Kinematic Analysis of Bubble Chamber Events: FIT

An IBM 709 Programme imposing Momentum and Energy Conservation on Measurements of complete Events

by

R. Böck
Propriété littéraire et scientifique réservée pour tous les pays du monde. Ce document ne peut être reproduit ou traduit en tout ou en partie sans l’autorisation écrite du Directeur général du CERN, titulaire du droit d’auteur. Dans les cas appropriés, et s’il s’agit d’utiliser le document à des fins non commerciales, cette autorisation sera volontiers accordée.

Le CERN ne revendique pas la propriété des inventions brevetables et dessins ou modèles susceptibles de dépôt qui pourraient être décrits dans le présent document; ceux-ci peuvent être librement utilisés par les instituts de recherche, les industriels et autres intéressés. Cependant, le CERN se réserve le droit de s’opposer à toute revendication qu’un usager pourrait faire de la propriété scientifique ou industrielle de toute invention et tout dessin ou modèle décrits dans le présent document.

© Copyright CERN, Genève, 1961

Literary and scientific copyrights reserved in all countries of the world. This report, or any part of it, may not be reprinted or translated without written permission of the copyright holder, the Director-General of CERN. However, permission will be freely granted for appropriate non-commercial use. If any patentable invention or registrable design is described in the report, CERN makes no claim to property rights in it but offers it for the free use of research institutions, manufacturers and others. CERN, however, may oppose any attempt by a user to claim any proprietary or patent rights in such inventions or designs as may be described in the present document.
Kinematic Analysis of Bubble Chamber Events: FIT
An IBM 709 Programme imposing Momentum and Energy Conservation on Measurements of complete Events

by

R. Bück

GENEVE
Table of Contents

I. Introduction
   1. Survey 2
   2. History and Acknowledgements 4

II. Mathematics in FIT
   1. Definition of Track parameters 5
   2. Apex Constraints (Momentum and Energy Conservation) 7
   3. Connecting Tracks 8
   4. Fitting Procedure 12

III. Programme Details 16

IV. Experience with the testing version 24

V. Improvements and extensions 25

References 29

Appendices 30
Kinematic Analysis of Bubble Chamber Events; FIT
An IBM 709 Programme imposing Momentum and Energy
Conservation on Measurements of Complete Events

R. Böck

Abstract

This report describes a programme which adjusts track measurements of a bubble chamber event such that they obey the principles of momentum and energy conservation at all interaction points in the event. The measured parameters for a track are 1/p, tan λ and φ (P = momentum, λ = dipping angle, φ = azimuth angle). Not all of these parameters, however, need to be measured for all tracks but can be introduced as unknowns. The topology of the event and its interpretation given in terms of masses for inbound, outbound and target particles is required as starting information.

The "fit" is carried out according to the principles of the least squares method. The formulae and the organization of the programme are described and some results from test calculations are given. Possible extensions of the programme are suggested.
I. Introduction

1. Survey

In the analysis of a track chamber experiment certain quantities of physical interest are calculated from the measurements made on the photograph of the events. Although the fine physical quantities are derived by a statistical analysis of the calculation results of the total number of events, the analysis of a single event thus forms a basic operation in the total analysis.

Due to the large amount of calculation involved in the single-event analysis many attempts have been made and described to solve this problem in a general way using modern computing equipment, but only recently the problems of analyzing the total information for an experiment are tackled with the same emphasis (e.g. the prospective FAIR programme of the Berkeley Propane Chamber group). The present paper represents another contribution to the computations for single events.

Single-event programmes usually are subdivided into sections being referred to as Geometry and Kinematics respectively. The first of them is meant to evaluate numerically the position for points, and positions, directions, curvatures and eventually the changes in curvature (first to third derivatives of coordinates) for tracks. It is also expected to give estimates of the uncertainties of these quantities due to measurement error. In the second part of the single event analysis these geometric quantities are taken as starting values to find, if possible, a unique interpretation of the event. If such an interpretation is found the geometrical values are corrected according to the method of least squares in order to make them obey the constraint of momentum-vector and energy conservation.
In practice the information coming from the photograph is generally not sufficient to distinguish between the correct hypothesis and all wrong assignments at a very early stage, so the fitting programme in the kinematical analysis of an event serves a double purpose:

a) to detect wrong interpretations
b) for a correct interpretation to impose the conservation constraints.

A wrong interpretation is detected by treating it as correct imposing the constraints mentioned, until some obvious contradiction to physical laws is encountered or until some other evidence exists that the event is behaving other than a good one, i.e. the iterative process does not converge within a certain number of steps or the corrections to measurements become highly improbable judging from the average uncertainty of the measurements and the rules of statistical fluctuations. Although experience shows that the selective property of the fitting programme is useful in many cases it is evidently not possible in general to decide about the absolute acceptability of an interpretation at the fitting stage. It is up to a higher level programme to select eventually between various hypothesis all giving a "good fit". Criteria for this final selection can be the fit results $\chi^2 = \text{minimum function (see equ. (11))}$ or the number of steps necessary to obtain the convergence. Eventually some additional information like track ionization or results from connected events can permit the preference of one solution to others.
2. History and Acknowledgements

The double purpose as outlined above and the ensuing importance of the fitting programme for the kinematical analysis has been recognized early and used for all large scale programmes. Most of these use versions of GUTS or APEX FIT, these two being the fitting programmes written for the Berkeley Hydrogen Chamber Group \[4\]* and the later (generalized) version written at CERN \[\] Both perform a fit for tracks originating at one apex only.

Although GUTS already has been modified to cope with a special class of two-vertex events it was felt that there was some need for a general multi-vertex fitting programme. During my stay with the Alvarez Group in Berkeley (summer 1960) many details of such a procedure were discussed and it was decided to write a tentative version of this programme suitable for both the Berkeley Hydrogen chamber analysis programmes and the CERN system.

This version was called FIT and successfully completed in the meantime. It was tested with a number of one-, two- and three-vertex events with encouraging results. The present report gives an outline of the formulae and methods used. The logical flow of the programme is described, but its detailed specification are not contained in this report.

For contributions to the ideas of this programme I should like to express my thanks to many people in the UCRL Berkeley Hydrogen Chamber Group and also in the Propane Chamber Group of the same laboratory, where a programme similar to FIT has been written meanwhile. I quote particularly A.H. Rosenfeld, F.T. So and H. White. Encouragements and suggestions obtained from colleagues of the IEP group at CERN are also warmly acknowledged.

* see references page 29
II. Mathematics in FIT

1. Definition of track parameters

Each track in the chamber is taken to be defined by three parameters, these being:

\[ x_{1i} = \frac{1}{p_i}, \]
\[ x_{2i} = \tan \lambda_i, \]
\[ x_{3i} = \varphi_i \]

(1)

where \( p_i \) = momentum, \( \lambda_i \) = dipping angle and \( \varphi_i \) = azimuthal angle for track \( i \). The definition of \( \lambda_i \) and \( \varphi_i \) is clear given the direction cosines for track \( i \) to be \( l_{x_i} = \cos \lambda_i \cos \varphi_i; \)
\( l_{y_i} = \cos \lambda_i \sin \varphi_i; \)
\( l_{z_i} = \sin \lambda_i. \)

In addition we define the sense of the track such that the direction cosines point away from the apex. In general for a visible (charged) track these parameters are obtained from measurements of film co-ordinates by the Geometry programme (the curvature of the track being proportional to \( 1/p \cos \lambda \)) and enter the Kinematics programme as measurements together with an estimate of their uncertainty. At this point the uncertainty includes errors made in measuring the track, errors due to the irregular behaviour of particles in the chambers (e.g. multiple Coulomb scattering) and eventually errors arising from the reconstruction method and transformation of reconstruction results (e.g. uncertainty of the index of refraction in the chamber, uncertainty of the magnetic field).

Additional information can be obtained, if a particle comes to rest in the chamber. From the range and the known mass assignment \( x_{1i} \) can be derived for such a track. Its error will come from the uncertainty of the track length and the straggling.
of the range-momentum relation. If a track is not visible but known to be straight and to pass through two measurable points, \( x_{2i} \) and \( x_{3i} \) can be evaluated for this track (e.g. production and decay point for a neutral particle). The errors in this case depend on the accuracy of the reconstructed points in the chamber. All three parameters \( x_{\nu i} \) finally are known to a certain accuracy if the track \( i \) belongs to the beam.

On the other hand generally some of these parameters can be evaluated only if the kinematical constraints are introduced. This is always the case for \( x_{1i} \) of a neutral track and also for its \( x_{2i} \) and \( x_{3i} \) if no visible interaction takes place. It is also possible that a track is visible but too short to allow for a measurement of curvature (missing \( x_{1i} \)). These missing quantities are called unmeasured variables and have to be treated differently from the measured ones.

A third parameter definition has been introduced by F.T. Solmitz [3] in splitting up the measured variables into well measured and poorly measured quantities. The reason for this is a purely numerical one. If namely an error matrix with very large elements together with error matrices of normal magnitude are used to construct a second matrix which later is inverted, it may cause this second matrix to become nearly singular and so create some numerical difficulties. These variables therefore are treated in a way different from well measured ones although mathematically the two, of course, are still equivalent. For the details of the procedure see section II.4.
2. Apex Constraints (Momentum and Energy Conservation)

a) Constraint Equations

An event in the terminology of this paper consists of a number of apices each representing a certain type of reaction, e.g. an interaction, a scatter or a decay. Each reaction is mathematically described by four equations, representing the conservation of the momentum vector and of the total energy.

The form in which these equations are introduced as constraints does not influence the result. As a form sufficiently general and obvious FIT uses the following relations:

\[
\begin{align*}
  f_1 &= \sum_i P_{x_i} = \sum_i P_i \cos \lambda_i \cos \varphi_i &= 0 \\
  f_2 &= \sum_i P_{y_i} = \sum_i P_i \cos \lambda \sin \varphi_i &= 0 \\
  f_3 &= \sum_i P_{z_i} = \sum_i P_i \sin \lambda &= 0 \\
  f_4 &= -M_T^T + \sum_i x_{\nu_i} U_i = -M_T^T + \sum_i \sqrt{x_{\nu_i}^2 + M_i^2} &= 0
\end{align*}
\]  

(2)

The parameters \( \lambda_i \) and \( \varphi_i \) have been introduced according to the definition above, i.e. always pointing away from the apex. The sign distinction between incoming and outgoing tracks therefore is necessary for the energy equation \( f_4 \) only. There by pure definition the incident energy (and \( M_T^T \)), are taken with the negative sign. In these equations all \( x_{\nu_i} \) are assumed to be given at the apex, where the constraints are formed. \( M_T^T \) is the target mass and will be zero for a decay.

Parameters entering the equations (2) are \( x_{\nu_i} (\nu = 1, \ldots, 3) \) and \( M_i \) for each track \( i \) and \( M_T^T \). It is recalled that the problem of FIT is to vary the \( x_{\nu_i} \) such that equations (2) are satisfied and in addition the minimum for the least squares function \( \chi^2 \).
is obtained ($\chi^2$ is defined in equ. (11) below). The masses $M_i$ and $M_j^r$ are taken to be constants and will not be considered as variables.

b) Matrix of derivatives

Given equations (2) it is easy to write down the matrix of derivatives $df/dx_i$ which is defined by $(df/dx_i)_{jk} = \frac{\partial f}{\partial x_{ki}}$ ($j = 1..4$; $k = 1..3$). This matrix plays an important role in the iterative procedure of FIT (Section II.4). For a track $i$ it is:

$$
\frac{df}{dx_i} = \begin{pmatrix}
- p_i^2 \cos \lambda_i \cos \varphi_i & - p_i \cos^3 \lambda_i \tan \lambda_i \cos \varphi_i & - p_i \cos \lambda_i \sin \varphi_i \\
- p_i^2 \cos \lambda_i \sin \varphi_i & - p_i \cos^3 \lambda_i \tan \lambda_i \sin \varphi_i & p_i^2 \cos \lambda_i \cos \varphi_i \\
- p_i^2 \sin \lambda_i & - p_i \cos^3 \lambda_i & 0 \\
\mp p_i^3 / v_i & 0 & 0
\end{pmatrix}
$$

(3)

The sign of the term $\frac{\partial f}{\partial x_{1i}}$ is positive for the incoming, negative for all outgoing tracks corresponding to the definition of $f_4$ in (2).

3. Connecting Tracks

a) Transformation of Variables along a Track

In equation (2) the assumption was made that all variables $x_{\nu_i}$ are given at the apex. This assumption however does not hold even if we assume that all variables and, if they are measured, their errors have been transformed to the apex. For a charged track connecting two apices both entering FIT the variables can be
defined at one end only, if they are assumed to be measured uniquely. The latter can be avoided by defining the track from both ends and adding constraint equations between the two sets of variables. Clearly this increases both the number of constraints and variables and thus the amount of calculation by a considerable factor. It is less logical, too, if the connecting track is measured only once. For FIT the unique definition of variables for connecting tracks was preferred. To be quite general a specification length \( L^s \) was introduced for all tracks giving the distance along the track from the apex to the point at which the variables are specified. For a connecting track \( L^s \) takes two different values according to which of the two adjacent apices is regarded. For this reason connecting tracks are counted twice for the total number of tracks and wherever quantities are kept in vectors or matrices where one index runs through the number of tracks, (quantities kept "trackwise"), they appear twice for connecting tracks.

There is no contradiction to the unique definition of the variables of a connecting track as the variables at the lower-range end of a connecting track are introduced as "dependent" variables, i.e. they are carried through for logical reasons, but will be used only as suitable functions of the "independent" variables at the higher-range end of the track.

The specification length \( L^s \) is set before entering FIT and may be zero for all tracks meaning that \( x_{vi} \) (variables at apex) and \( x_{v_i}^s \) (variables at specification point) are identical. For the lower-range end of a connecting track (i.e. for the dependent variables) \( L^s \) is set by FIT and is given by \( L^s = -|L - L_i^s| \) if \( i = \) index for the independent end of the track. For the sign convention of \( L \) and \( L^s \) see next paragraph.
To express $x_i$ in terms of $x_s$ FIT assumes that for a charged track:

a) the momentum changes along the track according to a tabulated relation between the quantities range and momentum mass.

b) the dip is constant.

c) the change in the azimuthal angle along the track is directly proportional to the length along the track, i.e. the projection of the track in the x-y-plane is a circle.

The equations therefore are:

$$
\begin{align*}
    x_{1i} &= \frac{1}{\frac{1}{p_i}} = \frac{1}{\text{PF} \left( \text{RF} \left( \frac{1}{x_{1i}} \cdot M_i \right) + L_i \right)} \\
    x_{2i} &= x_{2i}^s \\
    x_{3i} &= x_{3i}^s + \frac{L_i}{\epsilon_i} \cos \lambda_i
\end{align*}
$$

(4)

For neutral tracks the variables are assumed to be constant at all points of the track.

In the first equation above I have used PF (P-function) and RF (range-function) as conversion functions between range and momentum. $\frac{1}{\epsilon_i}$ is the projected curvature of the track. Evidently $\frac{1}{\epsilon_i}$ is a signed value with the sign depending on the charge of the track and the magnetic field definition only. The direction of the transformation is given by the sign of $L_i^s$. $L_i^s$ is negative, if $i$ refers to an incident track, i.e. if from the specification point to the apex one moves in particle direction (in UCRL 9099-terminology: if the equations (4) specify swimming downstream).
b) Derivatives

As has been described above, the variables entering FIT are not generally given at the apex but at some specification point. It is clear that one wants to refer in all equations to these variables \( x_{\nu_i}^s \). It has been shown how the constraint equations (2) can be formed transforming first the \( x_{\nu_i}^s \) into \( x_{\nu_i} \) using equations (4) and entering these into (2). The same procedure is possible for finding the derivation matrix \( \frac{df}{dx_i^s} \) (definition as for equ. (3)). Using matrix notation one obtains

\[
\frac{df}{dx_i^s} = \frac{df}{dx_i} \frac{dx_i^s}{dx_i} \tag{5}
\]

which in ordinary notation just expresses

\[
\frac{\partial f_i}{\partial x_{ki}^s} = \sum_{\nu} \frac{\partial f_i}{\partial x_{\nu_i}} \frac{\partial x_{\nu_i}}{\partial x_{ki}^s} \tag{5a}
\]

The matrix \( \frac{df}{dx_i^s} \) has been written down previously (equation (3)). For the second term in (5) we obtain from (4) for a track \( i \):

\[
\frac{dx_i}{dx_i^s} = \begin{pmatrix}
\frac{\Delta x_{\nu_i}^s}{\Delta x_{\nu_i}^s} & 0 & 0 \\
0 & 1 & 0 \\
0 & -\frac{L_i^s}{\epsilon_i} \sin \lambda_i \cos^2 \lambda_i & 1
\end{pmatrix} \tag{6}
\]

where \( \frac{\Delta x_{\nu_i}^s}{\Delta x_{\nu_i}^s} \) is obtained by numerical differentiation.
It has somewhat arbitrarily been assumed in (6), that $L^g_i$ and $\eta_i$ are constant. The term $\frac{\partial x_{3j_i}^g}{\partial x_{2j_i}^g}$, however, does not contribute very much and it seems more or less a matter of taste if instead of the procedure in FIT $x_{3j_i}^g - x_{3j_i}^g$ is kept constant or a still more complicated term is introduced. See remarks in section V.5.

4. **Fitting procedure**

In the previous paragraphs the variables and equations defining an event have been given. We now have to describe the method of least squares as applied to our problem. Apart from the introduction of the badly measured variables $[3]$ the notation is the same as in $[5]$ section V which itself is just an extract from standard statistics handbooks. Matrix notation is used throughout this paragraph.

a) **Notation**

- $m =$ the vector of "well measured variables" consisting of $x_{\nu_i}^g$ from all tracks in question as far as they are well measured (see section II.1);
- $c =$ the vector of corrections for $m$ such that $m + c$ are the fitted quantities;
- $\bar{c} =$ the approximate vector of corrections as found in a previous iteration. Initially set at zero;
- $G^{-1} =$ the error matrix for $m$. FIT assumes that no correlations for variables of different tracks be given and $G^{-1}$ therefore is made up of square submatrices along the diagonal with a maximum size of $3 \times 3$. All other elements are supposed to be zero;
- $m^* =$ the vector of the variables $x_{\nu_i}^g$ not contained in $m$ (i.e. poorly or unmeasured variables);
c* = the vector of corrections for m* such that m* + c* are the fitted quantities;

\( \vec{c} * \) = the approximate vector of corrections as found in a previous iteration. Initially set at zero;

G* = the inverse of the error matrix (weighting matrix) of m*. This matrix contains zero elements throughout a line or column referring to an unmeasured variable. Furthermore FIT assumes that all non-diagonal elements be zero (i.e. correlations between poorly measured variables are ignored);

f = the vector of constraint functions (Equations (6) for each apex);

B = the derivation matrix \( \frac{df}{d(m+c)} \);

B* = the derivation matrix \( \frac{df}{d(m+c*)} \);

Note: B and B* are made up from submatrices as described by equs. (5), (6) and (3).

a = a vector of Lagrangeian multipliers introduced as a conventional means to solve the least squares problem;

r = defined in equation (12a) as "residuals" to simplify the expressions;

\( \chi^2 \) = minimum function defined in equ. (11).

Note: the superscript T stands for matrix transposition.

b) Method of Finding Fitted Quantities

The two requirements for the fitted variables m + c and m* + c* are that the constraints must be satisfied

\[ f = f(m + c, m* + c*) = 0 \] (10)

and that from the many possible sets of c, c* the one is chosen which makes the function

\[ \chi^2 = c^T G c + c^* T G* c* \] (11)

to a minimum.
With the definition of $G^*$ above the elements of $c^*$ referring to unmeasured variables do not enter (11). It is not possible to solve this problem in one step, as (10) is a set of non-linear functions in $m + c$, $m^* + c^*$ given by (6). One can, however, assume linearity and expand (10) into a Taylor series giving

$$f = f(m + \bar{c}, m^* + \bar{c}^*) + B(c - \bar{c}) + B^*(c^* - \bar{c}^*) = 0$$

where $B = \frac{df}{d(m+c)}$ and $B^* = \frac{df}{d(m^*+c^*)}$ which can be written

$$f = Bc + B^*c^* + r = 0$$

with

$$r = f(m + \bar{c}, m^* + \bar{c}^*) - B\bar{c} - B^*\bar{c}^*$$

To solve for $c, c^*$ one introduces $\alpha$ rewriting (11)

$$\chi^2 = c^T G c + c^*^T G^* c^* + 2\alpha^T f$$

The minimum condition leads us to

$$\left(\frac{d\chi^2}{dc}\right)_T = 2(Gc + B^T\alpha) = \text{zero vector}$$

and

$$\left(\frac{d\chi^2}{dc^*}\right)_T = 2(G^*c^* + B^T\alpha) = \text{zero vector}$$

From (14) we get

$$c = -G^{-1}B^T\alpha$$

which is introduced in (12) to give

$$\alpha = G_B(B^*c^* + r)$$

with $G_B = (B G^{-1} B^T)^{-1}$. If we use this to eliminate $\alpha$ in (14a) we obtain

$$c^* = -(G^* + B^* G_B B^*)^{-1} B^T G_B r = -K^{-1} B^T G_B r$$

(17)
This is the result for $c^*$ and, together with (16) and (15), gives all quantities desired. As the linearity assumption (12) is only an approximation, (10) will not generally be satisfied if (12) is and the procedure has to start again using $c$ and $c^*$ as improved values $\bar{c}$ and $\bar{c}^*$.

c) Estimation of errors for fitted quantities

In equations (15), (16) and (17) $c$ and $c^*$ are written down as linear functions of $r$ and, together with the definition of $r$, as general functions of $m$ and $m^*$. It is common to linearize these functions again to obtain $m + c$ and $m^* + c^*$ as linear functions of $m$ and $m^*$ (see also [2], [4]). Once this principle is agreed on the error matrix for $m + c$ and $m^* + c^*$ is obtained by (see appendix 1):

\[
G_{m+c}^{-1} = \frac{d(m+c)}{dm} G^{-1} \left( \frac{d(m+c)}{dm} \right)^T + \frac{d(m+c)}{dm} G_{m+c}^{-1} \left( \frac{d(m+c)}{dm} \right)^T \quad (18a)
\]

\[
G_{m^*+c^*}^{-1} = \frac{d(m^*+c^*)}{dm} G^{-1} \left( \frac{d(m^*+c^*)}{dm} \right)^T + \frac{d(m^*+c^*)}{dm} G_{m^*+c^*}^{-1} \left( \frac{d(m^*+c^*)}{dm} \right)^T \quad (18b)
\]

In these equations $G^{-1}$ and $m^*$ are, of course, only defined for the poorly measured quantities. On the other hand $G_{m^*+c^*}^{-1}$ refers to the complete vector $m^* + c^*$. Additionally $\approx_{0}$ (18a) and (18b) one may want the correlations between $c + m$ and $c^* + m^*$ given by

\[
G = \frac{d(m+c)}{dm} G^{-1} \left( \frac{d(m+c)}{dm} \right)^T + \frac{d(m+c)}{dm} G_{m+c}^{-1} \left( \frac{d(m+c)}{dm} \right)^T \quad (18c)
\]

The derivatives and the matrix arithmetic is given in appendix 2. The resulting matrices are:

\[
G_{m+c}^{-1} = G^{-1} - G^{-1}_B T G B G^{-1} + G^{-1}_B T G B K^{-1}_B T G B G^{-1} \quad (19a)
\]

\[
G_{m^*+c^*}^{-1} = K^{-1} \quad (19b)
\]

\[
C = G^{-1}_B T G B K^{-1}
\]
III. Programme Details

1. General

The formulae to be programmed in FIT have been described in the previous section. This section will explain in some detail how they were programmed and by which means space (and, to some extent, time) requirements of the programme could be reduced. Necessarily this section will be closely related to the flowchart (appendix 3), which splits the problem up into different boxes representing FORTRAN subroutines. All subroutines are steered by FIT, which also carries out the tests indicated in the flowchart by diamonds at the branching points. The following description of subroutines will be kept as detailed as necessary and in terms general enough to avoid too close a relation to the compiling system used (FORTRAN). The notation is the same as in section II.

2. PARAM

The first routine is meant to furnish the necessary programme parameters. The routine sets the following criteria and tolerances in accordance with DATEST and ENTEST:

a) maximum number of iterations permitted
b) maximum number of cutsteps* (consecutive)
c) maximum number of cutsteps* (total)
d) maximum number of steps with \( \Sigma |f| \) non-convergence

e) maximum number of steps with \( \Sigma |\Delta x| \) non-convergence

f) maximum \( |f| \) permitted at final step

g) maximum \( |\Delta x| \) permitted at final step

h) maximum \( \chi^2 \) permitted at any step

* see DATEST (III.6), ** see ENTEST (III.9)
In addition initial conditions are set up, i.e. all counters are set at zero. All parameters and counters are kept in an indexed parameter bank.

3. SETUP

Like PARAM this routine serves to prepare the iterative part of FIT. It is supposed to provide the following routines with the starting information in a proper form. In all lists and arrays made up trackwise connecting tracks appear twice*. By means of a connection marker the distinction between independent and dependent variables and reference to the corresponding independent variables can be made.

The starting values coming from SETUP are those listed below:

- $m_i$: array of well measured variables
- $\bar{c}_i$: array of corrections to $m$, starting values zero
- $m^*_i$: array of badly or unmeasured variables
- $\bar{c}^*_i$: array of corrections to $m^*$, starting values zero
- $x_{si}$: complete array of variables $m+\bar{c}, m^*+\bar{c}^* (\nu = 1..3, i = \text{track nr.})$
- $y_{ii}$: complete array of corrections ($\bar{c}, \bar{c}^*$), starting values zero ($\nu = 1..3, i = \text{track nr.}$)
- $G_{ii}^{-1}$: square (maximum $3 \times 3$) error matrices for the well measured quantities of each track ($i = \text{track nr.}$)
- $G^*$: array of either weights $\frac{1}{\sigma^2}$ (badly measured variables) or 0 (unmeasured variables)
- $M_i$: array of masses ($i = \text{track nr.}$)
- $L_{si}$: array of specification lengths ($i = \text{track nr.}$)
- $L_i$: array of total lengths ("
- $\frac{1}{e_i}$: array of projected curvatures ($i = \text{track nr.}$)
- $M^\tau_j$: array of target masses ($j = \text{apex nr.}$)

* See remarks in section II.3a, page 9
In addition lists of cross references between tracks and points and between tracks and variables in arrays like \( m \) and \( m^\ast \) the index of which does not run through track numbers are made up. One of these is the array of variable description codes indicating for each variable its status (well, badly or unmeasured). As SETUP is highly dependent on conventions used in the programmes outside FIT it seems pointless to go into more details of how these lists are obtained. The testing version assumes no other information about the topology of the event than point and track labels. Obvious mistakes in the data entering lists and arrays lead to an immediate error return from FIT to the higher level programme.

An error return is reached also if the space set aside for arrays and matrices is found to be insufficient. The testing version allows for three apices with a total of 12 tracks. 10 variables may be badly or unmeasured.

4. CONNY

As the \( x_{\nu_i}^s \) for connecting tracks \( i \) are defined uniquely but appear twice in the array, the two entries have to be made equal. Because of the different definition from the two ends of a connecting track the angles are inverted in this process:

\[
\lambda_{\text{inv}} = -\lambda; \quad \phi_{\text{inv}} = \phi + \pi.
\]

This holds for the \( x_{\nu_i}^s \)-vector serving for the calculation of constraints and derivatives. In the \( \phi_{\nu_i}^s \)-array no inversion is done.
5. GOTOX

As a preparation for the calculation of constraints and derivatives the array of \( x_{v1} \) variables transformed to the apex (\( v = 1..3, i = \) track nr.) is formed. Together with this transformation (equ. (4)) for charged tracks and for \( L_i^S \neq 0 \) the derivation matrix (6) is calculated. For this purpose \( 3 \times 3 \) locations are reserved for each track. The numerical differentiation for \( \frac{\partial x_{li}}{\partial x_{pi}} \) is done with a constant difference in range \( (\Delta R_i = 0.05 R_i) \) at apex and specification point. The quotient in the resulting differences of \( x_{li} \) and \( x_{pi} \) is the term wanted. An array of \( R_i = \) range for track i at the apex is obtained at the same time.

6. DATEST

This routine has been inserted to stop the iteration process as soon as the quantities at hand turn out to be physically meaningless and cannot be saved. A wrong interpretation of the event can thus be detected at an early stage. At present only two tests are applied.

a) The momentum must be positive for any track.

b) The range of any particle at the apex must be longer than the length of the track.

The second test is done, of course, for charged tracks only, and implies the first test. In case this test fails an estimated error both for the length and the range has to be considered before the failure is reported.

If an error is detected before any iteration has taken place, the data set-up is incorrect and generally no means of saving the hypothesis exists. If a failure occurs later the
cutstep procedure as in GUTS (see [4] and [6]) is used once or several times to take a mean value between the (wrong) values of the last step and the physically meaningful values of the previous iteration. This procedure is applied to $\xi^S_{\nu_i}$ and $x^S_{\nu_i}$. The previous values are always kept in separate arrays. GOTOX has to be called again after such a procedure before the calculation can continue. Both the total number and the consecutive number of cutsteps are counted and compared against the corresponding parameters. If any of them exceeds the maximum the process is stopped and FIT reaches an error exit.

7. MAXIES

MAXIES is the first routine of two to carry out one iteration. Using the $x_{\nu_i}$ array and eventually $\frac{dx_{\nu_i}}{dx^S_{\nu_i}}$ the $f$-array and a full matrix $\frac{df}{dx^S_{\nu_i}}$ of dimension $4 \times 3$ for each track are obtained using equations (2), (3) and (for charged tracks with $L^S \neq 0$ only) (5). In case of a connecting track the different definition of the angles at the two ends must be considered by changing the sign of the second column (i.e. of the terms $\frac{\partial f_{\nu_i}}{\partial x^S_{\nu_i}}$ (j = 1...4)) at the dependent side of the track.

Next the matrices and vectors used in the fitting procedure are prepared. The array $r$ (equation (13)) is obtained using $f$, $\frac{df}{dx^S_{\nu_i}}$ and $\xi^S_{\nu_i}$ which avoids the splitting up into measured and badly or unmeasured variables at this point. Further $B$, $B^*$, $B^{-1}$ and $B^{-1}B^T$ are obtained using the array of variable description codes. For $B$ and $B^{-1}$ only non-zero elements* are kept, i.e. four elements for each well measured variable. They would form part of a column in the full matrix whose dimension would be $(4 \times \text{number of apices}) \times (\text{number of well measured variables})$. The elements stored this way for dependent variables would, in the full matrix, appear in the same

* non-zero elements means elements as in equ. (3).
columns as the ones calculated for the corresponding independent variables, i.e. a column referring to a variable of a connecting track has eight non-zero elements. B* however is produced as a full matrix including the zeros because the storage requirements are not excessive and a different way of storing would complicate considerably its later use. The same applies, of course, to $BG^{-1}B^T$. This matrix has $(4 \times \text{number of splices})$ rows and columns and is made up of $4 \times 4$ submatrices. These submatrices which I can index $jk$ are non-zero (a) if $j = k$ and (b) if there is a connection between apex $j$ and apex $k$.

As a last step MAXIES evaluates $G_B^*B^*$ and $K = G^* + B^T G_B^* B^*$. 

8. NEWVAL

This subroutine is just a continuation of MAXIES and the two form a logical unit as was already mentioned above. By simple matrix arithmetic using machine language subprogrammes one obtains with equa. (17), (16) and (15) in this order the new parameters $c^*$ and $c$. In evaluating equation (15) the form in which $G^{-1}B^T$ is given must carefully be considered: for non-connecting tracks one obtains $c_m = (G^{-1}B^T)_m \alpha_j$ where $m$ runs through some values corresponding to the variables at hand and $\alpha_j$ is a subvector of four elements pertaining to the $j$-th apex which is the apex the variables $c_m$ are related to. For a connecting track we get the formula $c_m = (G^{-1}B^T)_m \alpha_j + (G^{-1}B^T)_n \alpha_k$ where the indices $n$ and $k$ refer to variables and the apex at the other end of the track. For connecting tracks the elements of $c$ are evaluated at one end only. After $x_{yi}^s$ and $c_{yi}^s$ have been preserved they are replaced with the new values combining $m + c$ with $m^* + c^*$ and $c$ with $c^*$ according to the variable description code. $\chi^2$ is obtained from

$$\chi^2 = -a^T r$$  \hspace{1cm} (20)
which is equivalent to equ. (11) but can be calculated much easier. (To prove the identity use equ. (12), (14) and (14a)).

9. ENTEST

Having found new values for all parameters this subroutine tests the acceptability of this solution. In principle I decided to rely on two simple tests rather than on the expressions given in [4] which certainly have a better statistical foundation but take a considerable amount of time and space to be evaluated and, considering all errors from rounding, do not give much more confidence in the results obtained. The two tests in the existing version of FIT are:

a) The \(|f|\)-test. The magnitude of each element out of the f-vector is tested against the tolerance set in PARAM (see III.2). If each element is less than or equal to this parameter the test is successful. This procedure is permitted as all constraints are evaluated in the same dimension (GeV).

b) The \(|\Delta x|\)-test. The magnitude of each element of \(c^s\) is compared with the corresponding element of the previous step. For \(1/P\) the difference is divided by \(1/P\) to obtain a relative value, for the angles the absolute differences are formed. If the sum of the magnitudes of such differences for each track is below the acceptance parameter, the test is successful. The total sum of \(|f|\)-s or \(|\Delta x|\)-s respectively is formed over all existing elements or tracks and compared against the lowest value obtained so far. If the new value is higher a \(\Sigma |f|\) - or \(\Sigma |\Delta x|\)-non-convergence is recorded. If the new value is lower, it replaces the odd one. There is a maximum number of permitted successive non-convergences of each type leading to an error exit of FIT if exceeded. The total
number of iterations is limited as well. If all the counters are in the permissible region but one or both of the tests fail, the next iteration is entered starting with GOTOX. If both tests are successful the iteration is stopped and HOME is entered (see below).

In addition to the \(|f|\)- and \(|\Delta x|\)-tests two checks are carried out on \(\chi^2\). Although logically they should be performed in DATEST the considerable amount of events failing this test and the amount of time saved by checking \(\chi^2\) at an earlier stage justifies sufficiently their appearance in EMT2EST. The two conditions are:

a) \(\chi^2\) must be less than an acceptance parameter
b) \(\chi^2\) must be positive.

A failure leads to an immediate error exit from FIT. A negative \(\chi^2\) can be due to an incorrect error set-up or machine failure only.

10. **HOME**

This routine contains all operations necessary after a good solution has been found. It completes the solution and, like an inverse of SETUP, communicates it in a suitable form to the higher level programme. The latter problem being too specific (see remarks in III.3), I restrict myself to a few notes on the completion of the results, i.e. the calculation of errors, correlations and related quantities.

a) No correlations between variables from different tracks are calculated. Only the complete \(3 \times 3\) error matrix for the variables of a track is evaluated.
b) To save unnecessary operations and for reasons of space requirements the following matrices only are evaluated and kept fully: $G_B$ from $BG^{-1}B^T$ and $K^{-1}$ from $K$ (the inversions are not carried out explicitly during the iteration), $G_B K^{-1} B^T$ and $G_B B^* K^{-1} B^* G_B$.

c) The necessary elements out of $(19a)$ and $(19c)$ for each track are then evaluated multiplying corresponding lines or columns of the matrices just mentioned with $G^{-1}B^T$. The procedure for connecting tracks is obvious if one follows the definition of $G^{-1}B^T$ as given in III.7.

d) The elements out of $(19b)$ for each track are obtained by picking them out of $K^{-1}$. The combination with the elements from $(19a)$ and $(19c)$ is governed by the variable description code.

e) The difference between the measured and the fitted value of each variable divided by the statistical error of this difference (sec appendix 2) is evaluated for each well or badly measured variable.

IV. Experience with the testing version

The testing version of FIT was written in FORTRAN and debugged on the IBM 709 computer at CERN under the 709 FORTRAN Monitor System. About 30 events altogether were used as testing material.

a) For some 20 single-apex events the differences between results from FIT and those obtained by the standard APEXFIT programme (for one apex only) were found very small. They are due to the first parameter in APEXFIT ($1/P\cos\lambda$) being different from the one in FIT ($1/P$).
b) The rest of the events consisted of two or three vertex-events representing either double $\pi^+$-scatterings at a beam energy of 320 MeV or $\bar{p}$-annihilations at rest with one or two visible $K^0$-decays. The results show clearly one advantage of FIT: It can shift much information along a connecting track into another apex. All errors obtained with FIT are lower than in case of single vertex fitting. (In fact this is a theoretical necessity). The advantage against fitting at a good end and introducing these results into the fit for the next vertex is the uniqueness of the results.

V. Improvements and Extensions

The testing version of FIT as described above has already started to find its way into other laboratories. I do not think, however, that any production programme will use exactly the version I have outlined. The programmes around it will be different and generally there are as many different solutions for a complex problem as there are people thinking about it. The version which will be built into the next CERN production programme (successor of GAP 1) will be different, too. Some of the ideas which have come up to me since I wrote the testing version of FIT I will describe in the following.

1. FIT now takes the three parameters of a track to be well measured, badly measured or completely unknown (see II.1). I would like to introduce a fourth possibility for them, the fixed parameter. This is defined by not being subject to any correction whatsoever and does simply not appear in the corresponding vectors and matrices. A useful application would be a reaction at rest, where the momentum is fixed at zero, or the one described below.
2. As the mass is generally such a fixed parameter it would be easy to call the mass a track variable like momentum and angles. Thus one defines the four apex constraints purely by general parameters (except the target mass). The description code for the mass would then generally tell the programme that it is a fixed parameter. In certain cases, however, it is desirable to set it "unknown" and fit the mass value for a track.

3. The assumption made in II.3. that connecting charged tracks are sufficiently described by the range-momentum-relation (momentum) and the helical shape of the track (angles) turns out to be very bad, if the tracks are either long or of low energy or if the chamber liquid is heavy. Two solutions are possible for this problem.

a) A very simple solution would be to keep the range-momentum condition at any rate and replace the angle constraints as in equ. (4) by the equations

\[ x^2_{2i} = x^s_{2i} + d^2_2 \]
\[ x^3_{3i} = x^s_{3i} + d^3_3 \]

with \( d^2_2 \) and \( d^3_3 \) being determined by the "measurement" differences, i.e. differences between purely geometrical results at the specification point and the apex. For this I assume for instance that \( L^s = L \) or \( L^s = 0 \) and that angles are given by the geometrical reconstruction at both ends of the track. The introduction of these formulae would simplify the \( \frac{dx}{dx^s} \) matrix (equa. (6)) in that apart from \( \Delta x^s_{1i}/\Delta x^s_{1i} \) all elements would be out of a unit matrix. Instead of a \( 3 \times 3 \) matrix this single term is kept only and the first column of \( \frac{df}{dx^s_1} \) is changed instead of performing the matrix multiplication in equ. (5).
b) A more general solution (suggested by D. Johnson, UCRL Berkeley) would be to have a marker for each variable indicating if it is dependent in the sense of section II.3. or independent. In this way one could create two independent ends of a connecting track. For the constraints as described in this report the dependency marker is set for all three variables of a connecting track together. In the version suggested momentum and angles would be treated separately and the angular or all connection constraints could be suppressed according to some criteria (e.g. for long tracks of low energy).

Clearly solution (b) in some cases corresponds better to the actual situation than (a). It might be dangerous, however, to let the connection information get completely out of control. FIT might be tempted to twist the two free ends of a connecting track in an unreasonable way, if no restriction is provided. This will be especially true in case of a connecting track with "unmeasured" variables.

The ideal solution for this problem would be to introduce the parameters establishing the connection like "measurements" with certain "errors" due to straggling and multiple scattering respectively. The deviations of those constraints would also have to enter the minimum function $\chi^2$. As this needs substantial changes in FIT I will try to get around it by using solution (a) above.

4. $L^8$ at present is made up once per track only. People used to KICK and GUTS (see [2]) will perhaps prefer that there should be another quantity describing the point along the track at which the momentum has been measured. (The "Endword" feature of KICK). It would be easy, of course, to introduce in FIT two different definitions of $L^8$ for the momentum and the angles respectively. This change will not be made for the CERN version of FIT.
5. DATEST and ENTEST are both not satisfactory in the form described. It is desirable that at least in DATEST somewhat more elaborate tests are built in to stop calculation for wrong mass assignments as soon as possible. On the other hand most of the data entering FIT are measurements and thus highly influenced by human and machine mistakes of all kinds - which makes it difficult to give a verdict over a certain hypothesis if no other FIT results are available at the same time.

One of the tests, however, will certainly have to be improved. The $\chi^2$-test (see section III.9.) must be carried out regarding both $\chi^2$ and the number of degrees of freedom. These two quantities together can be converted into a probability using the theoretical $\chi^2$-distribution and would thus give a reasonable test quantity. With reliable error assignments the cut can then certainly be performed at comparatively rather low $\chi^2$-values.

6. Finally it should be pointed out that HOME (section III.10) can be extended to calculate a number of functions of the fitted variables and their errors which require inter-track-correlations and hence cannot be evaluated outside FIT. Such quantities are for instance the centre-of-mass angles and moments.
References


Appendix 1

Error propagation in matrix notation

The most general form of error propagation written in matrix notation results from the well known relation giving the errors $e_{fj}$ for linear functions $f_j$ ($j = 1..n_f$) of independent measurements $m_i$ ($i = 1..n_m$) with errors $e_i$:

$$e_{fj} = \xi_i a_{ji} m_i$$  \hspace{1cm} (1)

If

$$e_{fj}^2 = e_i^2 a_{ji}^2$$ \hspace{1cm} (2)

then

$$e_{gk} = \xi_j b_{kj} f_j$$ \hspace{1cm} (3)

If we define $g_k$ ($k = 1..n_g$) to be again linear functions of the $f_j$-s defined by

$$g_k = \xi_j b_{kj} f_j$$ \hspace{1cm} (3a)

then the errors can be obtained correctly only if the $g_k$-s are written as functions of the original measurements $m_i$. This gives us

$$e_k = \xi_j b_{kj} \xi_i a_{ji} m_i$$ \hspace{1cm} (3b)

leading to

$$e_{gk}^2 = e_i^2 (\xi_j b_{kj} a_{ji})^2$$ \hspace{1cm} (4)

If we write equas. (1) thru (4) in matrix notation defining

$$(A)_{ji} = a_{ji}, \quad (B)_{kj} = b_{kj}, \quad \text{and } G^{-1}_m\text{ to be a } (n_m \times n_m) \text{ diagonal matrix with } (G^{-1}_m)_{ii} = e_i^2$$

then we obtain

$$(1') \quad f = Am$$

$$(2') \quad G^{-1}_f = A G^{-1}_m A^T$$

$$(3') \quad g = B f$$

2359/p
(3a') \[ g = B A m \]

and

(4') \[ G_m^{-1} = B A G_m^{-1} (B A)^T = B A G_m^{-1} A^T B^T \]

We can rewrite equ. (4')

(4a') \[ G_g^{-1} = B G_f^{-1} B^T \]

thus having a means to express the error matrix of a function of statistically correlated variables as a function of the error matrix of these variables only. This is possible because (2') and (4') are more general than (2) and (4), which only give the diagonal elements out of \( G_f^{-1} \) and \( G_g^{-1} \) respectively.

The generalization for non-linear functions is obvious and gives for a function

(5) \[ f = f(x) \]

the error matrix

(6) \[ G_f^{-1} = \left( \frac{df}{dx} \right) \left( \frac{df}{dx} \right)^T G_x^{-1} \left( \frac{df}{dx} \right) \]

if (5) is linearized using a Taylor series

\[ f = f(x_0) + \left( \frac{df}{dx} \right) (x - x_0) \]

with the definition \[ \left( \frac{df}{dx} \right)_{ij} = \left( \frac{\partial f_i}{\partial x_j} \right) \].
Appendix 2

Derivatives and Error Matrices for fitted quantities

This appendix is related to section II.4.b and equation numbers refer to those used in that section.

In evaluation of (18a), (18b) and (18c) the derivatives of \( m + c \) and \( m^* + c^* \) are used with respect to \( m \) and \( m^* \). In this meaning \( m^* + c^* \) has to include elements corresponding to unmeasured quantities whereas \( m^* \) refers to (badly) measured quantities only. For this reason I use in the following the convention, that quantities with a bar underneath are defined as in section II.4, but with the columns corresponding to badly measured variables only. If the lines corresponding to unmeasured variables are missing I write a bar on top of the symbol. For the derivatives we obtain from (15) and (16)

\[
\frac{d (m + c)}{dm} = E - G^{-1} B^T \frac{d\alpha}{dm} \quad (E = \text{unit matrix})
\]

\[
= E - G^{-1} B^T G_B \left( B^* \frac{dc^*}{dm} + \frac{dr}{dm} \right)
\]

from (17) one obtains

\[
\frac{dc^*}{dm} = -K^{-1} B^* G_B \frac{dr}{dm}
\]

and from (12a) and the definition of \( B \)

\[
\frac{dr}{dm} = \frac{df}{dm} = B
\]

We have thus found the expression for \( \frac{d (m + c)}{dm} \) and

\[
\frac{d (m^* + c^*)}{dm} = \frac{dc^*}{dm}.
\]
Similarly we get from (15) and (16)

\[
d\frac{(m + c)}{dm^*} = -G^{-1}B^T\frac{d\alpha}{dm^*}
\]

\[
= -G^{-1}B^T G_B (B^* \frac{dc^*}{dm^*} + \frac{dr}{dm^*})
\]

Using (17) one can write

\[
\frac{dc^*}{dm^*} = -K^{-1}B^*G_B \frac{dr}{dm^*}
\]

and as above one substitutes

\[
\frac{df}{dm^*} = \frac{df}{dm^*} = B^*
\]

so that the final results are:

\[
d\frac{(m^* + c^*)}{dm^*} = E - K^{-1}B^*G_B B^* = K^{-1}G^*
\]

(as \(E\) can be written as \(K^{-1}K\))

\[
d\frac{(m^* + c^*)}{dm} = -K^{-1}B^*G_B B
\]

\[
d\frac{(m + c)}{dm^*} = G^{-1}B^T G_B (B^* K^{-1}B^T G_B B^* - B^*) = -G^{-1}B^T G_B B^* K^{-1}G^*
\]

\[
d\frac{(m + c)}{dm} = E + G^{-1}B^T G_B (B^* K^{-1}B^T G_B B - B)
\]

Introducing the above derivatives into the propagation formulae (18) the result is

\[
G_m^{-1} = K^{-1}G^* G^{-1} G^* K^{-1} + K^{-1}B^T G_B B G^{-1}B^T G_B B^* K^{-1}
\]

\[
= K^{-1}G^* K^{-1} + K^{-1}B^T G_B B^* K^{-1} = K^{-1}(G^* + B^* G_B B^*) K^{-1}
\]

\[
= K^{-1}
\]
\[ G_{m+c}^{-1} = G^{-1} B^T G B B^* K^{-1} G^* G_{m+c}^{-1} G^* K^{-1} B^T G B B^{-1} \]
\[ + G^{-1} B^T G B B^* K^{-1} B^* G B B^{-1} B \]
\[ + G^{-1} B^T G B B^* K^{-1} B^* G (B^T G B B - B) B \]
\[ + G^{-1} B^T G B (B^* K^{-1} B^* B^* G (B^T G B B - B)) B^{-1} (B^T G B B^* K^{-1} B^* G - B^T G B B)^{-1} \]
\[ = G^{-1} B^T G B B^* K^{-1} G^* K^{-1} B^* G B B^{-1} + G^{-1} \]
\[ + G^{-1} B^T G B B^* K^{-1} B^* G B G^{-1} - G^{-1} B^T G B B G^{-1} \]
\[ + G^{-1} B^T G B B^* K^{-1} B^* G B B^{-1} - G^{-1} B^T G B B G^{-1} \]
\[ + G^{-1} B^T G B B^* K^{-1} B^* G (B G^{-1} B^T G B B) B^{-1} \]
\[ - G^{-1} B^T G B B^* K^{-1} B^* G (B G^{-1} B^T G B B) B^{-1} \]
\[ - G^{-1} B^T G B B^* K^{-1} B^* G B B^{-1} \]
\[ + G^{-1} B^T G B B^* K^{-1} B^* G B G^{-1} \]
\[ = G^{-1} B^T G B B^* K^{-1} G G^* B^* G B B^{-1} + G^{-1} \]
\[ - G^{-1} B^T G B B G^{-1} \]
\[ + G^{-1} B^T G B B^* K^{-1} B^* G B B^* K^{-1} B^* G B G^{-1} \]
\[ = G^{-1} G^* G_{m+c}^{-1} G^* G^{-1} \]

It is interesting that you can write
\[ G_{m+c}^{-1} = G^{-1} - G_{m+c}^{-1} \]

which is easy to show using
\[ \frac{d G}{d m} = \frac{d(m+c)}{d m} \text{ and } \frac{d G}{d m} = \frac{d(m+c)}{d m} - E \]
Appendix 3

Flow Chart of FIT

FIT IN

PARAM
SET CRITERIA,
TOLERANCES,
COUNTERS

CONNY
SET DEPENDENT
VARIABLES

SETUP
GET LISTS
AND STARTING
VALUES

ERROR
O.K.

GOTOX
TRANSFORM
TO APICES

DATEST
TEST
VARIABLES

NEWVAL
SOLVE FOR
NEW VALUES
AND \( \chi^2 \)

MAXIES
GET DERIVA-
TIVES AND
MATRICES

ENTEST
TEST IF
SOLUTION
SATISFACTORY

MORE
STEPS
REQU Required

NO SOLUTION
POSSIBLE

HOME
CALCULATE
ERROR MATRICES
& STORE RE-
SULTS

FIT OUT
Appendix 4

Numerical Example

The following event was the first multi-vertex event used in debugging FIT and consists of two consecutive $\pi^+$ scatters in the CERN 30 cm Hydrogen chamber (320 MeV $\pi^+$ beam); it was calculated:

a) by fitting each vertex separately with the information coming from the geometry,

b) by fitting the event as a whole using FIT.

The following table needs no explanation except that the three rows of each track give the measurements, the results for single vertex fitting and the results from FIT.

One track (labelled BD) did not converge in the geometry for some reason (let's assume it was an error of the measuring machine); hence the parameters of these tracks were "unmeasured" and no values are listed in the "measurement" row.
The example is somewhat unfortunate in that the errors of the angle are obviously underestimated. This results in a large $\chi^2$ for apex A. It shows clearly, however, how much is gained by increasing the number of degrees of freedom in the errors, particularly for the momenta and for apex B. One of the three photographs used for the measurement of this event is shown on the next page.
EXAMPLE AS GIVEN IN APPENDIX 4