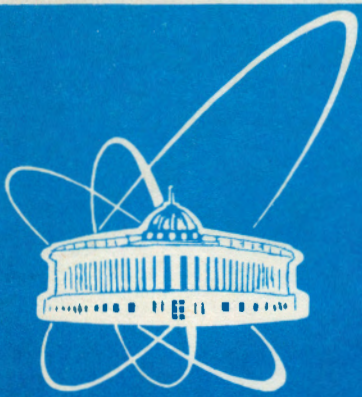

ОБЪЕДИНЕННЫЙ
ИНСТИТУТ
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ

Дубна

98-318

E10-98-318

S.N.Dymov, V.S.Kurbatov, I.N.Silin, S.V.Yaschenko

CONSTRAINED MINIMIZATION IN *C++* ENVIRONMENT

Submitted to «Nuclear Instruments and Methods A»

1998

1 Introduction

In this parer we describe two realizations (in C++ language) of constrained minimization for χ^2-like functionals. One of them is the algorithm of the FUMILI code, which was available for users as a part of CERN library [1]. The description of this algorithm was published in Russian [2] at the end of the 1960s. Due to the fact that the access to this publication is not easy for an English reader, we give a short description of the FUMILI algorithm. This algorithm is now coded in the C++ language.

The second part is the realization of the idea proposed by one of the authors (I.N. Silin) for solving the constrained minimization problem in a general case, when constraints are of arbitrary type (arbitrary equalities and inequalities) [3]. Technically, here constraints are taken into account by the method of penalty functions (though there are other ways of doing it [3]). The algorithm described below was tested on the model data for the calibration process $pp \to d\pi^+$ under the conditions of the ANKE setup [4].

2 Algorithm of FUMILI

For simplicity, let us assume that the function to be minimized has the form [1]

$$\chi^2 = \frac{1}{2}\sum_{j=1}^{n}\left(\frac{f_j(\vec{x}_j,\vec{\theta}) - F_j}{\sigma_j}\right)^2, \qquad (1)$$

where $f_j(\vec{x}_j,\vec{\theta})$ are the measured functions at the points $\vec{x}_j$, F_j are the values of the measured functions, σ_j are their errors, $\vec{\theta}$ are parameters to be estimated.

The minimum condition is

$$\frac{\partial\chi^2}{\partial\theta_i} = \sum_{j=1}^{n}\frac{1}{\sigma_j^2}\cdot\frac{\partial f_j}{\partial\theta_i}\left[f_j(\vec{x}_j,\vec{\theta}) - F_j\right] = 0, \qquad i = 1 \div m, \qquad (2)$$

where m is the number of parameters.

Expanding the left side of eq. 2 in parameter increments and retaining only linear terms we get

$$\left(\frac{\partial\chi^2}{\partial\theta_i}\right)_{\vec{\theta}=\vec{\theta}^0} + \sum_k\left(\frac{\partial^2\chi^2}{\partial\theta_i\partial\theta_k}\right)_{\vec{\theta}=\vec{\theta}^0}\cdot(\theta_k - \theta_k^0) = 0.$$

Here $\vec{\theta}^0$ is some initial value of parameters. In a general case:

$$\frac{\partial^2\chi^2}{\partial\theta_i\partial\theta_k} = \sum_{j=1}^{n}\frac{1}{\sigma_j^2}\cdot\frac{\partial f_j}{\partial\theta_i}\cdot\frac{\partial f_j}{\partial\theta_k} + \sum_{j=1}^{n}\frac{(f_j - F_j)}{\sigma_j^2}\cdot\frac{\partial^2 f_j}{\partial\theta_i\partial\theta_k}. \qquad (3)$$

[1] all the following can be easily generalized to the case where the covariance matrix of the function f_j has non-diagonal terms

In the FUMILI algorithm an approximate expression for $\partial^2\chi^2/\partial\theta_i\partial\theta_k$ is used when the last term in eq. 3 is discarded (it is often done, not always wittingly, and sometimes causes trouble), i.e.:

$$\frac{\partial^2\chi^2}{\partial\theta_i\partial\theta_k} \cong Z_{ik} = \sum_{j=1}^{n}\frac{1}{\sigma_j^2}\cdot\frac{\partial f_j}{\partial\theta_i}\cdot\frac{\partial f_j}{\partial\theta_k}.$$

Then the equations for parameter increments are

$$\left(\frac{\partial\chi^2}{\partial\theta_i}\right)_{\vec{\theta}=\vec{\theta}^0} + \sum_k Z_{ik}\cdot(\theta_k - \theta_k^0) = 0, \qquad i = 1 \div m.$$

A remarkable feature of the algorithm is the technique for step restriction. For an initial value of the parameter $\vec{\theta}^0$ a parallelepiped P_0 is built with the center at $\vec{\theta}^0$ and axes parallel to the coordinate axes θ_i. The lengths of the parallelepiped sides along the i-th axis are $2\cdot b_i$, where b_i is such a value that the functions $f_j(\vec{\theta})$ are quasi-linear all over the parallelepiped. If the step $\Delta\vec{\theta}$ gives a new point $\vec{\theta}^1 = \vec{\theta}^0 + \Delta\vec{\theta}$ outside P_0, the crossing $\vec{\theta}^1$ of the vector $\Delta\vec{\theta}$ with the surface of P_0 is found and taken as a new value for the parameter. After selection of the new value for the parameter, it is checked, whether the function reduction is big enough compared with the expected on the quadratic approximation. If it is not, the step reduction is performed. Some parallelepiped lengths can be increased too.

In addition, FUMILI takes into account simple linear inequalities in the form:

$$\theta_i^{\min} \le \theta_i \le \theta_i^{\max}. \qquad (4)$$

They form a parallelepiped P (P_0 may be deformed by P). If the value of the parameter lies on the surface of P and the gradient component is such that χ^2 is not going to increase outside P, the corresponding parameter is fixed.

Then the step is calculated for all non-fixed parameters and if some parameters, lying on the surface of P, go beyond P, one of them is temporary fixed too (the parameter, for which the ratio $|\Delta\theta_i|/\sqrt{(Z^{-1})_{ii}}$ is maximal) and so on.

The criterion for the end of the iteration process is the requirement that all parameters are fixed due to only the gradient sign and step increments for non-fixed parameters

$$|\Delta\theta_i| < \varepsilon\cdot\sqrt{(Z^{-1})_{ii}},$$

where ε is a small figure ~ 0.01. Because the number of fixation combinations is finite, the number of steps will be finite, at least in the convex quadratic case.

Very similar step formulae are used in FUMILI for the negative logarithm of the likelihood function with the same idea of linearizing the functional argument.

3 Minimization of χ^2 functionals with arbitrary constraints

3.1 Formulation of the problem

Again, let us assume that the function to be minimized has the same form eq. 1, but in addition to simple linear constraints (eq. 4) there are two more types of constraints: nonlinear inequalities and equalities

$$a_r \le \phi_r(\vec{\theta}) \le b_r, \qquad r = 1 \div m_d, \tag{5}$$

$$\psi_s(\vec{\theta}) = c_s, \qquad s = 1 \div m_e. \tag{6}$$

Here $\phi_r(\vec{\theta})$, $\psi_s(\vec{\theta})$ are the regular functions of the parameter $\vec{\theta}$; a_r, b_r, m_d are the low and upper boundaries of the inequalities and their number; c_s, m_e — any constant and number of equations. Here regularity is taken to mean continuous second-order derivatives. The problem of taking into account the constraints in the form of the equalities of type (eq. 6) was solved before [5, 6, 7]. As for the constraints in the form of inequalities (eq. 5), the authors did not know a simple solution until one of them (I.N. Silin) proposed a method for taking them into account [3]. According to [3], any constraint of the form $a_r \le \phi_r(\vec{\theta}) \le b_r$ can be replaced by a simple inequality and equality

$$a_r \le t_r \le b_r, \tag{7}$$

$$\phi_r(\vec{\theta}) = t_r. \tag{8}$$

Here t_r is an additional variable constrained by two boundaries a_r, b_r, (eq. 8) is a constraint in the form of the equation. You can see that constraints (eq. 7) have the same form and structure as (eq. 4), so we can combine them and introduce just one type of simple constraint:

$$\theta_i^{\min} \le \theta_i \le \theta_i^{\max},$$

where index i changes in a wider range: $i = 1, ..., m, ..., m + m_d$ and for $i > m$

$$\theta_i = t_{i-m}, \qquad \theta_i^{\min} = a_{i-m}, \qquad \theta_i^{\max} = b_{i-m}.$$

Then the problem of constrained minimization in a general case can be reformulated as follows: find a minimum of function (eq. 1) under the constraints:

$$\theta_i^{\min} \le \theta_i \le \theta_i^{\max}, \qquad i = 1, ..., m, ..., m + m_d, \tag{9}$$

$$\xi_u(\vec{\theta}) = d_u, \qquad u = 1, ..., m_d, ..., m_d + m_e, \tag{10}$$

where for $1 \le u \le m_d$, $\xi_u = \phi_u$, $d_u = t_u$ and for $m_d < u \le m_d + m_e$, $\xi_u = \psi_{u-m_d}$, $d_u = c_{u-m_d}$.

After such reformulation the number of the parameters to be fitted and of the constraints in the form of simple inequalities of type (eq. 9) becomes $m+m_d$; number of constraints in the form of equations of type (eq. 10) becomes $m_d + m_e$.

When non-simple constraints are only equations they can be taken into account either by the method proposed in [7] or by the penalty function method. Here we use the latter. In such an approach the minimum of the function

$$\Phi = \frac{1}{2}\sum_{j=1}^{n}\left(\frac{f_j(\vec{x}_j,\vec{\theta}) - F_j}{\sigma_j}\right)^2 + \frac{1}{2}T\left(\sum_{r=1}^{m_d}\frac{(\phi_r - \theta_{r+m})^2}{\sigma_r^2} + \sum_{s=1}^{m_e}\frac{(\psi_s - c_s)^2}{\sigma_s^2}\right). \tag{11}$$

Here T is the penalty factor (normally it is sufficiently big number), σ_r, σ_s are formally calculated errors of constraints. In a penalty method function a minimum of eq. 11 is searched for as $T \to \infty$.

3.2 Iteration scheme

Let us rewrite eq. 11 in the form

$$\Phi = \Phi_1 + \frac{1}{2}\sum_{r=1}^{m_d}\frac{(\phi_r - \theta_{r+m})^2}{w_r},$$

where

$$\Phi_1 = \frac{1}{2}\sum_{j=1}^{n}\left(\frac{f_j(\vec{x}_j,\vec{\theta}) - F_j}{\sigma_j}\right)^2 + \frac{1}{2}T\sum_{s=1}^{m_e}\frac{(\psi_s - c_s)^2}{\sigma_s^2}$$

and $w_r = \sigma_r^2/T$. Under a chosen T the minimum condition is

$$\frac{\partial\Phi}{\partial\theta_k} = \frac{\partial\Phi_1}{\partial\theta_k} + \sum_{r=1}^{m_d}\frac{\partial\phi_r}{\partial\theta_k}\cdot\frac{(\phi_r - \theta_{r+m})}{w_r} = 0, \qquad (k = 1, ..., m), \tag{12}$$

$$\frac{\partial\Phi}{\partial\theta_{r+m}} = -\frac{(\phi_r - \theta_{r+m})}{w_r} = 0, \qquad (r = 1, ..., m_d). \tag{13}$$

In both equation 12 and 13 derivatives are taken only for those parameters which are not fixed; i.e. $k \ne i_f$, $r+m \ne i_f$, where i_f is the index of a fixed parameter. The functions on the left side of eq. 12 and eq. 13 depend on $m + m_d$ parameters. Near the minimum we can expand the left sides of the equations in parameter increments retaining only linear terms. For eq. 12 we have

$$\left[\frac{\partial\Phi_1}{\partial\theta_k} + \sum_{r=1}^{m_d}\frac{\partial\phi_r}{\partial\theta_k}\cdot\frac{(\phi_r - \theta_{r+m})}{w_r}\right] + \sum_{l=1}^{m}\left[\frac{\partial^2\Phi_1}{\partial\theta_k\partial\theta_l} + \sum_{r=1}^{m_d}\frac{\partial\phi_r}{\partial\theta_k}\cdot\frac{\partial\phi_r}{\partial\theta_l}\cdot\frac{1}{w_r}\right]\cdot\delta\theta_l - \tag{14}$$

$$\sum_{r=1}^{m_d}\frac{\partial\phi_r}{\partial\theta_k}\cdot\frac{\delta\theta_{r+m}}{w_r} = 0.$$

We wrote eq. 15 in the approximation of the functional argument linearization method [8], in which the derivatives $\partial^2\phi/\partial\theta_k\partial\theta_l$ are discarded. All values of functions and derivatives are taken at the current values of the parameters. Let us also remark

that index l ($l \neq i_f$) in the second term runs over indices of non-fixed parameters. Analogically, for eq. 13:

$$[\phi_r - \theta_{r+m}] + \sum_{l=1}^{m} \frac{\partial \phi_r}{\partial \theta_l} \cdot \delta\theta_l - \delta\theta_{r+m} = 0. \tag{15}$$

From eq. 15 we have for the non-fixed parameter θ_{r+m} ($r = 1 \div m_d$)

$$\delta\theta_{r+m} = [\phi_r - \theta_{r+m}] + \sum_{l=1}^{m} \frac{\partial \phi_r}{\partial \theta_l} \delta\theta_l. \tag{16}$$

Substituting eq. 16 into eq. 15 we will obtain after some algebra:

$$G_k + \sum_{l=1}^{m} Z_{kl} \cdot \delta\theta_l = 0, \tag{17}$$

where

$$G_k = \frac{\partial \Phi_1}{\partial \theta_k} + \sum_{r=1}^{m_d} \left[\frac{\partial \phi_r}{\partial \theta_k} \cdot \frac{(\phi_r - \theta_{r+m})}{w_r} \right], \qquad Z_{kl} = \frac{\partial^2 \Phi_1}{\partial \theta_k \partial \theta_l} + \sum_{r=1}^{m_d} \frac{\partial \phi_r}{\partial \theta_k} \cdot \frac{\partial \phi_r}{\partial \theta_l}.$$

A remarkable feature of the last expression is that the index l runs only over non-fixed parameters $l = 1 \div m$, the index r runs only over those inequalities for which additional parameters t_r (eq. 8) are fixed!

Finally, the solution of eq. 17 is

$$\delta\vec{\theta} = -(Z^{-1} \cdot G).$$

The increments of the additional parameters $\delta t_r = \delta\theta_{r+m}$ are calculated according to formula 16.

The advantage of this iteration scheme is that the matrix inversion only of order $m \times m$ is done irrespective of the number of constraints.

4 Test

Both realizations described above are coded in C++ and tested on the model data for the calibration reaction $pp \to d\pi^+$ under the conditions of the ANKE setup [4]. According to the plans, ANKE will consist of three sub-detectors: a side detector, forward and backward ones. At the moment the side detector is fully assembled, for the forward detector only a scintillation hodoscope is ready. The side detector consists of two scintillation hodoscopes (START, STOP), two proportional chambers with three sensitive planes each. It permits one to reconstruct all the kinematic parameters of the ejectiles passing through the side detector. The scintillation hodoscope in the forward detector is capable of measuring the coordinates of the particle and its time of flight.

The first data were obtained in May and July this year, accuracies are being studied. The main calibration process for the analysis of detector performance is the reaction $pp \to d\pi^+$, so we took this process for the tests. A number of events were simulated for the beam kinetic energy $T_{\text{beam}} = 425$ MeV with the π^+ meson passing through the side detector and the deuterons passing through the scintillation hodoscope of the forward detector. Simulation was done by the GEANT code with all physical processes switched on except the decay of π^+ mesons. In the case where the kinematic parameters of the beam proton and secondary π^+ are known, there is one constraint in the form of an equality, namely the missing mass of the process should be equal to the mass of the deuteron:

$$(E_{\text{beam}} + M_p - E_{\pi^+})^2 - (\vec{p}_{\text{beam}} - \vec{p}_{\pi^+})^2 = M_d^2, \tag{18}$$

where E_{beam}, E_{π^+} are the energies of the beam proton and the secondary π^+ meson, $\vec{p}_{\text{beam}}$, $\vec{p}_{\pi^+}$ are their 3-momenta, M_p, M_d are the masses of the proton and the deuteron respectively.

As was said above for the deuterons which are detected by the forward hodoscope, their coordinates and time of flight (TOF_d) will be measured, as we hope, with the accuracies permitting 4c fit (using all 4 conservation laws). Because at the moment not all accuracies are known, we assumed that their coordinates and times of flight are between some boundaries and put requirements in the form of three inequalities:

$$y_{\min} \leq y_d \leq y_{\max}, \qquad z_{\min} \leq z_d \leq z_{\max}, \qquad t_{\min} \leq TOF_d \leq t_{\max}. \tag{19}$$

The first two requirements come from the geometrical dimensions of the scintillation hodoscope, the last one from the simulation data. Three functions y_d, z_d, t_d were expressed as functions (in the form of polynomials to the third order inclusive) of two angles of the pion in the laboratory system of coordinates.

The total number of fitted parameters was 6, the first three are angles θ_{xz},θ_{yz} of the pion relative to the beam proton and the pion momentum in the laboratory system. The last three parameters were additional parameters t_r, corresponding to three inequalities (eq. 19). Initial pion angles were always 0, initial momenta were calculated as a function of these angles. The coordinates of the pion detected in the side detector were expressed as functions of three pion variables — two angles and momentum. The total number of events was ~ 3000, the maximum number of iterations was 40.

Two fits corresponding to two different realizations, described above, were performed. In the first fit the constraint in the form of non-linear equation (eq. 18) was disabled, in the second it was enabled. In figure 1 the accuracies for the both realizations are shown. Figures 1a,1b,1c are for first fit, figures 1d,1e,1f are for the second one. It is necessary to stress drastic improvement of accuracy in $\Delta p/p$ in the second case, which is the result of additional constraint.

Each event was fitted for three values of the penalty factor T. The initial value was selected by the formula

$$T = 100 \cdot \frac{n_{\text{exp}}}{n_{\text{con}}},$$

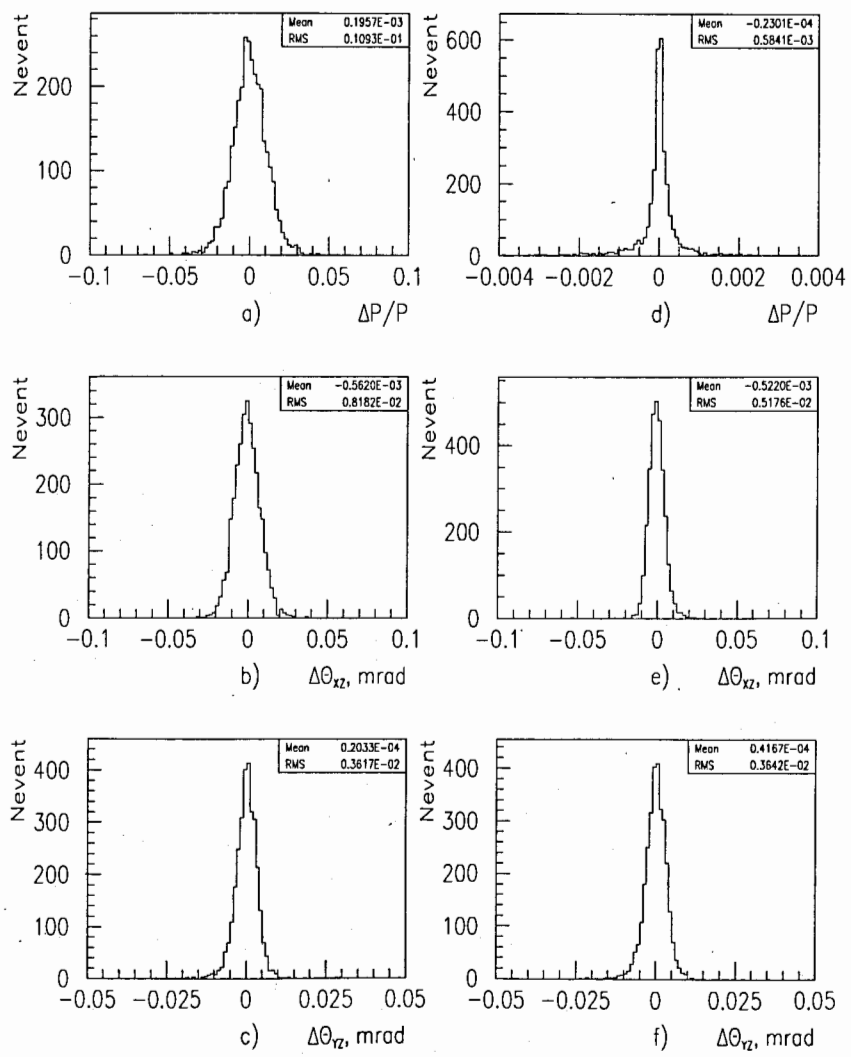

FIGURE 1. Accuracies of the particle kinematic parameter determination.

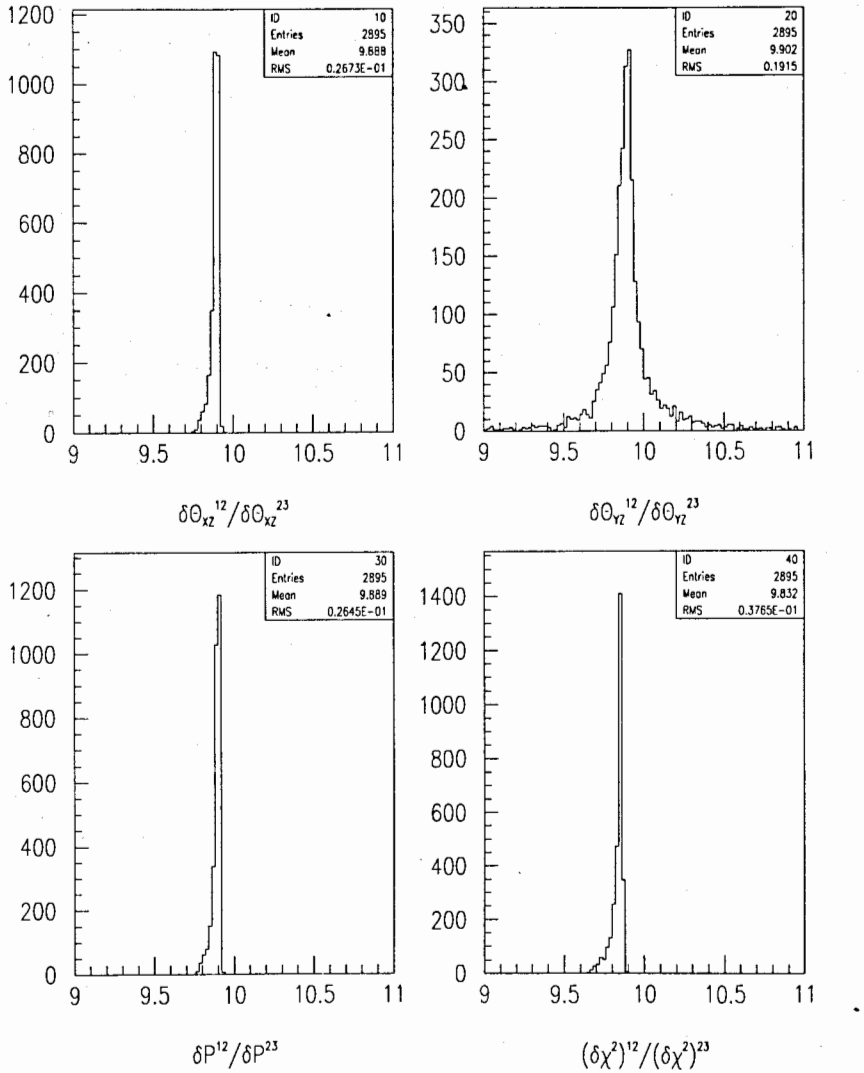

FIGURE 2. Illustration of Richardson approximation.

where n_{exp} is the number of experimental points, n_{con} is the number of constraints. In our case $n_{exp} = 6$, $n_{con} = 4$.

Each successive value of T was ten times larger than the previous one. According to [3], in this case we should have the convergence according to Richardson, i.e. the difference of parameter values in the minimum $\Delta_{32} = p_3 - p_2$ should be 10 times smaller than $\Delta_{21} = p_2 - p_1$. Here p_1 means the value of the fitted parameter for the first value T_1, p_2 for the second T_2 and p_3 for T_3.

In figure 2 the ratios $\Delta\theta_{xz}^{32}/\Delta\theta_{xz}^{21}$, $\Delta\theta_{yz}^{32}/\Delta\theta_{yz}^{21}$, $\Delta p^{32}/\Delta p^{21}$, $\Delta(\chi^2)^{32}/\Delta(\chi^2)^{21}$ are shown. It is seen that they are close to 10, it means that the statements made in [3] are correct.

5 Conclusion

Two codes are developed for minimization of χ^2-like functionals in the C++ language. One of them is realization of the FUMILI code with constraints in the form of simple boundaries. The second one is the minimization with constraints of any type. With FUMILI as a starting point, the C++ code is developed and tested on model data. The results of the test show high performance of the algorithms developed. In conclusion, the authors express their gratitude to the colleagues from the ANKE collaboration for the necessary details.

References

[1] I.N. Silin, *CERN Program Library D* **510**, FUMILI, 1983.

[2] I.N. Silin, Appendix III in the book "Statisticheski metodi v eksperimentalnoi fizike", Atomizdat 1976, translated into Russian from W.T. Eadie et al., "Statistical Methods in Experimental Physics", CERN, Geneva, 1971, North–Holland Publishing Company.

[3] I.N. Silin, "Resolute progress in a constrained minimization problem", *JINR Rapid communications*, Nr 3 [89]-98 pp. 25–30.

[4] Forschungszentrum Jülich, Germany, Annual Report 1996. (All the details about ANKE setup and related bibliography can be found in this report).

[5] J.P. Berge, F.T. Solmitz and H.D. Taft, *Rev. Sei. Instr.* **32** (1961) 538.

[6] R. Bock, CERN 60–30 (1960).

[7] V.S. Kurbatov, I.N. Silin, *Nucl. Instr. and Meth. A* **345** (1994) 346-350.

[8] S.N. Sokolov and I.N. Silin, *Preprint JINR* D-810, Dubna (1961).

Received by Publishing Department
on November 12, 1998.

Дымов С.Н. и др. E10-98-318

Минимизация с ограничениями в среде *C* ++

Основываясь на идеях, предложенных одним из авторов (Силин И.Н.), разработано программное обеспечение для фитирования данных с ограничениями. Ограничения могут быть произвольного типа (равенствами и неравенствами). Был использован простейший из возможных подходов. Широко известная программа FUMILI реализована на языке *C* ++. Ограничения в форме неравенств $\varphi(\vec{\theta}) \geq a$ заменялись равенствами вида $\varphi(\vec{\theta}) = t$ и простыми неравенствами типа $t \geq a$. При рассмотрении равенств применялся метод квадратичных штрафных функций. Программное обеспечение тестировалось на модельных данных установки ANKE (COSY, Forschungszentrum Jülich, Germany).

Работа выполнена в Лаборатории ядерных проблем и Лаборатории вычислительной техники и автоматизации ОИЯИ.

Препринт Объединенного института ядерных исследований. Дубна, 1998

Dymov S.N. et al. E10-98-318

Constrained Minimization in *C* ++ Environment

Based on the ideas, proposed by one of the authors (I.N.Silin), the suitable software was developed for constrained data fitting. Constraints may be of the arbitrary type: equalities and inequalities. The simplest of possible ways was used. Widely known program FUMILI was realized to the *C* ++ language. Constraints in the form of inequalities $\varphi(\vec{\theta}) \geq a$ were taken into account by change into equalities $\varphi(\vec{\theta}) = t$ and simple inequalities of type $t \geq a$. The equalities were taken into account by means of quadratic penalty functions. The suitable software was tested on the model data of the ANKE setup (COSY accelerator, Forschungszentrum Jülich, Germany).

The investigation has been performed at the Laboratory of Nuclear Problems and at the Laboratory of Computing Techniques and Automation, JINR.

Preprint of the Joint Institute for Nuclear Research. Dubna, 1998