, --help` Display short help message and exit.
- `-V, --version` Display version information and exit.

### 4.2. GUI

Besides the command-line interface the Fitter offers an user friendly GUI (graphical user interface). Fitter's main window, shown in figure 2, is implemented in the standard manner with menu-bar, tool-bar and status-bar. All menu items exactly correspond to the command-line options described above. The command-line to be executed is just formed by choosing corresponding menu items and is displayed on the status-bar. Some of the most frequently used menu items are duplicated as buttons or combo-boxes on the tool-bar. A window with fitting results arises after fit is complete.
5. Examples

5.1. Robust feature example
Here is a simple example of the Robust feature usage. File test.dat is an example of input data for fitter. It is just a generated gaussian distribution, contaminated by uniformly distributed random noise.

Figure 2. Fitter's main window.

Figure 3. Highly noisted data fitting example.
Typing in command line "% fitter test.dat –o fit1.dat –std g"
we get the following result:

Model.................................................................................................................................................Gaussian

| Mean                                      | -0.4576 +/- 0.1719 |
|-------------------------------------------|--------------------|
| RMS                                       | 1.7571 +/- 0.3680  |
| Amplitude                                 | 7.1333 +/- 0.5719  |
| Background                                | 17.3032 +/- 0.6111 |

Chi square........................................................................................................................................1.1068

and the fitting curve shown in figure 3 (left).

This fit is not good enough, thus, we try "% fitter test.dat –o fit2.dat –std g -r" invoking robust fitting.
This gets a better result:

Model..............................................................................................................................................Gaussian

| Mean                                      | -0.2564 +/- 0.1595 |
|-------------------------------------------|--------------------|
| RMS                                       | 0.9985 +/- 0.1638  |
| Amplitude                                 | 7.8680 +/- 1.2633  |
| Background                                | 18.6319 +/- 0.5641 |

Chi square........................................................................................................................................1.0708

The fitting curve (figure 3, right) is much better this time.

5.2. SANS/YuMO example
To demonstrate the fitter usage, we have used the results of neutron measurements, obtained on the
apo ferritin protein sample obtained from Aldrich corporation. The file apdn.dat containing a SANS spectra is used as fitter input. A spherical shell model, which is one of the most adequate for apo ferritin, is chosen. The corresponding command line is the following: "% fitter apdn.dat –o apdn.fit –yumo ss".

This gives the following result:

Model...........................................................................................................................................Spherical shell with resolution

| Outer radius                              | 59.6547 +/- 0.0576 |
|-------------------------------------------|--------------------|
| Inner radius                              | 40.1314 +/- 0.0700 |
| Amplitude                                 | 28.3414 +/- 0.1721 |
| Background                                | 0.0191 +/- 0.0003  |

Chi square31.2243

and the fitting curve shown in figure 4.
6. Conclusion

Fitter is used as a part of a program complex on spectrometer YuMO. It is designed to be easily understood by the experimenters. Both command line interface and GUI are provided. Despite of simplicity of offered models wide usage of fitter for many major appendices is supposed. The further development of the program is connected to expansion of offered models and introduction in the program of processing of the account of the structural factor. Fitter's design allows the simple adding of new models. The program is demanded by users due to using the resolution of YuMO spectrometer also.

References

[1] Library "JINRLIB", FITTER. http://wwwinfo.jinr.ru/programs/jinrlib/fitter/indexe.html
[2] Soloviev A G, Stadnik A V, Islamov A Hand Kuklin A I 2003 FITTER: The package for fitting experimental data of the YuMO spectrometer by theoretical form-factors. Version 1.0. Long Write-Up and User's Guide. (Dubna: JINR) p
[3] Soloviev A G, Stadnik A V, Islamov A Kand Kuklin A I 2008 Fitter. The package for fitting a chosen theoretical multi-parameter function through a set of data points. Application to experimental data of the YuMO spectrometer. Version 2.1.0. Long Write-Up and User's Guide. (Dubna: JINR) p
[4] Hughes D J, Wallace J Rand Holtzman R H 1948 Physical Review 73 1277
[5] Svergun D 1993 Journal of Applied Crystallography 26 258
[6] Svergun D 1994 Acta Crystallographica Section A 50 391
[7] Svergun D 1992 Journal of Applied Crystallography 25 495
[8] Svergun D, Barberato Cand Koch M H J 1995 Journal of Applied Crystallography 28 768
[9] Svergun D I 1999 Biophys J 76 2879
[10] Konarev P V, Petoukhov M Vand Svergun D I 2001 Journal of Applied Crystallography 34 527
[11] Kozin M Band Svergun D I 2000 Journal of Applied Crystallography 33 775
[12] Frank laboratory of neutron physics. http://flnp.jinr.ru/34/
[13] Kuklin A I, Islamov A Kand Gordeliy V I 2005 Neutron News 16 16
[14] Kozin M Band Svergun D I 2000 Journal of Applied Crystallography 33 775
[15] Murugova T N, Solodovnikova I M, Yurkov V I, Gordeliy V I, Kuklin A I, Ivankov O I, Kovalev Y S, Popov V I, Teplova V Vand Yaguzhinsky L S 2011 Neutron News 22 11
[16] Molchanov V S, Philippova O E, Khokhlov A R, Kovalev Y Aand Kuklin A I 2006 Langmuir 23 105
[17] Kosacheva E, Kudryavtsev D, Bakeeva R, Kuklin A, Islamov A, Kudryavtseva L, Sopin Vand Konovalov A 2006 Colloid Journal 68 713
[18] Haramagatti C R, Islamov A, Gibhardt H, Gorski N, Kuklin Aand Eckold G 2006 Physical Chemistry Chemical Physics 8 994
[19] Islamov A, Haramagatti C R, Gibhardt H, Kuklin Aand Eckold G 2006 Physica B: Condensed Matter 385-386, Part 1 791
[20] Andreeva S A, Fomenkov I, Islamov A K, Kuklin A I, Filippova O Eand Khokhlov A R 2005 Polymer Science A 47 194
[21] Ozerin A N, Svergun D I, Volkov V V, Kuklin A I, Gordelij V I, Islamov A K, Ozerina L Aand Zavorotnyuk D S 2005 Journal of Applied Crystallography 38 996
[22] Kříž J, Pleštil J, Pospišil H, Kadlec P, Koňák Č, Almásy Land Kuklin A I 2004 Langmuir 20 11255
[23] Lebedev D, Filatov M, Kuklin A, Islamov A, Stellbrink J, Pantina R, Denisov Y, Toperverg Band Isaev-Ivanov V 2008 Crystallography Reports 53 110
[24] Kovalev Y, Levkovich N, Kuklin Aand Apel’ P 2009 Colloid Journal 71 634
[25] Bakeeva R, Kosacheva E, Razina I, Kudryavtsev D, Kovalev Y, Kuklin A, Kudryavtseva Land Sopin V 2008 Kinetics and Catalysis 49 631
[26] Herec M, Islamov A, Kuklin A, Gagos Mand Gruszecki W I 2007 Chem Phys Lipids 147 78
[27] Anitas E M, Balasoiu M, Bica I, Osipov V Aand Kuklin A I 2009 Optoelectronics and Advanced Materials - Rapid Communications 3 621
[28] Balasoiu M, Anitas E M, Bica I, Osipov V A, Orelovich O L, Savu D, Savu S, Erhan Rand Kuklin A I 2008 Optoelectronics and Advanced Materials - Rapid Communications 2 730
[29] Balasoiu M, Craus M, Anitas E, Bica I, Plestil Jand Kuklin A 2010 Physics of the Solid State 52 917
[30] Balasoiu M, Craus M L, Kuklin A I, Plestil J, Haramus V, Islamov A H, Erhan R, Anitas E M, Lozovan M, Tripadus V, Petrescu C, Savu D, Savu Sand Bica I 2008 Journal of Optoelectronics and Advanced Materials 10 2932
[31] Kuklin A I, Kutuzov S A, Gabriel A, Eckold G, Utrobin P K, Smirnov A A, Ivankov A, Islamov A K, Kovalev Y S, Feoktistov A, Rogachev A V, Kirilov A Sand Gordeliy V I. 2007. Measurements with New Type of Position Sensitive Detector in Backscattering and Small-angle Setup, Dubna

[32] Eckold Gand Gorski N 2001 Colloids and Surfaces A: Physicochemical and Engineering Aspects 183-185 361

[33] Rogachev A V, Cherny A Y, Ozerin A N, Gordeliy V Iand Kuklin A I 2007 Crystallography Reports 52 500

[34] Lebedev D V, Baitin D M, Islamov A K, Kuklin A I, Shalguev V K, Lazov V Aand Isaev-Ivanov V V 2003 FEBS Letters 537 182

[35] CERN Program Library Long Writeup D506 MINUIT http://wwwasdoc.web.cern.ch/wwwasdoc/minuit/

[36] Feigin L Aand Svergun D I 1987 Structure Analysis by Small-Angle X-Ray and Neutron Scattering (New York: Plenum Press) p 335

[37] Ostanevich Y M 1988 Makromolekulare Chemie. Macromolecular Symposia 15 91

[38] Huber P 1981 Robust statistics (New-York: John Wiley and Sons) p

[39] Ososkov G. Robust regression for the heavy contaminated sample. Proc. 2nd International Tampere Conference in Statistics, Tampere, Finland, 1987:

[40] Chernov N, Kolganova Eand Ososkov G 1999 Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 433 274

[41] Fast Light Toolkit. http://www.fltk.org

[42] The Gluplot Plotting Package. http://wwwinfo.jinr.ru/programs/jinrlib/gluplot/indexe.html