

E10-2003-36

A. G. Soloviev¹, A. V. Stadnik, A. H. Islamov, A. I. Kuklin

FITTER².

**THE PACKAGE FOR FITTING EXPERIMENTAL DATA
OF THE YuMO SPECTROMETER
BY THEORETICAL FORM FACTORS.**

Version 1.0. Long Write-Up and User's Guide

¹E-mail: solovjev@lynx.jinr.ru

²Fitter web site: <http://www.jinr.ru/programs/jinrlib/fitter/indexe.html>

1 Introduction

The method of small-angle neutron scattering (SANS) [1, 2] is frequently used for research of fundamental and applied problems in condensed matter physics, biology and chemistry. At JINR such experiments are performed at the Frank Laboratory of Neutron Physics. The major feature of small-angle scattering is the opportunity of the analysis is the disorder systems structure. This method is often an only way to obtain direct structural information of systems with disordered and partly ordered arrangement of the nuclear density with characteristic sizes of about 10-10000 angstroms.

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 $F(q)$. Certainly, the nearby particles can essentially influence a scattering curve. The contribution of inter-partial effects is described with the help of the so-called structural factor while the effect of the particle is expressed by a form-factor. 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 [3].

Nevertheless, for the charged particles without shield and for any concentrated solutions it is impossible to neglect the structural factor. There are some ways to calculate the structural factor of the charged particles using various approaches of the Opshtein-Cernike equation. The analytical solution of the structural factor was obtained by Haiter and Penfold.

At the same time, in spite of the wide SANS method application and already more than thirty-year 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 / momentum transfer by various theoretical models. The small-angle neutron spectrometer YuMO after its modernization is actively used for many structural problems by scientists from Russia and other countries all over the world [3, 4, 5, 6, 7, 8, 9].

The existing program complex on YuMO spectrometer has been created for many years by different people. It is written mostly in the old-fashioned Fortran language and suffers from the lack of instructive manuals. Therefore, a real need arises now to update the YuMO software. The following should be done under this update:

- taking into account the new spectrometer setup;
- rewriting software in the modern object-oriented form (i.e. C++);
- making programs more “user friendly”;
- providing each program by the necessary long write-up manual.

This handbook gives a long write-up description of the program for fitting the curves received as a result of experiment on the small-angle spectrometer YuMO. As examples, some of its applications for samples without interaction are presented.

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 [12] procedures in the current version. The important feature of Fitter’s design is its expandability: both new models and new minimising algorithms can easily be added to the existing ones.

2.1 Theoretical Models

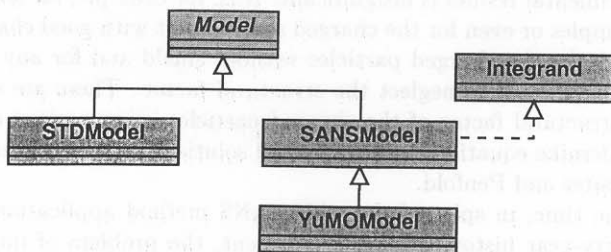


Figure 2.1: Fitter’s model module class hierarchy

Fitter’s model module is designed as follows (fig.2.1). 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**

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

- **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**

$$\begin{aligned} I(Q) &= A \Phi^2(QR) + B \\ \Phi(t) &= 3 \frac{\sin t - t \cos t}{t^3} \end{aligned}$$

- **Ellipsoid of revolution with half-axes a and av**

$$\begin{aligned} I(Q) &= A \int_0^1 \Phi^2 \left[Qa \sqrt{1 + x^2(v^2 - 1)} \right] dx + B \\ \Phi(t) &= 3 \frac{\sin t - t \cos t}{t^3} \end{aligned}$$

- **Cylinder of the radius R and the length H**

$$\begin{aligned} I(Q) &= A \int_0^1 \Lambda_1^2 \left(QR \sqrt{1 - x^2} \right) S^2(QHx/2) dx + B \\ \Lambda_1(t) &= 2J_1(t)/t \\ S(t) &= \sin t/t \end{aligned}$$

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

$$\begin{aligned}
 I(Q) &= A \int_0^1 \Psi_{EC}(Q, a\sqrt{1-x^2}) S^2(QHx/2) dx + B \\
 \Psi_{EC}(Q, a) &= \frac{1}{\pi} \int_0^\pi \Lambda_1^2 \left(Qa \sqrt{\frac{1+v^2}{2} + \frac{1-v^2}{2} \cos y} \right) dy \\
 \Lambda_1(t) &= 2J_1(t)/t \\
 S(t) &= \sin t/t
 \end{aligned}$$

- **Parallelepiped** with the ribs a , b , c

$$\begin{aligned}
 I(Q) &= A \int_0^1 \Psi_p(Q, b\sqrt{1-x^2}, a) S^2(Qbcx/2) dx + B \\
 \Psi_p(Q, b, a) &= \frac{2}{\pi} \int_0^{\pi/2} S^2(Qa \sin y/2) S^2(Qb \cos y/2) dy \\
 S(t) &= \sin t/t
 \end{aligned}$$

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

$$\begin{aligned}
 I(Q) &= A \left[\Phi(QR_1) - \left(\frac{R_2}{R_1} \right)^3 \Phi(QR_2) \right]^2 + B \\
 \Phi(t) &= 3 \frac{\sin t - t \cos t}{t^3}
 \end{aligned}$$

More detailed information about SANS models is available in [1].

2.1.3 SANS Models with YuMO spectrometer resolution

These models are implemented to fit data measured on the YuMO spectrometer [10] operated on the 4-th channel of the fast pulsed reactor IBR-2 [11]. YuMO models are the same as SANS, but they take into account the spectrometer resolution.

- **SANS Model with resolution**

$$\begin{aligned}
 I(Q) &= \frac{1}{\sqrt{2\pi}\sigma} \int_{Q-3\sigma}^{Q+3\sigma} I_{SANS}(q) \exp \left\{ -\frac{1}{2} \left(\frac{q-Q}{\sigma} \right)^2 \right\} dq \\
 \sigma &= \sqrt{2\Delta Q}
 \end{aligned}$$

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:

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

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 MINIGRAD 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 [12].

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 $h(d_i)$

$$\begin{aligned}
 D(d_i) &= (1 - \epsilon)g(d_i) + \epsilon h(d_i), \quad \epsilon \in [0, 1] \\
 g(d_i) &= \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{\frac{-d_i^2}{2\sigma^2}\right\} \\
 h(d_i) &= \alpha \exp\{-\beta|d_i|\}
 \end{aligned}$$

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.

$$\begin{aligned}
 w_i &= \begin{cases} \left(1 - \frac{|f(x_i) - y_i|^2}{c_T\chi^2}\right)^2, & |f(x_i) - y_i| \leq c_T\chi^2 \\ 0, & |f(x_i) - y_i| > c_T\chi^2 \end{cases} \\
 c_T &= 5
 \end{aligned}$$

In chi-square function data values are multiplied by their weights:

$$\chi^2 = \frac{1}{\sum_{i=1}^N w_i - N_{\text{parms}}} \sum_{i=1}^N w_i \left(\frac{f(x_i) - y_i}{\Delta y_i} \right)^2$$

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).

$w_i^{(0)} \rightarrow \text{minimization} \rightarrow \chi^2 \rightarrow w_i^{(1)} \rightarrow \text{minimization} \rightarrow \chi^2 \rightarrow w_i^{(2)} \dots$

For detailed explanation see [13, 14, 15].

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
% gmake
% gmake install
```

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 **-{sans|yumo|std}model** [-r] [-v] [-ofile] [-Hfile] [-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

-sans model SANS models. One of the following (see '**-sans help**' for list):

- b* Ball
- c* Cylinder
- c2* Elliptical cylinder
- e2* Two axis ellipsoid
- 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
- c2* Elliptical cylinder with resolution
- e2* Two axis ellipsoid with resolution
- p* Parallelepiped with resolution
- ss* Spherical shell with resolution

-std model Standard models. One of the following (see '**-std help**' for list):

- g* Gaussian
- 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

PROCESSING OPTIONS

- r, -robust** Robust fitting. Useful for highly noised data.
- v, -verbose** Output MINUIT processing information. No output is produced by default. Repeating this option determines how much output will be produced:
 1. minimum MINUIT output
 2. normal MINUIT output
 3. additional output giving intermediate results
 4. 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. Rc-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. Rc-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 (Windows only)

For Windows users the Fitter offers an user friendly GUI (graphical user interface). Fitter's main window, shown in fig.4.2, is implemented in the standard for Windows applications 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.

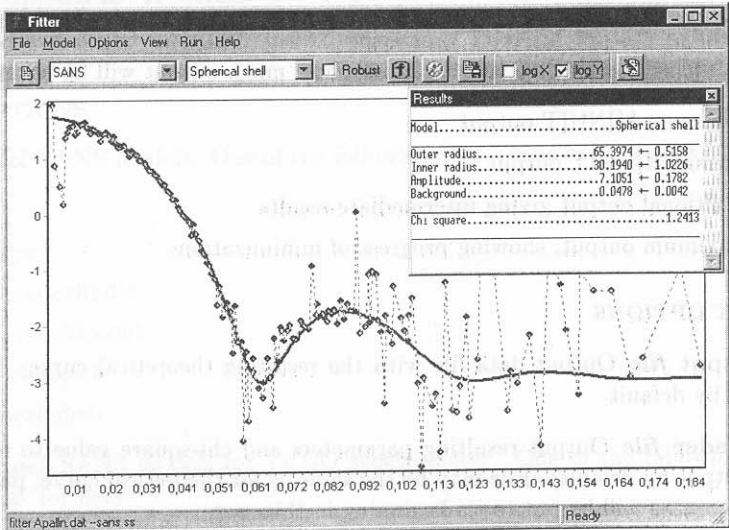


Figure 4.2: Fitter's main window

4.3 Examples

4.3.1 SANS/YuMO example

To demonstrate the fitter usage, we have used the results of neutron measurements, obtained on the apoferritin 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 apoferritin, is chosen. The corresponding command line is the following:

```
% fitter apdn.dat -o apdn.fit --sans ss
```

This gives the following result:

```

-----
Model.....Spherical shell
-----
Outer radius.....65.3974 +- 0.5158
Inner radius.....30.1940 +- 1.0226
Amplitude.....7.1051 +- 0.1782
Background.....0.0478 +- 0.0042
-----
Chi square.....1.2413
-----

```

and the fitting curve shown in fig.4.3.

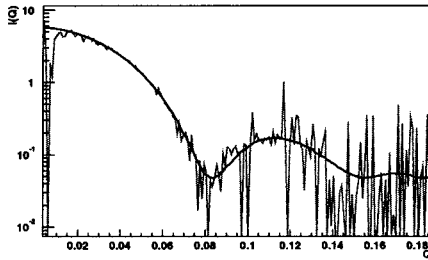


Figure 4.3: Apoferritin data fitting example

4.3.2 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.

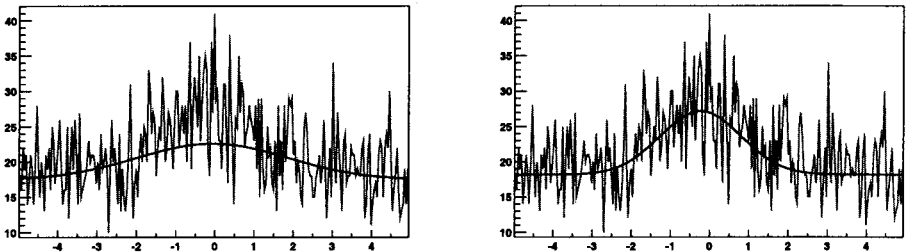


Figure 4.4: Highly noisted data fitting example

Typing in command line

```
% fitter test.dat -o fit1.dat --std g
```

we get the following result:

```
-----  
Model.....Gaussian  
-----  
Mean.....-0.0645 +- 0.2349  
RMS.....1.9471 +- 1.5077  
Amplitude.....5.1483 +- 4.4472  
Background.....17.4474 +- 4.7152  
-----  
Chi square.....1.1525  
-----
```

and the fitting curve shown in fig.4.4 (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.....-0.2332 +- 0.1171  
RMS.....1.0345 +- 0.1316  
Amplitude.....9.0263 +- 0.9474  
Background.....18.1460 +- 0.4043  
-----  
Chi square.....1.0352  
-----
```

The fitting curve (fig.4.4, right) is much better this time.

5 Conclusion

Fitter is a first step on development of a program complex on spectrometer YuMO. It is designed to be easily understood by the experimenters. Both UNIX standard command line interface and Windows 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.

Acknowledgements

The authors would like to thank Dr. G.Ososkov for his collaboration, fruitful discussions and some useful remarks.

References

- [1] *Feigin, L.A. & Svergun, D.I.* (1987) Structure analysis by small-angle X-ray and neutron scattering. New York: Plenum Press, 335 pp.
- [2] *V.I.Gordeliy, A.I.Kuklin*, Small-angle neutron scattering on reactor IBR-2 (in russian). JINR Commn. **P13-2002-250**
- [3] *G.Eckold, N.Gorski*, Small-angle neutron scattering from tetradecyltrimethylammonium bromide in NaBr aqueous solutions. Colloids Surfaces A: Physicochemical and Engineering Aspects, 2001, 183-185, 361-369
- [4] *Yu.D.Zaroslov, V.I.Gordeliy, A.I.Kuklin, A.H.Islamov, O.E.Philippova, A.R.Khokhlov and G.Wegner*, Self-Assembly of Polyelectrolyte Rods in Polymer Gel and in Solution: Small-Angle Neutron Scattering Study. Macromolecules 2002, **35**, 4466-4471
- [5] *O.A.Bannyh, V.M.Blinov, A.I.Kuklin, V.A.Semenov, V.V.Sumin, A.V. Tamonov*, Studying of disintegration nitrogenous austenith alloy X24A using methods of neutrons scattering (in russian). Metals, N.5, 55-59, 2002
- [6] *J. Plestil, H. Pospisil, P.Kadlec, Z.Tuzar, J.Kriz, V.I.Gordeliy*, SANS study of multilayer nanoparticles based on block copolymer micelles. Polymer **42** (2001) 2941-2946
- [7] *N.Gorski, J.Kalus*, Temperature dependence of the sizes of tetradecyltrimethylammonium bromide Micelles in aqueous solutions, Langmuir. 2001, **17**, 4211-4215
- [8] *D.Uhrikova, N. Kucerka, A. Islamov, V. Gordeliy, P. Balgavy*, Small-Angle Neutron Scattering Study of N-Dodecyl-N, N-dimethylamine N-Oxide Induced Solubilization of Dioleoylphosphatidylcholine Bilayers in Liposomes. General Physiology and Biophysics **20** (2001) 183-189
- [9] *Bakeeva R.F., Kosacheva E.M., Kudryavtsev D., Kudryavtseva L.A., Raevska A., Kuklin A., Islamov A.Kh. and V.F. Sopin*, Substrate specificity of hydrolysis of phosphor ether in hexagonal mesophase of the system cetyldimethylamine bromide/NaOH/water. Liquid Cryst. and Applications V.1 (2001) pp.31-34.
- [10] *Yu.M.Ostanevich*, Time-of-flight small-angle scattering on pulsed neutron sources. Makromol. Chem., Macromol. Symp. **15**, 91-103 (1988)

- [11] <http://nfdfn.jinr.ru/ibr2.html>
- [12] <http://wwwinfo.cern.ch/asdoc/minuit/minmain.html>
- [13] *P.Huber*, Robust statistics. Wiley, New-York (1981)
- [14] *G.Ososkov*, Robust regression for the heavy contaminated sample. Proc. 2-nd International Tampere Conference in Statistics (1987) (Tampere, Finland)
- [15] *N.Chernov*, *E.Kolganova*, *G.Ososkov*, Robust Methods for the RICH Ring Recognition and Particle Identification. Nuclear Inst. Meth. A433 (1999) 247-278.

Received on February 26, 2003.

Соловьев А. Г. и др.

E10-2003-36

FITTER.

Программа для подгонки экспериментальных данных спектрометра YuMO с помощью теоретических формфакторов. Версия 1.0. Описание и руководство пользователя

FITTER — программа на языке C++, предназначенная для фитирования экспериментальных данных заданной многопараметрической функцией. Метод заключается в минимизации функции хи-квадрат. Дополнительно можно использовать робастный метод. Программа FITTER создана для анализа данных малоуглового рассеяния нейтронов, и в ней реализованы соответствующие теоретические модели. Для более широкой применимости добавлены также некоторые общепринятые модели (гауссиан и полиномы).

Работа выполнена в Лаборатории нейтронной физики им. И. М. Франка ОИЯИ.

Сообщение Объединенного института ядерных исследований. Дубна, 2003

Soloviev A. G. et al.

E10-2003-36

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

FITTER is a C++ code 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. FITTER was designed for a small-angle neutron scattering data analysis, using respective theoretical models. Commonly used models (Gaussian and polynomials) are also implemented for wider applicability.

The investigation has been performed at the Frank Laboratory of Neutron Physics, JINR.

Communication of the Joint Institute for Nuclear Research. Dubna, 2003

Макет *Т. Е. Попеко*

Подписано в печать 19.03.2003.

Формат 60 × 90/16. Бумага офсетная. Печать офсетная.

Усл. печ. л. 1,18. Уч.-изд. л. 2,05. Тираж 300 экз. Заказ № 53819.

Издательский отдел Объединенного института ядерных исследований
141980, г. Дубна, Московская обл., ул. Жолио-Кюри, 6.

E-mail: publish@pds.jinr.ru

www.jinr.ru/publish/