

VERTEX FITTING OF SEVERAL HELICES IN SPACE

G.N. PATRICK

Rutherford Appleton Laboratory, Chilton, Didcot, Oxon, UK

B. SCHORR

CERN, Geneva, Switzerland

Received 13 June 1985

A method is presented for the determination of the common interaction vertex of high-energy particle tracks for the case where the spectrometer target is situated in a strong magnetic field. A least-squares technique is used to fit the vertex position and provide improved estimates of the individual track parameters subject to several constraints. Examples of the method for various track multiplicities are discussed.

1. Introduction

When fitting the measured tracks of elementary particles originating from a common vertex (intersection point) the fitting process is often performed in two stages. Firstly, the individual tracks are fitted independently to model curves and secondly, the parameters of the fitted model curves are modified within certain error limits so that the curves intersect at a common vertex.

This paper deals with the second step for the case where the vertex is situated within a strong dipolar magnetic field. In particular, the following simplifying assumptions have been made.

- (a) The magnetic field B in the volume where the particles are detected is constant and homogeneous such that their tracks can be represented by helicoidal trajectories.
- (b) The errors of the track measurements have a Gaussian distribution.
- (c) Least-squares fitting is applied to each track independently to determine estimates of the helix parameters and the covariance matrices of the estimates.
- (d) All observed tracks emanate from a single, common vertex.

Under these assumptions, and given the estimates from point c, a method is proposed to find the least-squares estimate of the common vertex for $n \geq 2$ helices. A computer program in FORTRAN IV has been written in order to test the method. Results are discussed in sect. 5.

The case of $n = 2$ helices has been handled by Royston and Gregory [1] in 1966 using Lagrange multipliers and applying Newton's method to minimize the Lagrange function of the problem. For $n > 2$ this method is too expensive. In the present paper an alternative method is therefore proposed which reduces the original linear least-squares problem with $7n + 3$ unknowns and $3n$ nonlinear constraints to an unconstrained nonlinear least squares problem with $4n + 3$ unknowns. In order to solve the latter problem numerically a generalization of Moré's [2] implementation of the Levenberg-Marquardt method is proposed.

In sect. 2 the mathematical problem is formulated. Sect. 3 discusses the Levenberg-Marquardt algorithm and sect. 4 gives numerical results of the calculated examples.

2. The mathematical problem

Consider n helices originating from the common vertex (x, y, z) with the parameter representation as illustrated in fig. 1 and given by

$$\begin{aligned}
 X_i &= x_i + r_i (\cos \beta - \cos \beta_i), \\
 Y_i &= y_i + r_i (\sin \beta - \sin \beta_i), \\
 Z_i &= z_i + s_i r_i (\beta - \beta_i) \tan \lambda_i,
 \end{aligned}
 \tag{1}$$

where

- i = 1, 2, . . . , n ,
- x_i, y_i, z_i = the point on the helix for azimuthal angle $\beta = \beta_i$,
- λ_i = dip angle,
- r_i = radius of curvature,
- s_i = +1 or -1 according to whether the particle is moving counterclockwise or clockwise.

In physical terms,

$$r_i = \frac{c \cos \lambda_i}{\alpha_i},
 \tag{2}$$

with

$$\begin{aligned}
 c &= \frac{\text{const.}}{|B|}, \quad (\text{const.} = 3 \times 10^{-4}), \\
 \alpha_i &= \frac{1}{|p_i|},
 \end{aligned}$$

where B is the magnetic field (whose major component is defined along the positive z -axis), and p_i is the momentum of the particle.

In most practical applications it can be assumed that both one coordinate, say x_i , and s_i are known exactly. For our algorithm, however, it is more appropriate to assume that x_i is not known precisely. The reason for that will become clear later.

For the unknown (column) vectors $\mathbf{h}_i = (x_i, y_i, z_i, \alpha_i, \beta_i, \lambda_i)$, it may be assumed that estimates $\tilde{\mathbf{h}}_i$, together with their (positive definite) covariance matrices \mathbf{C}_i^{-1} are available from independent least-squares fits of the individual tracks. From the general assumptions (a) to (d) in sect. 1 it then follows that the error vectors

$$\boldsymbol{\epsilon}_i = \tilde{\mathbf{h}}_i - \mathbf{h}_i, \quad i = 1, 2, \dots, n,
 \tag{3}$$

are independent of each other and have a six-dimensional normal distribution each with mean vector zero and (approximately) known covariance matrix \mathbf{C}_i^{-1} .

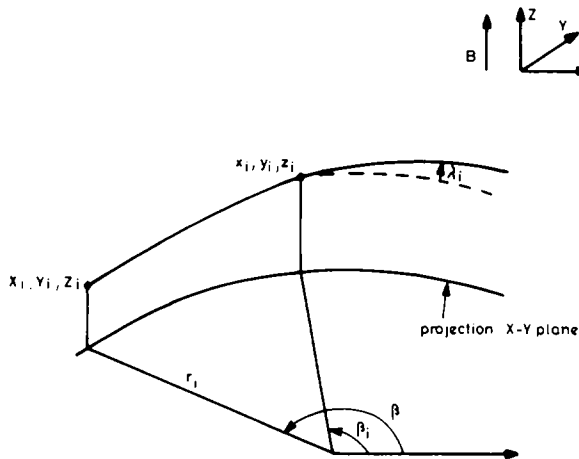


Fig. 1. Schematic representation of track helix and associated parameters.

By construction, the helices with parameters $\tilde{\mathbf{h}}_i$, do not, in general, have a common vertex. Therefore, new parameters \mathbf{h}_i^0 are determined such that they fit best, in the least-squares sense, to $\tilde{\mathbf{h}}_i$, and the resulting helices have a common vertex. This problem is called the *vertex problem*.

Mathematically the vertex problem can be formulated as follows.

Minimize the quadratic function

$$\Phi(\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_n) = \frac{1}{2} \sum_{i=1}^n \epsilon_i' \mathbf{C}_i \epsilon_i \quad (4)$$

under the constraints

$$\begin{aligned} x_i + r_i(\cos v_i - \cos \beta_i) &= x, \\ y_i + r_i(\sin v_i - \sin \beta_i) &= y, \\ z_i + s_i r_i(v_i - \beta_i) \tan \lambda_i &= z, \end{aligned} \quad (5)$$

for $i = 1, 2, \dots, n$, where v_i are the unknown azimuthal track angles at the unknown vertex (x, y, z) .

This is a linear least-squares problem with $3n$ nonlinear constraints and $7n + 3$ unknowns. Its main complication lies in the presence of the constraints (5). However, according to (2) r_i only depends on α_i and λ_i such that x_i , y_i and z_i can be written as functions of $(\alpha_i, \beta_i, \lambda_i, v_i)$ and (x, y, z) by rewriting (5) as

$$\begin{aligned} x_i &= x - r_i(\cos v_i - \cos \beta_i), \\ y_i &= y - r_i(\sin v_i - \sin \beta_i), \\ z_i &= z - s_i r_i(v_i - \beta_i) \tan \lambda_i, \end{aligned} \quad (6)$$

for $i = 1, 2, \dots, n$. Insertion of (6) into (4) makes the problem an unconstrained nonlinear least squares problem with $4n + 3$ unknowns. In the next section a method is proposed to solve this problem numerically.

3. The Levenberg–Marquardt algorithm

For convenience new notations are introduced. The $m = 4n + 3$ unknowns are combined into a vector $\mathbf{a} = (a_1, a_2, \dots, a_m)'$ by setting

$$\begin{aligned} a_1 &= \alpha_1, & a_2 &= \beta_1, & a_3 &= \lambda_1, & a_4 &= v_1, \\ a_5 &= \alpha_2, & a_6 &= \beta_2, & a_7 &= \lambda_2, & a_8 &= v_2, \\ &\vdots & &\vdots & &\vdots & &\vdots \\ a_{4n-3} &= \alpha_n, & a_{4n-2} &= \beta_n, & a_{4n-1} &= \lambda_n, & a_{4n} &= v_n, \\ a_{4n+1} &= x, & a_{4n+2} &= y, & a_{4n+3} &= z. \end{aligned}$$

The eqs. (6) can then be written as

$$\begin{aligned} x_i &= a_{4n+1} - r_i(\cos a_{4i} - \cos a_{4i-2}), \\ y_i &= a_{4n+2} - r_i(\sin a_{4i} - \sin a_{4i-2}), \\ z_i &= a_{4n+3} - s_i r_i(a_{4i} - a_{4i-2}) \tan a_{4i-1}, \end{aligned} \quad (7)$$

with

$$r_i = \frac{c \cos a_{4i-1}}{a_{4i-3}}, \quad (8)$$

where (8) follows from (2).

The components of the vectors

$$\mathbf{h}_i = (x_i, y_i, z_i, \alpha_i, \beta_i, \lambda_i)' = (x_i, y_i, z_i, a_{4i-3}, a_{4i-2}, a_{4i-1})'$$

for $i = 1, 2, \dots, n$ are combined into a vector $\mathbf{Y} = (Y_1, Y_2, \dots, Y_{6n})'$ by setting

$$\begin{array}{cccccc} Y_1 = x_1, & Y_2 = y_1, & Y_3 = z_1, & Y_4 = a_1, & Y_5 = a_2, & Y_6 = a_3, \\ Y_7 = x_2, & Y_8 = y_2, & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ Y_{4n-5} = x_n, & \dots & \dots & \dots & \dots & Y_{4n-1} = a_{4n-1} \end{array}$$

From (7) to (9) it follows that $\mathbf{Y} = \mathbf{Y}(\mathbf{a})$.

Finally, the square matrix \mathbf{C} of dimension $6n$ is introduced. It is a block diagonal matrix with \mathbf{C}_i , $i = 1, 2, \dots, n$ as diagonal blocks.

With these notations the function Φ in (4) becomes a function of \mathbf{a} (for which no new symbol is introduced) which can be written as

$$\Phi(\mathbf{a}) = \frac{1}{2}(\tilde{\mathbf{Y}} - \mathbf{Y}(\mathbf{a}))'\mathbf{C}(\tilde{\mathbf{Y}} - \mathbf{Y}(\mathbf{a})), \quad (10)$$

where $\tilde{\mathbf{Y}}$ are the given quantities from the independent least-squares fits of the n helices.

In order to minimize $\Phi(\mathbf{a})$ Moré [2] proposed a special implementation of a method known under the name Levenberg–Marquardt method. He did it for the case where \mathbf{C} is equal to the identity matrix. His algorithm, however, can easily be generalized to the case given in (10). A brief outline of the Levenberg–Marquardt method is given here. Details of the implementation can be found in ref. [2].

If, for given \mathbf{a} , one could minimize

$$\Phi(\mathbf{a} + \mathbf{p}) = \frac{1}{2}(\tilde{\mathbf{Y}} - \mathbf{Y}(\mathbf{a} + \mathbf{p}))'\mathbf{C}(\tilde{\mathbf{Y}} - \mathbf{Y}(\mathbf{a} + \mathbf{p})) \quad (11)$$

as a function of \mathbf{p} , then $\mathbf{a} + \mathbf{p}$ would be the solution of the minimization problem. Suppose \mathbf{p} is in some sense small such that $\mathbf{Y}(\mathbf{a} + \mathbf{p})$ may be linearized.

Consider

$$\phi(\mathbf{p}) = \frac{1}{2}(\tilde{\mathbf{Y}} - \mathbf{Y}(\mathbf{a}) - \mathbf{J}\mathbf{p})'\mathbf{C}(\tilde{\mathbf{Y}} - \mathbf{Y}(\mathbf{a}) - \mathbf{J}\mathbf{p}), \quad (12)$$

where

$$\mathbf{J} = \mathbf{J}(\mathbf{a}) = \mathbf{Y}'(\mathbf{a})$$

is the Jacobian matrix.

Minimizing ϕ in (12) is a linear least-squares problem for which the explicit solution can be given. The method which uses (12) iteratively to find the minimum of (11) is called the Gauss–Newton method. It works well and fast if the initial \mathbf{a} is in a reasonable neighbourhood of the solution. However, if (12) is a bad approximation of (11) the Gauss–Newton method does not work. In this case the Levenberg–Marquardt method solves the following constrained linear least-squares problem instead:

$$\min_{\mathbf{p}} \{ \phi(\mathbf{p}) : \mathbf{p}'\mathbf{D}'\mathbf{D}\mathbf{p} \leq \Delta \},$$

where Δ is an appropriately chosen constant and \mathbf{D} is a diagonal matrix which takes into account the scaling of the problem. If $\mathbf{J}'\mathbf{C}\mathbf{J}$ is not rank deficient it can be shown that the solution of (12) is given by

$$\mathbf{p} = \mathbf{p}(\alpha) = (\mathbf{J}'\mathbf{C}\mathbf{J} + \alpha\mathbf{D}'\mathbf{D})^{-1}\mathbf{J}'\mathbf{C}(\tilde{\mathbf{Y}} - \mathbf{Y}(\mathbf{a})) \quad (13)$$

for some $\alpha \geq 0$. There are two possibilities. Either $\alpha = 0$ and $\mathbf{p}'\mathbf{D}'\mathbf{D}\mathbf{p} \leq \Delta$ (Gauss–Newton case). Or $\alpha > 0$ and $\mathbf{p}'\mathbf{D}'\mathbf{D}\mathbf{p} = \Delta$.

Moré's implementation of the Levenberg–Marquardt method includes rules how to choose Δ and \mathbf{D} in each iteration step and uses a simple method proposed by Hebden [3] to compute the parameter $\alpha \geq 0$ in

each step in an efficient way. That means, if a Gauss–Newton step yields $\mathbf{p}'\mathbf{D}'\mathbf{D}\mathbf{p} > \Delta$ (with $\mathbf{p} = \mathbf{p}(0)$ in (13)), $\alpha > 0$ has to be determined such that $\mathbf{p}'(\alpha)\mathbf{D}'\mathbf{D}\mathbf{p}(\alpha) = \Delta$, where $\mathbf{p}(\alpha)$ is given by (13). Hebden's method solves this equation economically.

Details about Moré's algorithm are omitted here. They can be found in [2].

In the next section results are given for examples of simulated events.

4. Results of examples

Events from three elastic photoproduction channels were generated by the Monte Carlo method using the program SAGE [4]:

(a) $\gamma p \rightarrow \rho(776)p \rightarrow \pi^+ \pi^- p$,

where $20 < E_\gamma < 70$ GeV;

(b) $\gamma p \rightarrow \rho'(1500)p \rightarrow \pi^+ \pi^- \pi^+ \pi^- p$,

where $20 < E_\gamma < 70$ GeV;

(c) $\gamma p \rightarrow M_x p \rightarrow \pi^+ \pi^- \pi^+ \pi^- \pi^+ \pi^- \pi^+ \pi^- \pi^+ \pi^- p$,

where $150 < E_\gamma < 170$ GeV,

and M_x is a "jetlike" object with a mass generated between 7 and 8 GeV/ c^2 according to P , truncated phase-space.

These events were then processed through the GEANT [5] program, to simulate the geometric acceptance and resolution of the charged particle detectors comprising the OMEGA' spectrometer situated in the CERN West experimental area. In each case, the interaction vertices were generated uniformly over a 60 cm long cylindrical hydrogen target of radius 2.4 cm. This target was situated within the magnetic field of the spectrometer as illustrated by the experimental geometry shown in fig. 2. The following simplifying conditions were made:

(a) Multiple scattering and energy losses of the charged particles were ignored.

(b) A constant magnetic field of 19 kG was assumed throughout the volume of the spectrometer magnet.

Finally, the simulated detector digitisings produced by this stage were analysed by the event reconstruction program TRIDENT [6]. This program performed the pattern recognition of the charged tracks, and fitted their individual trajectories to estimate their momentum vectors and covariance matrices at a fixed point down-stream (x -direction) of the interaction vertex. Cuts were applied to the reconstructed events to obtain clean samples for testing the above algorithm.

Since in sect. 2 it was assumed that all x_i had measurement errors in order to be able to formulate the problem as an unconstrained problem, we gave x_i an artificial variance of 0.1 which corresponds to a relative error of less than 1% in our examples.

This is equivalent to using a penalty function method for the first n constraints in eqns. (5). The examples will show that this approach is adequate to determine the vertex with sufficient precision.

A FORTRAN program was written in single precision and run on a CDC 7600. The normal equations for the linear least-squares problem in each iteration step were solved by Cholesky decomposition. QR decomposition with column pivoting as suggested by Moré was not used simply because a library routine for the Cholesky decomposition was available whilst for the QR decomposition no such routine was available.

The initial values for the unknowns were chosen as

$$\alpha_i = \bar{\alpha}_i, \quad \beta_i = \bar{\beta}_i, \quad \lambda_i = \bar{\lambda}_i, \quad v_i = 0.99 \text{ sign}(s_i) \bar{\beta}_i$$

for $i = 1, 2, \dots, n$. For the unknown vertex the midpoint of the target was taken.

The program terminated after the k th iteration step if the following criteria were satisfied simulta-

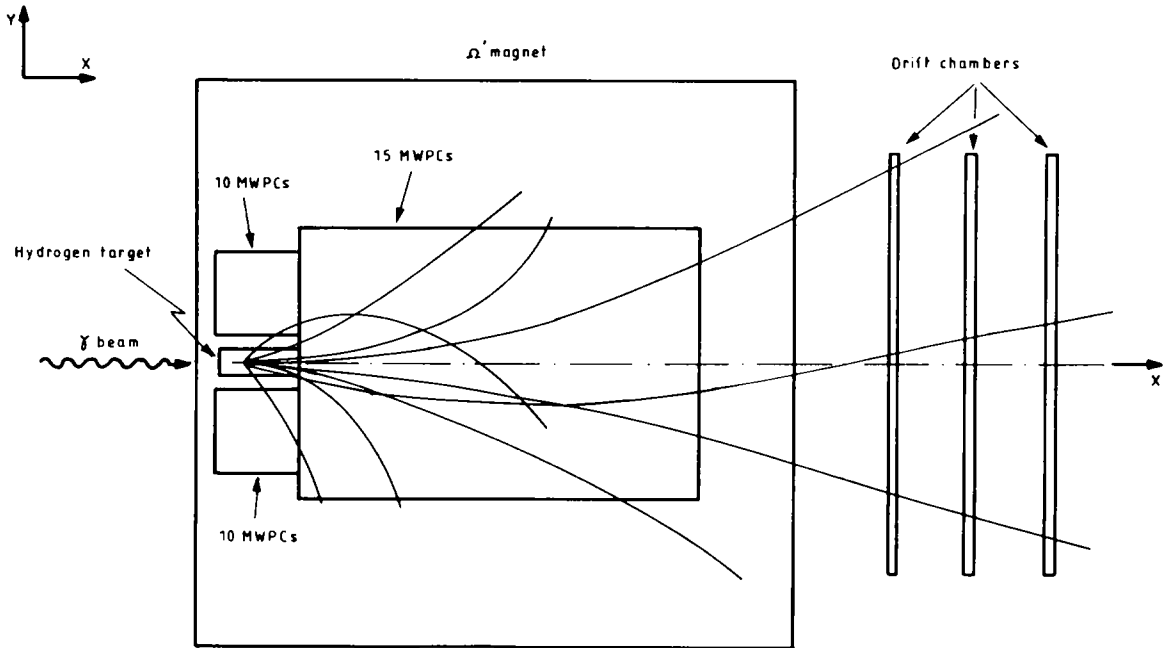


Fig. 2. Experimental set-up of OMEGA' spectrometer for simulated events.

neously:

$$\phi_k - \phi_{k-1} \leq \epsilon(1 + \phi_k),$$

$$\|\mathbf{D}_k \mathbf{p}_k\| \leq \epsilon^{1/2}(1 + \|\mathbf{a}_k\|),$$

$$\|\text{grad } \phi_k\| \leq \epsilon^{1/3} \phi_k,$$

with $\epsilon = 10^{-6}$.

The scaling diagonal matrices \mathbf{D}_m were computed as described by Moré [2] (adaptive scaling). The initial value of the constant Δ for the Levenberg–Marquardt method was taken as one tenth of

$$\max(\|\mathbf{D}_0 \mathbf{a}\|, \|\mathbf{D}_0 \mathbf{p}\|),$$

where \mathbf{D}_0 is the initial scaling matrix, \mathbf{a} the starting vector as described above and \mathbf{p} the solution vector of the Gauss–Newton step ($\alpha = 0$) at the beginning of the iteration procedure. Updating of Δ was done as described by Moré [2].

There was data available for $n = 2, 4, 13$ and 15 . Table 1 shows the average number of iterations for the different events available for which the algorithm converged. It must be said that there were some cases for which the method did not converge. This was most probably due to near singularity of some error matrices.

Table 1
Number of iterations for various track multiplicities n

Reconstructed multiplicity n	Number of events	Number of iterations for convergence
2	14	4–9
4	2	7
13	4 (1)	7–9 (21)
15	1	6

The table shows that convergence was reached in less than ten iterations apart from one case with $n = 13$ where 21 iterations were needed.

The program also returned estimates for the standard deviations of (x, y, z) . The estimate was always within 2 to 3 standard deviations of the true solution apart from one case with $n = 2$ where the error limits were 10 times too small.

Summarizing, we may say that these tests show that the appropriate modification of Moré's implementation of the Levenberg–Marquardt method yields promising results for the vertex fitting problem.

References

- [1] R.J. Royston and J. Gegory, *Com. ACM* 9 (1966) 280.
- [2] J.J. Moré, *Proc. Dundee Conf. on Numerical Analysis* (1977) ed., G.A. Watson, *Lecture Notes in Mathematics* 630 (Springer, Berlin, 1978).
- [3] M.D. Hebden, *United Kingdom Atomic Energy Research Establishment, Report TP 515*, Harwell, England (1973).
- [4] J. Friedman, *SAGE Reference Manual*, SLAC Computation Group Technical Memo 145 (1972).
- [5] R. Brun et al., *CERN DD/US/86* (June 1982).
- [6] J.C. Lassalle et al., *CERN DD/EE/79-2* (April 1982).