Nuclear Instruments and Methods in Physics Research A311 (1992) 139–150
North-Holland

NUCLEAR INSTRUMENTS & METHODS IN PHYSICS RESEARCH
Section A

Fast vertex fitting with a local parametrization of tracks

P. Billoir
LPNHE, Universités Paris VI et VII, Paris, France

S. Qian
CERN, Geneva, Switzerland / INFN, Frascati, Italy [1]

Received 17 May 1991 and in revised form 11 July 1991

A fast vertex fitting algorithm has been tested, which uses a local parametrization of tracks around a fixed point. This point is chosen to be close to the primary and short-lived secondary vertices, or to the decay vertex of a long-lived particle; hence useful approximations may be utilised. The extrapolation of the trajectory and its error matrix to the vertex region is performed once per track, even if the track association procedure needs to be iterated. The time spent on the computation of the fitting procedure is proportional to the number of tracks (n) instead of n^3 as in a standard least-squares method where the parameters of all tracks are fitted together at their common vertex. The method is used in the LEP experiment DELPHI at CERN. The Monte Carlo test has been made for the ZEUS experiment at DESY. The vertex fitting quality and the time consumption etc. have been examined.

1. Introduction

The vertex reconstruction (i.e. vertex finding and fitting) is an important task for studying the heavy flavour physics in experiments at high energy colliders. In order to effectively separate the secondary vertices from the primary one, all vertices have to be well reconstructed. In present and future colliders, the charged track multiplicity at the primary vertex can be quite high (in the order of 10 to 100); thus to have a fast vertex fitting algorithm is not only desirable, but also a necessity. This article presents a test for a fast vertex fitting method [1] using a local parametrization of tracks.

The goal of a "full" 3-dimensional vertex fit is to obtain the vertex position V and n 3-vectors track momenta $p_i (i = 1 \sim n)$ at the vertex as well as the associated covariance matrices. The input information for the vertex fit consists of 5-parameters q_i (for 3-D tracks curved in a magnetic field) and their covariance matrix $\mathbf{C}_i$ of each track at a reference point (fig. 1). The local (i.e. "perigee") parameters defined in section 3 describe the trajectory close to the origin point which is a first approximation of the primary or the short-lived secondary vertices; therefore the relationship between q_i and (V, p_i) is simple if the curvature effect may be neglected, or approximated at the first order. Also the

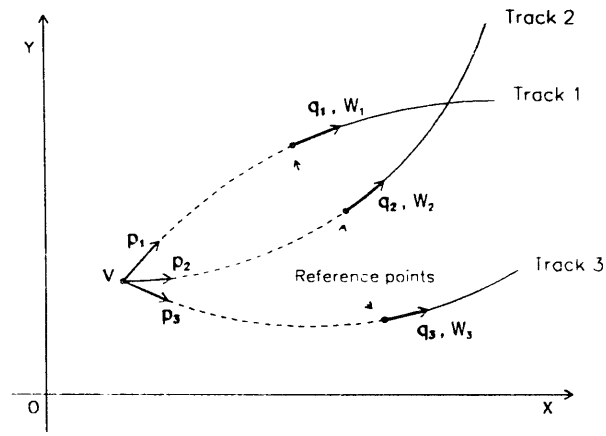

Fig. 1. Relation between q_i and (V, p_i) in a vertex fit.

"perigee" parameters themselves carry interesting physical information: the impact parameter and a good approximation of the track direction at the vertex. Moreover, by using the "perigee" parametrization, the "full fit" (i.e. the track momenta are re-evaluated with the constraint that the trajectories go through the fitted vertex position) can be reduced to an even faster "simple fit", which only estimates the vertex position without re-adjusting the track parameters.

In this article, the "full" vertex fitting method is briefly reviewed in section 2. The perigee parametrization of tracks is defined in section 3. The "simple" method for vertex estimation is derived in section 4. The test has used simulated tracks from the central

[1] Now in LAA group at CERN, Geneva, Switzerland.

tracking detector (CTD) and the vertex detector (VXD) of the ZEUS experiment at DESY. Section 5 explains the preparation of the test. Section 6 gives the test results. The conclusions are drawn in section 7.

2. The "full" vertex fitting algorithm

The method [1] can be summarized as follows. The charged track i is characterized by the parameters $q_{ij}(j = 1 \sim 5)$ and their weight matrix $\mathbf{W}_i$ ($= \mathbf{C}_i^{-1}$, i.e. the inverse of the covariance matrix) at a reference point. The q_{ij} are the functions of the vertex position $V \equiv (x_v, y_v, z_v)$ and the track momenta $p_{im}(m = 1 \sim 3)$ at the vertex, i.e. $q_{ij} = F_j(V, p_i)$. The goal is to find the V and p_i which minimize the χ^2,

$$\chi^2 = \sum_i \Delta q_i^{\mathrm{T}} \mathbf{W}_i \Delta q_i,$$

where

$$\Delta q_i = q_i^{\text{measured}} - F(V, p_i). \tag{1}$$

If q_i, within measurement errors, are linearly related to variations δV and δp_i, i.e.

$$F(V^0 + \delta V, p_i^0 + \delta p_i) = F(V^0, p_i^0) + \mathbf{D}_i \delta V + \mathbf{E}_i \delta p_i, \tag{2}$$

where δV and δp_i are variations around first approximations V^0 and p_i^0,

$$(D_i)_{jn} \equiv \frac{\partial F_j(V, p_i)}{\partial V_n},$$

$$(E_j)_{jm} \equiv \frac{\partial F_j(V, p_i)}{\partial p_{im}}, \quad (j = 1 \sim 5,\ n = 1 \sim 3,\ m = 1 \sim 3),$$

then eq. (1) can be expressed as

$$\chi^2 = \sum_i (\delta q_i - \mathbf{D}_i \delta V - \mathbf{E}_i \delta p_i)^{\mathrm{T}} \times \mathbf{W}_i (\delta q_i - \mathbf{D}_i \delta V - \mathbf{E}_i \delta p_i), \tag{3}$$

where $\delta q_i = q_i^{\text{measured}} - F(V^0, p_i^0)$. Minimizing the χ^2 w.r.t. V, i.e. $\partial \chi^2 / \partial V = 0$, we get

$$\left(\sum_i \mathbf{D}_i^{\mathrm{T}} \mathbf{W}_i \mathbf{D}_i\right) \delta V + \sum_i (\mathbf{D}_i^{\mathrm{T}} \mathbf{W}_i \mathbf{E}_i) \delta p_i = \sum_i \mathbf{D}_i^{\mathrm{T}} \mathbf{W}_i \delta q_i. \tag{4}$$

Minimizing the χ^2 w.r.t. p_i, i.e. $\partial \chi^2 / \partial p_i = 0$, we get

$$(\mathbf{E}_i^{\mathrm{T}} \mathbf{W}_i \mathbf{D}_i) \delta V + (\mathbf{E}_i^{\mathrm{T}} \mathbf{W}_i \mathbf{E}_i) \delta p_i = \mathbf{E}_i^{\mathrm{T}} \mathbf{W}_i \delta q_i. \tag{5}$$

The above two equations can be rewritten as

$$\begin{cases} \mathbf{A} \delta V + \sum_i \mathbf{B}_i \delta p_i = \mathbf{T} \\ \mathbf{B}_i^{\mathrm{T}} \delta V + \mathbf{C}_i \delta p_i = \mathbf{U}_i \end{cases}, \tag{6}$$

where

$$\mathbf{A} \equiv \sum_i \mathbf{D}_i^{\mathrm{T}} \mathbf{W}_i \mathbf{D}_i,\ \mathbf{B}_i \equiv \mathbf{D}_i^{\mathrm{T}} \mathbf{W}_i \mathbf{E}_i,\ \mathbf{C}_i \equiv \mathbf{E}_i^{\mathrm{T}} \mathbf{W}_i \mathbf{E}_i,$$

$$\mathbf{T} \equiv \sum_i \mathbf{D}_i^{\mathrm{T}} \mathbf{W}_i \delta q_i,\ \mathbf{U}_i \equiv \mathbf{E}_i^{\mathrm{T}} \mathbf{W}_i \delta q_i.$$

The solution for vertex position is

$$\delta V = \left(\mathbf{A} - \sum_i \mathbf{B}_i \mathbf{C}_i^{-1} \mathbf{B}_i^{\mathrm{T}}\right)^{-1} \left(\mathbf{T} - \sum_i \mathbf{B}_i \mathbf{C}_i^{-1} \mathbf{U}_i\right). \tag{7}$$

Notice that

$$\left(\mathbf{A} - \sum_i \mathbf{B}_i C_i^{-1} \mathbf{B}_i^{\mathrm{T}}\right)^{-1}$$

in eq. (7) is just the covariance matrix (i.e. Cov(V, V)) associated to δV. Substituting δV back into eq. (6), δp_i and Cov(p_i, p_j) as well as Cov(V, p_i) can be obtained [1].

Eq. (6) may be written in a matrix form as

$$\begin{pmatrix} \mathbf{A} & \mathbf{B}_1 & \mathbf{B}_2 & & \cdots & \mathbf{B}_n \\ \mathbf{B}_1^{\mathrm{T}} & \mathbf{C}_1 & 0 & & \cdots & 0 \\ \mathbf{B}_2^{\mathrm{T}} & 0 & \mathbf{C}_2 & 0 & \cdots & 0 \\ \cdot & \cdot & 0 & \cdot & & 0 \\ \vdots & \vdots & \vdots & & \ddots & \vdots \\ \mathbf{B}_n^{\mathrm{T}} & 0 & 0 & 0 & \cdots & \mathbf{C}_n \end{pmatrix} \begin{pmatrix} \delta V \\ \delta p_1 \\ \delta p_2 \\ \cdot \\ \vdots \\ \delta p_n \end{pmatrix} = \begin{pmatrix} \mathbf{T} \\ \mathbf{U}_1 \\ \mathbf{U}_2 \\ \cdot \\ \vdots \\ \mathbf{U}_n \end{pmatrix}. \tag{8}$$

Standard least-squares method for vertex fitting relies on inverting this $(3n + 3) \times (3n + 3)$ matrix (or a $(4n + 3) \times (4n + 3)$ matrix where multiple scattering is considered by introducing an extra parameter for each track [2]) with a number of operations proportional to n^3 (where n is the number of tracks). In contrast, our method uses only small matrices and the computing time is merely proportional to n (as shown by the results in section 6). Hence it is particularly suitable to be applied in an environment with high track multiplicity.

Another advantage of our method emerges when compared with the vertex fitting method described in ref. [2]: we handle the multiple scattering (m/s) only in the preliminary phase (i.e. the extrapolation of the covariance matrix of tracks to the vertex region, this is explained at the end of section 3). As this phase is not part of the vertex fitting procedure itself, m/s will not slow down the vertex fitting at all.

Note also that if the matrices $\mathbf{A}$, $\mathbf{B}_i$, $\mathbf{C}_i$, $\mathbf{T}$ and $\mathbf{U}_i$ are kept in memory, it is easy to add a track to (or to remove a track from) the fitted vertex, without total re-computation from the beginning [1].

Moreover, in this formalism it is easy to include information on the intersection region (i.e. the beam position and profile) for the primary vertex fit. If

$b \equiv (x_b, y_b, z_b)$ is the mean position of this intersection and $\mathbf{C}_b$ is its covariance matrix, one just needs to add $(V-b)^T\mathbf{C}_b^{-1}(V-b)$ to the χ^2 in eq. (1): this gives an additional term $\mathbf{C}_b^{-1}$ to the matrix $\mathbf{A}$, and $\mathbf{C}_b^{-1}(b-V^0)$ to the $\mathbf{T}$ in eq. (6). This feature allows for some interesting implementations. For instance, calling the vertex fitting routine with only one track is equivalent using a constraint on this track in order to improve its parameters; in the case of two back-to-back collinear tracks, it removes the nondetermination of the vertex position along the track direction.

3. The "perigee" parametrization $q \equiv (\epsilon, z_p, \theta, \phi_p, \rho)$

The aim of this parametrization is to give a precise and simple description of the trajectory in the neighborhood of the expected vertex. The extrapolation of the measured tracks to the vertex region is done once and for all, and the vertex fitting procedure (which may be repeated with different subsets of tracks) uses only a short range propagation.

We assume hereafter the magnetic field to be along beam direction (i.e. the z-axis); we define the "perigee" P to be the point of the closest approach of the trajectory (a helix) to the z-axis. If the origin O is chosen to be around the interaction region, this point is close to the primary vertex and short-lived secondary vertices. Fig. 2 shows the x–y projection, where the position of P is defined through the angle ϕ_p of the trajectory at P, and the algebraic value of $\epsilon \equiv$ OP. By convention, the sign of ϵ is positive if the angle from OP to the direction of the track is $+\pi/2$ (or equivalently if O is at the left side of the trajectory). The description of P in 3-D space is completed by the coordinate z_p, the polar angle θ of the trajectory with respect to z-axis, and its signed curvature ρ ($1/|\rho|$ is the radius of the curvature in the x–y projection, and the sign of ρ is positive if the trajectory is anticlockwise). With these conventions, the trajectory may be parametrized around P as:

$$\begin{cases} x \approx \epsilon \sin\phi_p + L\cos\phi_p - \dfrac{L^2\rho}{2}\sin\phi_p, \\ y \approx -\epsilon\cos\phi_p + L\sin\phi_p + \dfrac{L^2\rho}{2}\cos\phi_p, \\ z \approx z_p + L\cot\theta, \end{cases} \tag{9}$$

where L is a running parameter (i.e. the distance from P along the trajectory in the x–y projection). The terms with L^2 are small if L is small w.r.t. the radius of curvature.

To use this parametrization, i.e. $q = (\epsilon, z_p, \theta, \phi_p, \rho)$, in the vertex fit, we need to express these parameters as functions of the vertex parameters: the coordinates (x_v, y_v, z_v) of vertex V (see fig. 2) and the track parameters $p = (\theta, \phi_v, \rho)$ at V. Note that θ and ρ do not change when going from V to P. If we introduce the quantities $Q \equiv x_v\cos\phi_v + y_v\sin\phi_v$ and $R \equiv y_v\cos\phi_v - x_v\sin\phi_v$, calculations at first order in ρ give:

$$\begin{cases} \epsilon = -R - Q^2\rho/2, \\ z_p = z_v - Q(1-R\rho)\cot\theta, \\ \phi_p = \phi_v - Q\rho. \end{cases} \tag{10}$$

Then the matrices of derivatives (at the lowest order, because the higher precision on them is not needed) in eqs. (2)–(6) have forms as:

$\partial\downarrow/\partial\rightarrow$	x_v	y_v	z_v	θ	ϕ_v	ρ
ϵ	s	$-c$	0	0	Q	$-Q^2/2$
z_p	$-tc$	$-ts$	1	$-Q(1+t^2)$	$-Rt$	QRt
ϕ_p	$-\rho c$	$-\rho s$	0	0	1	$-Q$
	D matrix			E matrix		

(11)

where $c \equiv \cos\phi_v$, $s \equiv \sin\phi_v$, $t \equiv \cot\theta$.

If one wants to fit a secondary vertex far away from the primary vertex (e.g. the long-lived particle decay), the "perigee" parameters may be computed w.r.t. a displaced origin. The precision of the parametrization is acceptable when the vertex is in a range of about 15 ~ 20 mm (see section 6) around the origin chosen to compute the perigee. Thus, for most secondary vertices created by the short-lived particles, they may be fitted by using the reconstructed tracks parametrized with respect to the main origin without re-parametrization with respect to a shifted origin.

Let us also remark that it is very easy to compute

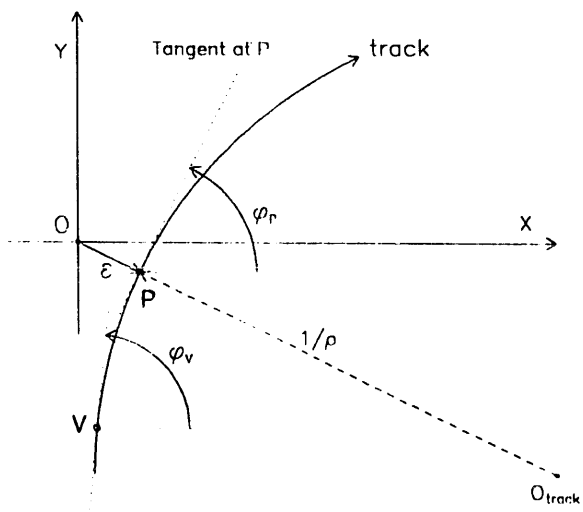

Fig. 2. Definition of the perigee parameters in x–y projection.

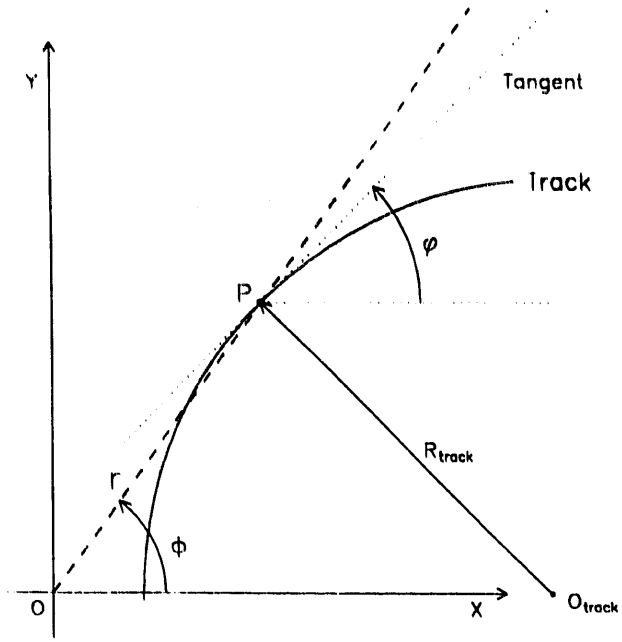

Fig. 3. A parametrization of the 2-D circular track at the reference point P.

the impact parameter D_0 in x–y projection with respect to the fitted vertex, from (ϵ, ϕ_p) and (x_v, y_v):

$$D_0 = \epsilon - (x_v \sin\phi_p - y_v \cos\phi_p). \tag{12}$$

We now give simplified expressions to propagate the error matrix from the reference point (see fig. 1) to the perigee region. If the original track parameters are in cylindrical-polar system, i.e. $(\Phi, z, \theta, \phi, \rho)$ at a fixed values of $r = \sqrt{x^2+y^2}$ (where θ and ϕ are the angles of the tangents to the track, r is the distance between the origin and the reference point, as shown in fig. 3 for the x–y projection), and if ϵ is small with respect to r and r is small with respect to the radius of curvature, we have:

$\partial\downarrow/\partial\rightarrow$	Φ	z	θ	ϕ	ρ
ϵ	r	0	0	$-r$	$-r^2/2$
z_p	0	1	$-r/\cos^2\theta$	0	0
ϕ_p	0	0	0	1	r

, (13)

In this propagation, we assume that the radius r is inside the vacuum beam-pipe, so that no multiple scattering (m/s) occurs. If there are some materials (located at r') between r and vertex, it may be necessary first to propagate the track from r to r' and to include the m/s contribution of the materials into the covariance matrix; then to extrapolate the error matrix to the vertex region by applying eq. (13) with substitution of r by r'.

4. The "simple" vertex fitting

By using the perigee parametrization, a simpler algorithm can be derived to estimate the vertex position without re-fitting the track parameters. In this algorithm, the variation of transverse errors along the track is neglected in the neighbourhood of the perigee, and $p = (\theta, \phi_v, \rho)$ is considered to be constant. Thus the 5×5 covariance matrix **C** (which is the inverse of **W** in eq. (1)) is reduced to its 2×2 submatrix **C'** corresponding to the variables ϵ and z_p, and the 5×3 matrix **D** in eq. (2) to its 2×3 submatrix **D'** which is the derivatives of (ϵ, z_p) w.r.t. (x_v, y_v, z_v), i.e. the first two rows in eq. (11).

Let **W'** be the inverse of the **C'**, and $q' = (\epsilon, z_p)$, then eqs. (2) and (3) can be reduced to

$$F(V^0 + \delta V, p_i^0) = F(V^0, p_i^0) + \mathbf{D}_i'\delta V, \tag{14}$$

$$\chi^2 = \sum_i (\delta q_i' - \mathbf{D}_i'\delta V)^T \mathbf{W}_i' (\delta q_i' - \mathbf{D}_i'\delta V). \tag{15}$$

The eqs. (4) and (7) become

$$\left(\sum_i \mathbf{D}_i'^T \mathbf{W}_i' \mathbf{D}_i'\right)\delta V = \sum_i \mathbf{D}_i'^T \mathbf{W}_i' \delta q_i', \tag{16}$$

$$\delta V = \left(\sum_i \mathbf{D}_i'^T \mathbf{W}_i' \mathbf{D}_i'\right)^{-1}\left(\sum_i \mathbf{D}_i'^T \mathbf{W}_i' \delta q_i'\right) = \mathbf{A}^{-1}\mathbf{T}. \tag{17}$$

It is convenient to choose V^0 (i.e. the first approximation of V) as the origin, hence $\delta V = V$ and $\delta q_i' = q_i'$. Let $x_{pi} = (x_p, y_p, z_p)_i$ to be the coordinates of the perigee P of the track i, then one sees that

$$\left.\begin{aligned} x_{pi} &= \epsilon_i \sin\phi_{pi} \\ y_{pi} &= -\epsilon_i \cos\phi_{pi} \\ z_{pi} &= z_{pi} \end{aligned}\right\}, \quad \text{or } x_{pi} = \mathbf{D}_i'^{-1} q_i'. \tag{18}$$

If we let $\mathbf{w}_i = \mathbf{D}_i'^T \mathbf{W}_i' \mathbf{D}_i'$, then the eqs. (15) and (17) can be rewritten as

$$\chi^2 = \sum_i (x_{pi} - V)^T \mathbf{w}_i (x_{pi} - V), \tag{19}$$

$$V = \left(\sum_i \mathbf{w}_i\right)^{-1}\left(\sum_i \mathbf{w}_i x_{pi}\right). \tag{20}$$

The error matrix on V is simply $(\sum_i \mathbf{w}_i)^{-1}$.

The speed comparison among different algorithms are shown in table 1. This "simple" method also provides a means to quickly reject the tracks which may not belong to the vertex. When solving V by eq. (20), the two factors in the right hand side can be stored; then when calculating the χ^2, by eq. (19), after having obtained V, the contribution to the χ^2 by each individual track i can be examined. If any track (say the track k) would contribute too much to the χ^2, the vertex could be re-estimated by subtracting the $\mathbf{w}_k$ and $\mathbf{w}_k x_{pk}$

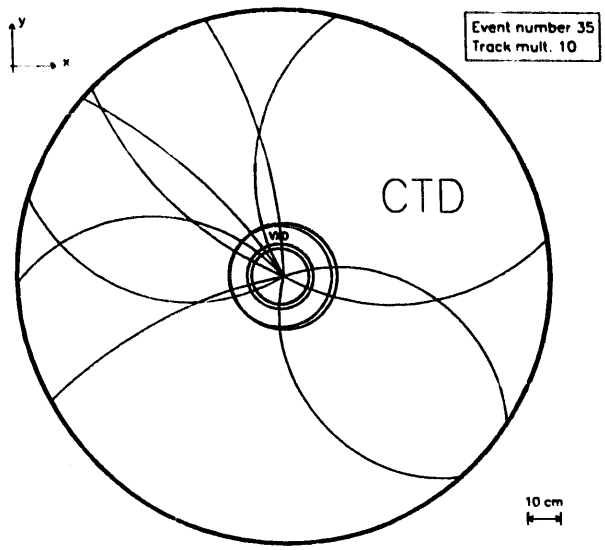

Fig. 4. A typical event after vertex fit in $x-y$ projection.

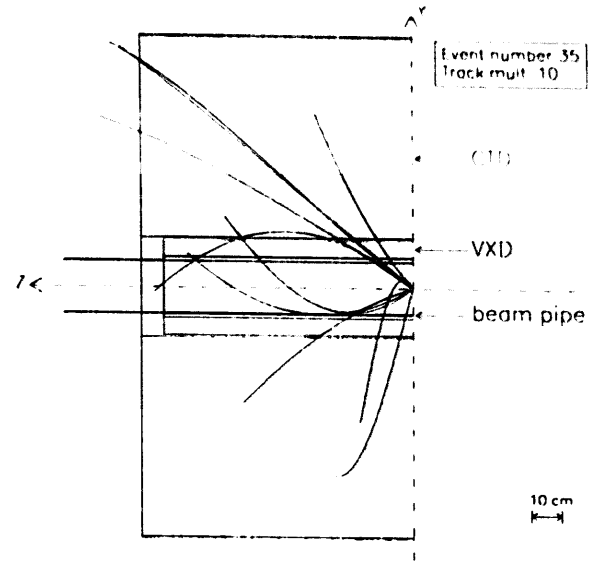

Fig. 6. The same event in $y-z$ projection.

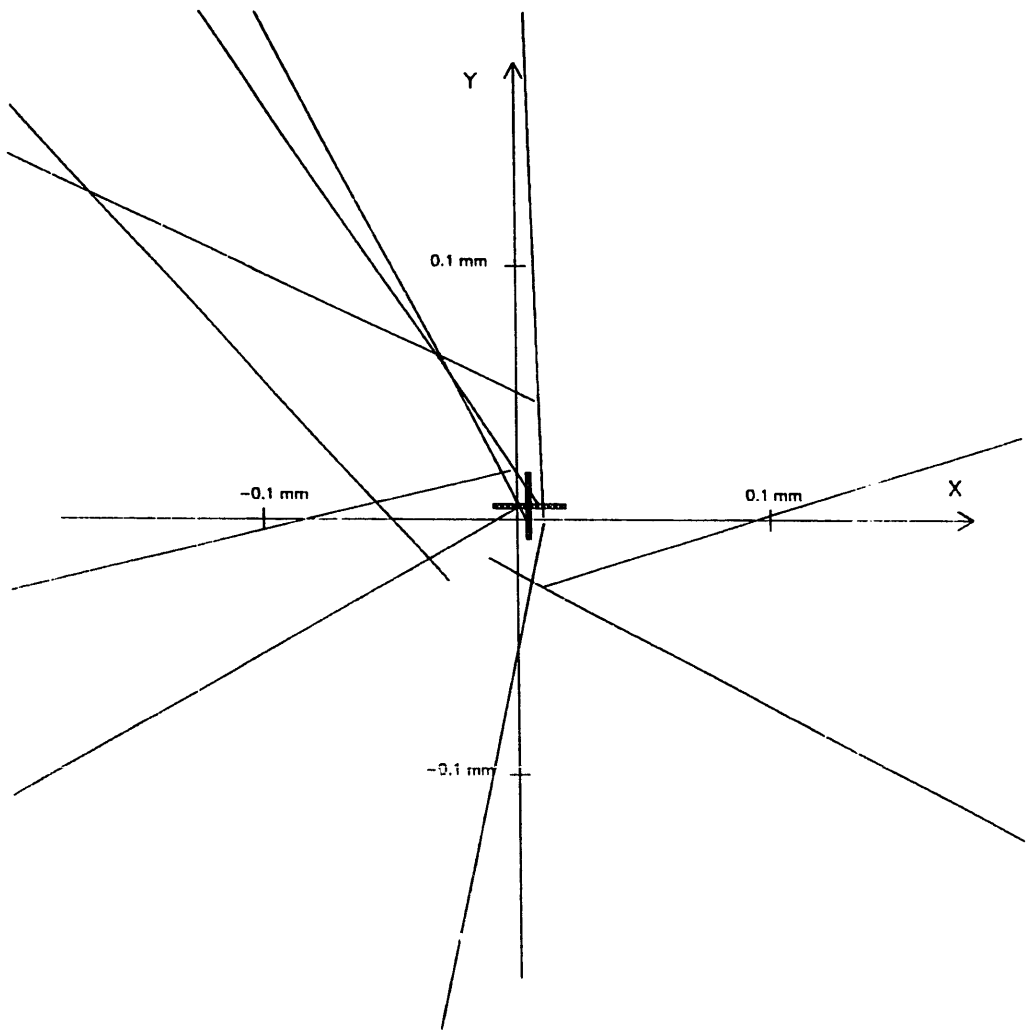

Fig. 5. Zoom view of the fig. 4 in the vertex area. The shadowed cross indicates the location of the fitted vertex and its error.

from the two stored factors of eq. (20), without a complete re-calculation from the beginning.

In the same way as mentioned at the end of section 2, the information of beam position can be introduced by adding $\mathbf{C}_b^{-1}$ to $\Sigma_i \mathbf{w}_i$ and $\mathbf{C}_b^{-1}\boldsymbol{b}$ to $(\Sigma_i \mathbf{w}_i x_{pi})$ in eq. (20).

5. Test procedures

The test is undertaken in the environment of ZEUS detector at DESY. To simplify the test, only two major components (i.e. VXD and CTD) of the tracking detectors are involved. Both components have the cylindrical configurations with axes parallel to the magnetic field (i.e. z-direction). The outline of these components can be seen from figs. 4 and 6. The VXD's dimensions are +975 mm and −615 mm in z. The sensitive region is from $r = 106.5$ mm to 139.5 mm. VXD does not measure z. Its resolution on r-ϕ (i.e. $\sigma_{r\text{-}\phi}$) is 35 ~ 160 μm depending on digitization distance. The sensitive region of CTD ranges from $r = 182$ mm to 794 mm, the middle point is at $r = 488$ mm. There are nine superlayers (each superlayer consists of 8 sense wires) inside CTD, four of them are stereo and the others are axial. The $\sigma_{r\text{-}\phi}$ of CTD is about 110 μm. The stereo layers provide σ_z of about 1 mm.

This Monte Carlo test can be divided into four steps as discussed below.

5.1. Event generation

For simplicity, only events with fixed track multiplicity are generated, this is also convenient for the timing (i.e. the speed) test. The charged track multiplicity has been chosen as 5, 10, 20 etc. The ϕ distribution of tracks is random and the event vertex is located at the origin. The p_T and p_{tot} distribution of tracks (fig. 7) are similar to the neutral current events at HERA energy (i.e. 30 Gev of electron + 800 Gev of proton) with $Q^2 \geqslant 10$ GeV^2 as used in ref. [3]. The tracking is undertaken inside CTD and VXD only, and in forward region only for further simplicity, so the polar angle θ is restricted in between 37.1° and 90°.

The multiple scattering (m/s) of materials (e.g. the beampipe, inner and outer walls of VXD, inner wall of CTD) can be turned on or off in event generation. The majority of the results shown in next section include m/s.

In order to see how the displacement of the origin affects the vertex fitting quality, a same set of events (with track multiplicity of 10) is generated with the vertex shifted from the origin (0, 0) at a fixed distance (10 ~ 30 mm) along x-axis. For the test that includes the beam profile information, the position (with the mean at (0, 0)) of the event vertices are smeared according to the HERA designed profile (i.e. $\sigma(x_b) \sim 270$ μm, $\sigma(y_b) \sim 80$ μm). The matrix $\mathbf{C}_b^{-1}$ is set to diag$(1/\sigma(x_b)^2, 1/\sigma(y_b)^2, 0)$.

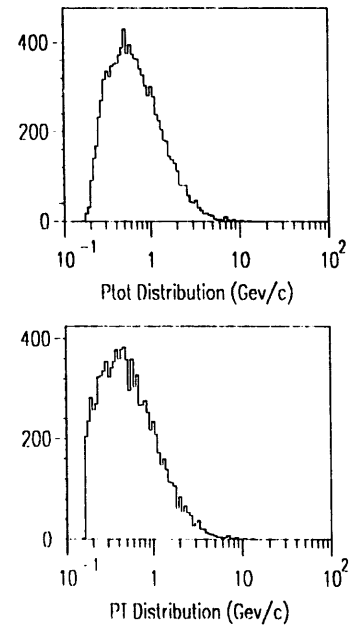

Fig. 7. p_{tot} and p_T distributions of the simulated tracks.

5.2. Track reconstruction

Inside VXD, the approach to get the smeared digitization points is the same as in ref. [3]. The tracks are reconstructed by the Kalman filtering method described in ref. [3] except the track finding efficiency has been assumed as 100%.

Inside CTD, the reconstructed tracks are simulated through the following steps.

(1) The generated tracks are extrapolated to the middle point of CTD, i.e. $r = 488$ mm.

(2) An approximated error matrix at the middle point is constructed by the Gluckstern's technics [4], see appendix A for details.

(3) The track parameters are smeared according to the above error matrix and the CTD's resolutions $\sigma_{r\text{-}\phi}$ and σ_z. The smearing with a correlation term in the error matrix is described in appendix A as well

(4) The error matrix and the smeared parameters are extrapolated backwards to the position of the last point of the VXD track, and matched with VXD track there by normal matching formalism (e.g. in refs. [3,5]).

5.3. Perigee parametrization

The combined track is perigee-parametrized (according to the definitions in section 3) w.r.t. the origin (0, 0) as the first approximation of the vertex position, even for the additional tests (i.e. the shift of the vertex position and the inclusion of the beam profile) mentioned at the end of subsection 5.1. The

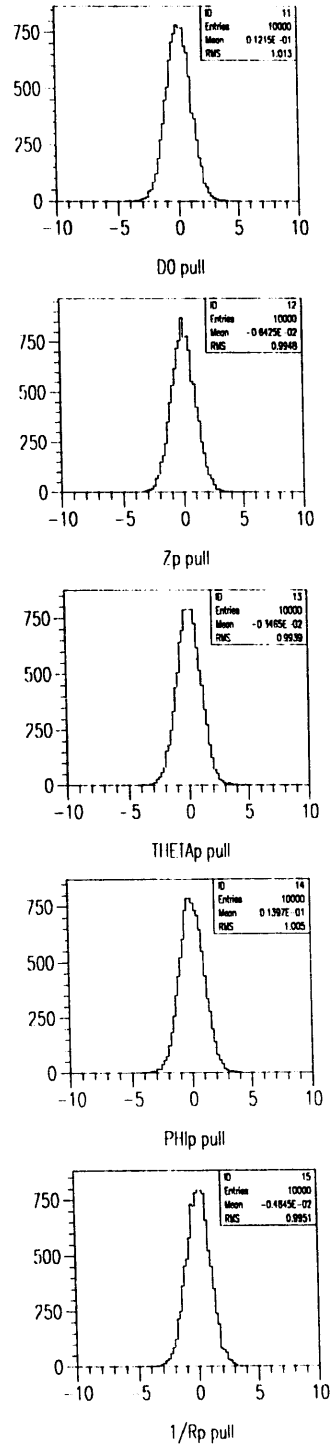

Fig. 8. Pull plots for the reconstructed tracks before vertex fitting.

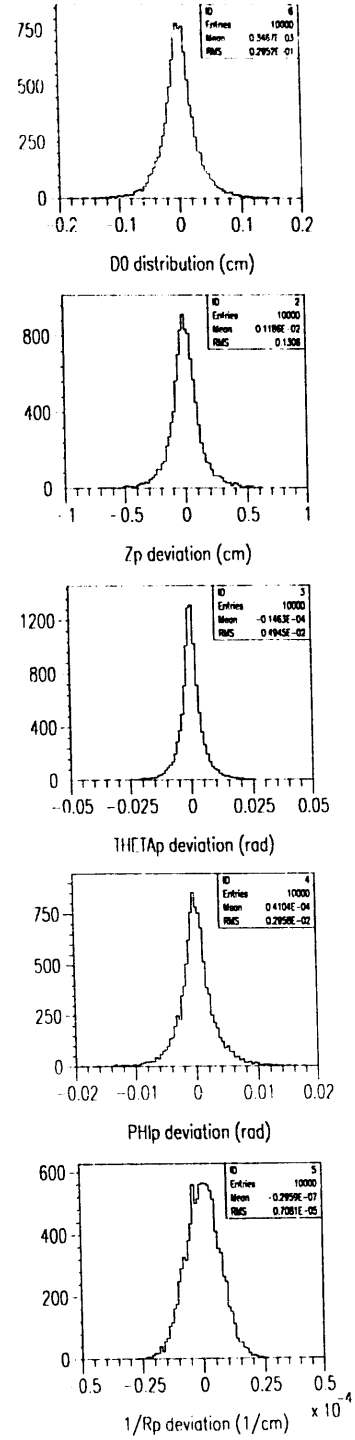

Fig. 9. Deviation plots in correspondence with the fig. 8.

parametrization procedure is coded as an "Interface" appearing in table 1.

5.4. Vertex fitting

The inputs of the "simple" vertex fitting procedure are the number of tracks, their perigee parameters and associated covariance matrices. The outputs are the fitted vertex position and its covariance matrix, χ^2 of the fit and the contribution of each track (denoted as χ_i^2).

The inputs of the "full" fit are the same as the "simple" one plus a first approximate vertex position which can be the output of a "simple" fit or just the

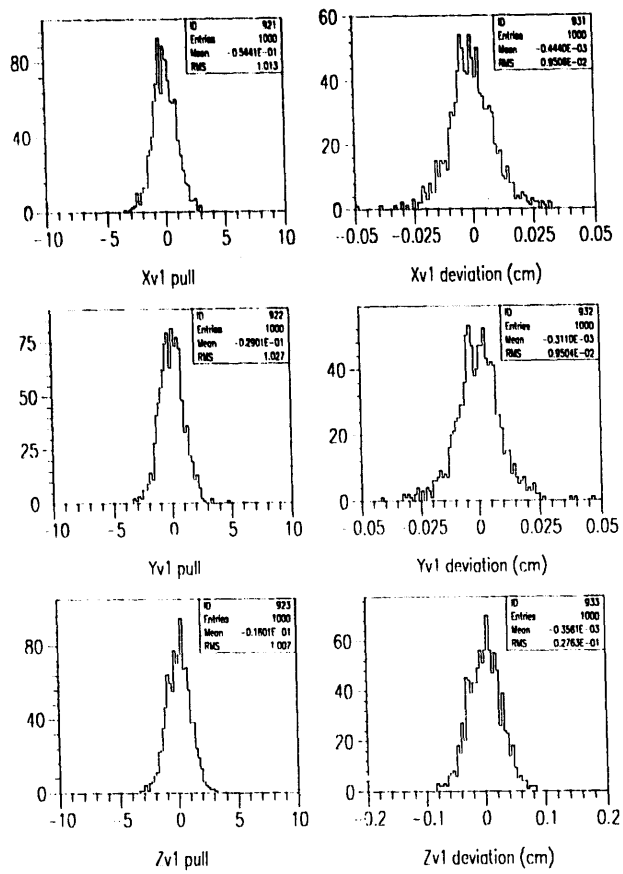

Fig. 10. Vertex fitting quality of the "simple" fitting method (i.e. without re-fitting the track).

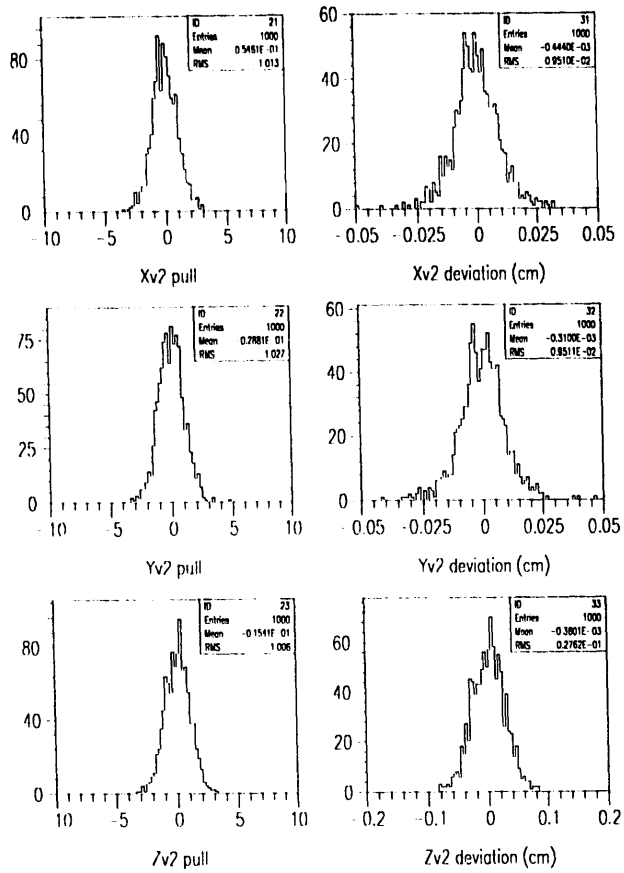

Fig. 11. Vertex fitting quality of the "full" fitting method (i.e. with re-fitting the track).

origin used in the perigee parametrization. The outputs are the same as the "simple" fit, but including also the re-fitted track parameters and their covariance matrices.

6. Test results

A vertex fit is meaningful only when the input tracks are reconstructed correctly. The pull plots (fig. 8) show the quality of the reconstructed tracks before the vertex fitting. The plots for all parameters are of the expected shape (i.e. the mean values are at 0 and with r.m.s. width of 1). From the deviation plots (fig. 9), one can get a quantitative idea about absolute value of the parameters when turning on multiple scattering.

Fig. 10 are the pull and deviation plots for the fitted vertex position by the "simple" fitting method. For the "full" fitting method, fig. 11 shows the quality of the fitted vertex; fig. 12 shows the quality of the re-fitted tracks. One can see a clear improvement by comparing them with fig. 9, especially in ϕ. Another check on the quality of the vertex fitting is the distribution of the probability of χ^2 (fig. 13). Its flatness indicates the fitting is creditable.

Fig. 14 shows the equivalence of the "simple" and "full" fits as far as the vertex position is concerned. Figs. 4 and 5 are the graphic display for a typical event (without turning on m/s) in the x–y projection; fig. 6 is for the y–z projection.

The result of the test for the origin displacement is shown in fig. 15. It can be seen that, for the vertices apart from the origin up to about 15 mm, the perigee parametrization w.r.t. the origin can assure a correct fit. From fig. 16, one sees the improvement on the precision of the vertex and track fitting by including the information of the beam profile, especially when track multiplicity is low.

The speed is compared in table 1. For the method with large matrix inversion, only CPU time for the inversion (i.e. without including any other operation) is monitored. For the fast methods, the column "Interface" indicates the time spent in the perigee parametrization. The absolute value of CPU time is in VAX8800 computer unit.

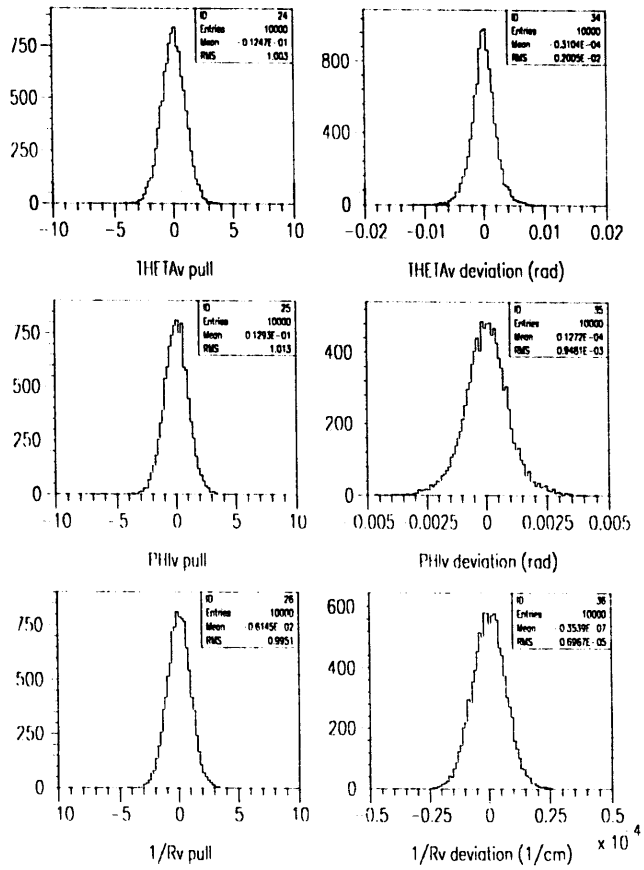

Fig. 12. Track re-fitting quality by the "full" fitting method.

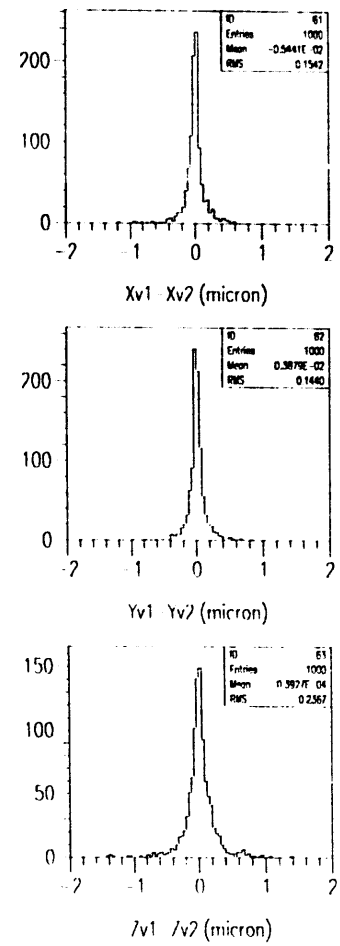

Fig. 14. Difference between vertex positions obtained by the "simple" and "full" fits.

7. Conclusions

The fast vertex fitting algorithm with the perigee parametrization performs very well in our test. The following features of the method may be concluded:

(1) It is indeed fast, especially for the primary vertex with high track multiplicity.

(2) The fitting quality is quite good and can be maintained within a range of 15 mm without need of re-parametrization of tracks. Also the easy introduction of the beam profile can improve the fitting quality for the primary vertex.

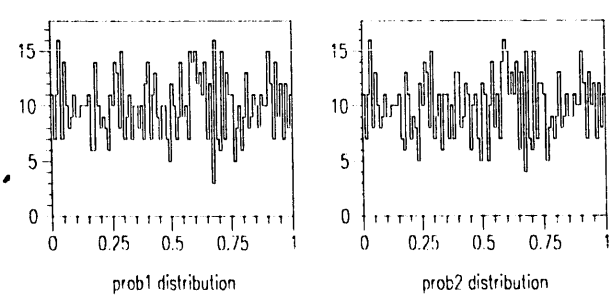

Fig. 13. Distributions of probability of χ^2 for the "simple" (1) and "full" (2) fitting methods.

(3) From the fitting point of view, the method can be applied to either primary or secondary vertices as long as the track bundling (to a potential vertex) has been done.

(4) It allows to perform the track addition to or the subtraction from a fitted vertex without fully re-fitting. This may be used to build up a vertex finding strategy that separates the primary and secondary vertices.

Therefore this algorithm hopefully provides an efficient way for vertex reconstruction.

Acknowledgements

We wish to thank Dr. O. Couet for the help on the graphic display by PAW; Dr. C. Williams for his generous help in correcting the proofs; and also Ms. K. Kang for the help on the word processing. The work has been supported by the LAA group at CERN, led by Prof. A. Zichichi, and INFN/Frascati of Italy, we would like to express our sincere gratitude to them.

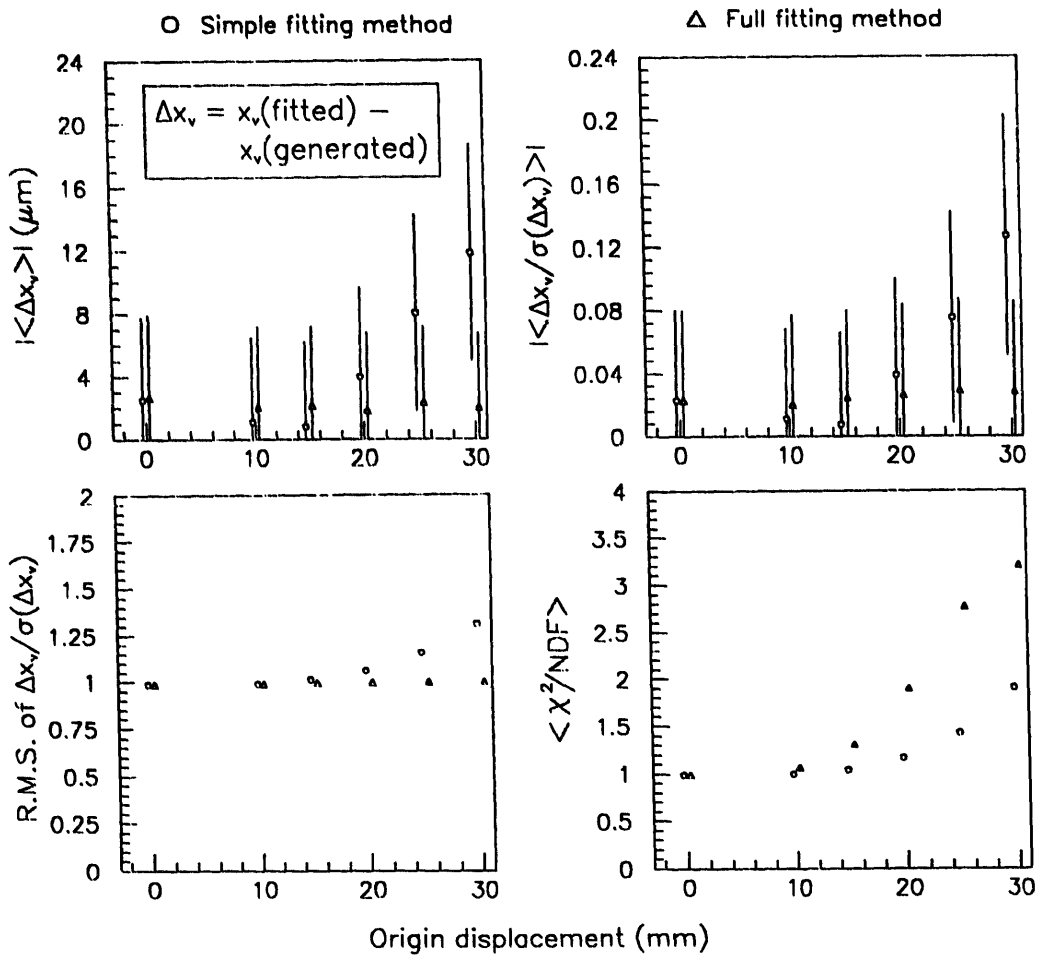

Fig. 15. Dependence of the vertex fitting qualities on the origin displacement.

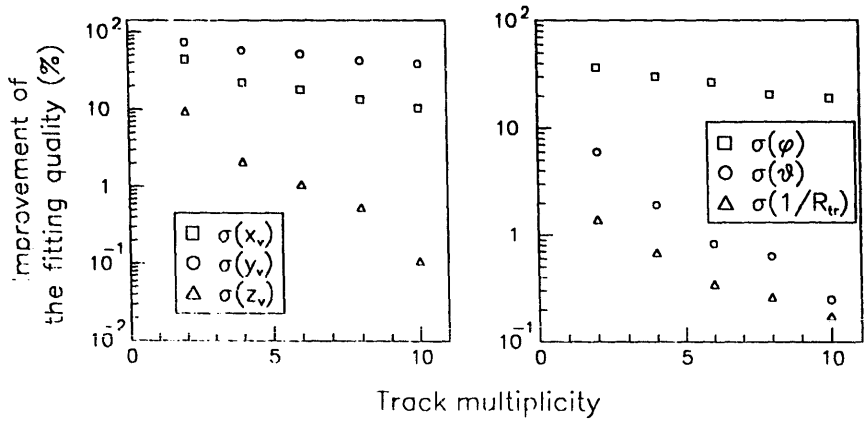

Fig. 16. Improvement on the fitting quality by the inclusion of the beam profile. The improvement is defined as: for example,

$$\text{Improvement in } \sigma(x_v) \equiv \frac{\sigma(x_v)(\text{no-beam}) - \sigma(x_v)(\text{beam})}{\sigma(x_v)(\text{no-beam})}.$$

Note: an improvement of 50% is equivalent to the factor 2 reduction in σ when the beam profile is included; an improvement of 75% is equivalent to the factor 4 reduction; etc.

Table 1
Timing comparison

Track multiplicity of vertex	Matrix inversion		Fast methods in this article		
	Dimension of matrix	CPU time [ms]	"Simple" fit [ms]	"Full" fit [ms]	Interface [ms]
5	23	13.5	1.0	2.3	3.7
10	43	85	1.4	5.2	8.8
20	83	670	3.1	11.5	13.7

Appendix A

Estimation of the error matrix at the middle point of a uniform detector

Suppose a 2-D detector is divided into $2N$ equi-spaced sensitive region. The position of the region i is at $x_i = ib$, where $b = L/N$, $i = -N \sim N$, the L is the half-length of the detector's sensitive area. The resolution at each position is σ. A 2-D circular track can be parametrized with

$$p_{r\text{-}\phi} = \left(r\Phi,\ \phi,\ \frac{1}{2R_{\mathrm{tr}}} \right),$$

see fig. 3. Its symmetric weight matrix can be constructed as

$$\mathbf{W} = \frac{1}{\sigma^2} \begin{pmatrix} \sum_{-N}^{N} 1 & \sum_{-N}^{N} x_i & \sum_{-N}^{N} x_i^2 \\ & \sum_{-N}^{N} x_i^2 & \sum_{-N}^{N} x_i^3 \\ & & \sum_{-N}^{N} x_i^4 \end{pmatrix}$$

$$\xrightarrow[N\to\infty]{} \frac{1}{\sigma^2} \begin{pmatrix} 2N & 0 & \frac{2}{3}N^3b^2 \\ & \frac{2}{3}N^3b^2 & 0 \\ & & \frac{2}{5}N^5b^4 \end{pmatrix}, \qquad \text{(A.1)}$$

where $\Sigma_{-N}^{N} x_i$ and $\Sigma_{-N}^{N} x_i^3$ vanish due to the anti-symmetric feature of the terms. The covariance matrix COV is the inverse of $\mathbf{W}$, i.e.

$$\mathrm{COV}(p_{r\text{-}\phi}) = \mathbf{W}^{-1} = \sigma_{r\text{-}\phi}^2 \begin{pmatrix} \frac{9}{8N} & 0 & \frac{-15}{8NL^2} \\ & \frac{3}{2NL^2} & 0 \\ & & \frac{45}{8NL^4} \end{pmatrix}. \qquad \text{(A.2)}$$

In the ZEUS CTD case, $N = 36$, $L = 323$ mm, $\sigma_{r\text{-}\phi} = 110$ μm.

For a 3-D track helixed along the magnetic field (i.e. the z-axis) with a large radius of curvature, the r-z projection is close to a straight line. The COV for the parameters $p_{r\text{-}z} = (z, \cot\theta)$ can be constructed similarly:

$$\mathrm{COV}(p_{r\text{-}z}) = \sigma_z^2 \begin{pmatrix} \frac{1}{2N_z} & 0 \\ & \frac{3}{2N_zL^2} \end{pmatrix}, \qquad \text{(A.3)}$$

where $N_z = 16$ (since there are only 32 stereo wire layers in CTD), $\sigma_z = 1$ mm.

Since there is a non-diagonal term in eq. (A.2), we can no longer smear the parameters simply only according to the diagonal terms. In other words, the correlated parameters can no longer be smeared independently. To take the correlation between two parameters (say x and y) into account, we may form another pair of parameters u and v as

$$u = \frac{1}{2}\left(\frac{x}{\sigma_x} + \frac{y}{\sigma_y} \right) \quad \text{and} \quad v = \frac{1}{2}\left(\frac{x}{\sigma_x} - \frac{y}{\sigma_y} \right), \qquad \text{(A.4)}$$

so that the u and v will be independent each other.

The errors on u thus are

$$\sigma_u^2 = \tfrac{1}{4}\sigma \text{ of } \left(\frac{x}{\sigma_x} + \frac{y}{\sigma_y} \right)^2 = \frac{1}{2}\left(1 + \frac{\sigma_{xy}}{\sigma_x\sigma_y} \right), \qquad \text{(A.5)}$$

where $\sigma_{xy}/(\sigma_x\sigma_y)$ is the correlation term in the normalized covariance matrix, e.g. it is $-\sqrt{5}/3$ in the normalized COV

$$\left(\text{i.e.} \begin{pmatrix} 1 & 0 & -\frac{\sqrt{5}}{3} \\ & 1 & 0 \\ & & 1 \end{pmatrix} \right)$$

in eq. (A.2). If we let x and y being the two correlated parameters in $p_{r\text{-}\phi}$ then

$$\sigma_u^2 = \frac{1}{2}\left(1 - \frac{\sqrt{5}}{3} \right) = 0.127322,$$

and

$$\sigma_v^2 = \frac{1}{2}\left(1 + \frac{\sqrt{5}}{3}\right) = 0.872678. \qquad \text{(A.6)}$$

Therefore the smearings of x and y can be done as

$$x = (\mathrm{RAN1}\sigma_u + \mathrm{RAN2}\sigma_v)\sigma_x + x_0$$
$$= (\mathrm{RAN1} \times 0.3568221 + \mathrm{RAN2} \times 0.9341724)\sigma_x + x_0$$
$$y = (\mathrm{RAN1}\sigma_u - \mathrm{RAN2}\sigma_v)\sigma_y + y_0$$
$$= (\mathrm{RAN1} \times 0.3568221 - \mathrm{RAN2} \times 0.9341724)\sigma_y + y_0,$$

where RAN1 and RAN2 are independent Gaussian distributed random numbers.

References

[1] P. Billoir, R. Frühwirth and M. Regler, Nucl. Instr. and Meth. A241 (1985) 115.
[2] D.H. Saxon, Nucl. Instr. and Meth. A234 (1985) 258.
[3] P. Billoir and S. Qian, Nucl. Instr. and Meth. A294 (1990) 219.
[4] R.L. Gluckstern, Nucl. Instr. and Meth. 24 (1963) 381.
[5] S. Qian, ZEUS Note 90-064 (1990).