Paul Avery
CBX 98–37
June 9, 1998
Apr. 17, 1999 (rev.)

Applied Fitting Theory VI

Formulas for Kinematic Fitting

I Introduction

I intend for this note and the one following it to serve as mathematical references, with examples, for kinematic fitting as implemented in the **KWFIT** [1] library. Until the recent advent of silicon tracking, CLEO made little use of kinematic fitting for reasons ranging from ignorance of what it can do, lack of knowledge of existing software tools and poorly understood charged tracking errors. However, Kalman fitted tracks and silicon tracking provide better tracking errors, and better *understood* errors, both of which are important for kinematic fitting to aid analyses. The recent availability of high quality position information has thrust kinematic fitting (as implemented in the **KWFIT** and **VCFIT** [2] libraries) into a far more central role in CLEO analysis, a role which I believe requires extensive documentation to enable physicists to understand the power and limitations of kinematic fitting. This note serves as that documentation. It will be constantly updated in my fitting home page* as I accumulate new information and examples. Most of the algorithms discussed here are implemented in **KWFIT**.

Before I begin, let me define what we are talking about. Kinematic fitting is a mathematical procedure in which one uses the physical laws governing a particle interaction or decay to improve the measurements describing the process. For example, the fact that the three particles coming from a $D^+ \to K^-\pi^+\pi^+$ decay must come from a common space point can be used to improve the momentum vectors of the daughter particles, thus improving the mass resolution of the D^+. Explicit formulations of this constraint and others based on mass, energy, momentum, etc. are described later.

The raw materials for kinematic fitting are charged and neutral tracks obtained, respectively, from measurements made in central tracking chambers and photon shower detectors. It is assumed that for each track a set of parameters and their covariance matrix is available. [3,4] The parameters and their covariance matrix contain all the necessary information to apply constraints; one does not need to worry about the detailed hit information.

For a somewhat more complicated example, consider the decay sequence

* www.phys.ufl.edu/~avery/fitting.html

$$\begin{aligned}\overline{B}^0 &\to D^{*+}\pi^- \\ &\quad D^{*+} \to D^0\pi^+ \\ &\quad\quad D^0 \to K^-\pi^+\pi^0\end{aligned} \tag{1}$$

Assuming that the neutral pion has been previously fit, there are several constraints which can be applied: (1) the $K^-\pi^+\pi^0$ mass can be forced to be equal to the D^0 mass (1 constraint); (2) the K^- and π^+ from D^0 decay intersect at a single space point (2*2 – 3 = 1 constraint); (3) the $D^0\pi^+$ mass can be constrained to the D^{*+} mass (1 constraint); (4) the D^0, the soft π^+ from the D^{*+} decay and the π^- from $\overline{B}^0$ decay should intersect at the same space point (3*2 – 3 = 3 constraints); and (5) the energies of the final state particles must add up to the beam energy (1 constraint). When the tracks are refit with these 7 constraints using the general algorithm discussed in the Appendix, their parameters will satisfy these constraints and the $D^{*+}\pi^-$ mass will be narrower than before.

II Track representation and notation

One of the rewards of writing notes on fitting theory is the number of people who stop me in the street to shake my hand, telling me breathless stories of how kinematic fitting changed their lives. Whenever I am accosted in this way, the question is invariably raised about why I represent tracks using the 7-parameter *W* representation, defined as $\alpha_W = \left(p_x, p_y, p_z, E, x, y, z\right)$, i.e., a 4-momentum and a point where the 4-momentum is evaluated. Why, my petitioners persist, do I not use the standard 5-parameter track representation used in track fitting? And aren’t seven parameters redundant?

Well, these are important questions, so let me try to explain my reasoning. I designate the 5-parameter representation the *C* format to distinguish it from the *W* format used here. The *C* format consists of three variables specifying the particle momentum and two variables defining its location in space (a more precise definition can be found in Appendix I). Most importantly, it does not specify the location in space of a particle production or decay; that would require a sixth parameter. The particle's initial position is defined to be at the point of closest approach to the reference point, irrespective of its actual origin.

For kinematic fitting it is important to choose a track representation which uses physically meaningful quantities and is complete. The first reason why the *C* format is insufficient for kinematic fitting is that it cannot handle neutral tracks because the curvature is 0 regardless of momentum. One could of course change the representation so that the curvature becomes the inverse momentum but that would introduce a different format. Secondly, the *W* format is much simpler to transport in a magnetic field. See Appendix II for details. Finally, the *C* format does not have enough information to represent general decays of particles. In kinematic fitting, the energy varies independently of the momentum because the mass is in general not constrained.

Also, as discussed in the preceding paragraph, the position where a particle decays requires three coordinates, one more than specified by the *C* format.

Appendix I describes a procedure for converting the 5 charged track parameters and their covariance matrix to the *W* format.

III General algorithm for kinematic fitting

The fitting technique is straightforward and is based on the well-known Lagrange multiplier method [3]. It is assumed that the constraint equations can be linearized and summarized in two matrices, which I label **D** and **d**; specific expressions for **D** and **d** are given in Section V for constraints most commonly encountered in HEP analyses.

Let $\boldsymbol{\alpha}$ represent the parameters for a set of *n* tracks (a total of 7*n* parameters). It has the form of a column vector

$$\boldsymbol{\alpha} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{pmatrix} \tag{2}$$

Initially the track parameters have the unconstrained values $\boldsymbol{\alpha}_0$, obtained from a track fit, for example. The *r* functions describing the constraints can be written generally as $\mathbf{H}(\boldsymbol{\alpha}) = \mathbf{0}$, where $\mathbf{H} \equiv (H_1, H_2, \ldots H_r)$. Expanding around a convenient point $\boldsymbol{\alpha}_A$ yields the linearized equations

$$\mathbf{0} = [\partial \mathbf{H}(\boldsymbol{\alpha}_A) / \partial \boldsymbol{\alpha}](\boldsymbol{\alpha} - \boldsymbol{\alpha}_A) + \mathbf{H}(\boldsymbol{\alpha}_A) \equiv \mathbf{D}\delta\boldsymbol{\alpha} + \mathbf{d}, \tag{3}$$

where $\delta\boldsymbol{\alpha} = \boldsymbol{\alpha} - \boldsymbol{\alpha}_A$. Thus we see that

$$\mathbf{D} = \begin{pmatrix} \frac{\partial H_1}{\partial \alpha_1} & \frac{\partial H_1}{\partial \alpha_2} & \cdots & \frac{\partial H_1}{\partial \alpha_n} \\ \frac{\partial H_2}{\partial \alpha_1} & \frac{\partial H_2}{\partial \alpha_2} & \cdots & \frac{\partial H_2}{\partial \alpha_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial H_r}{\partial \alpha_1} & \frac{\partial H_r}{\partial \alpha_2} & \cdots & \frac{\partial H_r}{\partial \alpha_n} \end{pmatrix} \qquad \mathbf{d} = \begin{pmatrix} H_1(\boldsymbol{\alpha}_A) \\ H_2(\boldsymbol{\alpha}_A) \\ \vdots \\ H_r(\boldsymbol{\alpha}_A) \end{pmatrix}$$

or $D_{ij} = \partial H_i / \partial \alpha_j$ and $d_i = H_i(\boldsymbol{\alpha}_A)$. The constraints are incorporated using the method of Lagrange multipliers [3] in which the χ^2 is written as a sum of two terms, e.g.

$$\chi^2 = (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0)^T \mathbf{V}_{\boldsymbol{\alpha}_0}^{-1} (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0) + 2\boldsymbol{\lambda}^T (\mathbf{D}\delta\boldsymbol{\alpha} + \mathbf{d}), \tag{4}$$

where $\boldsymbol{\lambda}$ is a vector of *r* unknowns, the Lagrange multipliers. Minimizing the χ^2 with respect to $\boldsymbol{\alpha}$ and $\boldsymbol{\lambda}$ yields two vector equations which can be solved for the parameters $\boldsymbol{\alpha}$ and their covariance matrix:

$$\begin{aligned}\mathbf{V}_{\alpha 0}^{-1}(\boldsymbol{\alpha}-\boldsymbol{\alpha}_0)+\mathbf{D}^T\boldsymbol{\lambda}=0\\ \mathbf{D}\delta\boldsymbol{\alpha}+\mathbf{d}=0\end{aligned} \tag{5}$$

The latter equation demonstrates clearly that the solution satisfies the constraints. The solution can be written [3]

$$\begin{aligned}\boldsymbol{\alpha}&=\boldsymbol{\alpha}_0-\mathbf{V}_{\alpha_0}\mathbf{D}^T\boldsymbol{\lambda}\\ \boldsymbol{\lambda}&=\mathbf{V}_D(\mathbf{D}\delta\boldsymbol{\alpha}_0+\mathbf{d})\\ \mathbf{V}_D&=\left(\mathbf{D}\mathbf{V}_{\alpha_0}\mathbf{D}^T\right)^{-1}\\ \mathbf{V}_\alpha&=\mathbf{V}_{\alpha_0}-\mathbf{V}_{\alpha_0}\mathbf{D}^T\mathbf{V}_D\mathbf{D}\mathbf{V}_{\alpha_0}\\ \chi^2&=\boldsymbol{\lambda}^T\mathbf{V}_D^{-1}\boldsymbol{\lambda}\\ &=\boldsymbol{\lambda}^T(\mathbf{D}\delta\boldsymbol{\alpha}_0+\mathbf{d})\end{aligned} \tag{6}$$

where $\delta\boldsymbol{\alpha}_0=\boldsymbol{\alpha}_0-\boldsymbol{\alpha}_A$. It can be shown that the diagonal elements of the $\boldsymbol{\alpha}$ covariance matrix are reduced in size, as expected by intuition.

Several things should be noted about the above solution. First, only a single matrix must be inverted, the $r\times r$ matrix $\mathbf{V}_D$. The changes to $\boldsymbol{\alpha}$ caused by the constraints are propagated by matrix multiplication. Second, the χ^2 is a sum of *r* distinct terms, one per constraint. However, identifying each of these terms with a particular constraint is only possible in a loose sense since the contribution of each constraint is correlated with all others through $\mathbf{V}_D$.

Finally, it is useful to compute how far the parameters have to move to satisfy a particular constraint *j*. The initial "distance from satisfaction" can be characterized by the quantity $(\mathbf{D}\delta\boldsymbol{\alpha}_0+\mathbf{d})_j$ and the number of standard deviations away from satisfying the constraint is easily calculated to be

$$\sigma_j=\frac{D_{ji}\delta\alpha_{0i}+d_j}{\sqrt{\left(\mathbf{V}_D^{-1}\right)_{jj}}}$$

This information can be used to provide criteria for rejecting background in addition to the overall χ^2.

IV Example: Back to back constraint

Let's apply the algorithm to a simple case where we want two particles to be back to back. This is equivalent to writing the following constraint equations (this is similar to the constraint discussed in Section V.4):

$$\begin{aligned} p_{x1} + p_{x2} &= 0 \\ p_{y1} + p_{y2} &= 0 \\ p_{z1} + p_{z2} &= 0 \end{aligned} \tag{7}$$

Initially, the track parameters have the values $\boldsymbol{\alpha}^0 = \begin{pmatrix} \alpha_1^0 \\ \alpha_2^0 \end{pmatrix}$, where

$$\boldsymbol{\alpha}_1^0 = \begin{pmatrix} p_{x1}^0 \\ p_{y1}^0 \\ p_{z1}^0 \\ E_1^0 \\ x_1^0 \\ y_1^0 \\ z_1^0 \end{pmatrix} \qquad \boldsymbol{\alpha}_2^0 = \begin{pmatrix} p_{x2}^0 \\ p_{y2}^0 \\ p_{z2}^0 \\ E_2^0 \\ x_2^0 \\ y_2^0 \\ z_2^0 \end{pmatrix} \tag{8}$$

and the initial 7 × 7 track covariance matrices are denoted $\mathbf{V}_{1_0}$ and $\mathbf{V}_{2_0}$. The parameters can be expanded about these values (i.e., $\boldsymbol{\alpha}_A = \boldsymbol{\alpha}^0$) giving for **D** and **d**:

$$\begin{aligned} \mathbf{D} &= \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \\ \mathbf{d} &= \begin{pmatrix} p_{x1}^0 + p_{x2}^0 \\ p_{y1}^0 + p_{y2}^0 \\ p_{z1}^0 + p_{z2}^0 \end{pmatrix} \end{aligned} \tag{9}$$

where I have ignored the bending of the track as a function of position.

A clever reader might notice that I have not included the change in energy caused by shifts in momentum for tracks with fixed mass (i.e., those found from track fitting or shower finding). However, this constraint is already "built in", in the sense that the initial track covariance matrix already includes the correlation of energy with momentum. Additional constraints will move the track parameters around in such a way as to preserve the mass*.

The matrix $\mathbf{V}_D$ is given by

$$\mathbf{V}_D = \left(\mathbf{D}\mathbf{V}_{\alpha_0}\mathbf{D}^T\right)^{-1} = \begin{pmatrix} \left(\mathbf{V}_{1_0}\right)_{11} + \left(\mathbf{V}_{2_0}\right)_{11} & \left(\mathbf{V}_{1_0}\right)_{21} + \left(\mathbf{V}_{2_0}\right)_{21} & \left(\mathbf{V}_{1_0}\right)_{31} + \left(\mathbf{V}_{2_0}\right)_{31} \\ \left(\mathbf{V}_{1_0}\right)_{12} + \left(\mathbf{V}_{2_0}\right)_{12} & \left(\mathbf{V}_{1_0}\right)_{22} + \left(\mathbf{V}_{2_0}\right)_{22} & \left(\mathbf{V}_{1_0}\right)_{32} + \left(\mathbf{V}_{2_0}\right)_{32} \\ \left(\mathbf{V}_{1_0}\right)_{13} + \left(\mathbf{V}_{2_0}\right)_{13} & \left(\mathbf{V}_{1_0}\right)_{23} + \left(\mathbf{V}_{2_0}\right)_{23} & \left(\mathbf{V}_{1_0}\right)_{33} + \left(\mathbf{V}_{2_0}\right)_{33} \end{pmatrix}^{-1} \tag{10}$$

* It's best to have a couple of beers before thinking seriously about this.

The Lagrange multipliers can be calculated from $\boldsymbol{\lambda} = \mathbf{V}_D\left(\mathbf{D}\delta\boldsymbol{\alpha}_0 + \mathbf{d}\right)$, yielding the expressions

$$\begin{aligned}
\lambda_1 &= (\mathbf{V}_D)_{11}\left(p^0_{x1} + p^0_{x2}\right) + (\mathbf{V}_D)_{12}\left(p^0_{y1} + p^0_{y2}\right) + (\mathbf{V}_D)_{13}\left(p^0_{z1} + p^0_{z2}\right) \\
\lambda_2 &= (\mathbf{V}_D)_{21}\left(p^0_{x1} + p^0_{x2}\right) + (\mathbf{V}_D)_{22}\left(p^0_{y1} + p^0_{y2}\right) + (\mathbf{V}_D)_{23}\left(p^0_{z1} + p^0_{z2}\right) \\
\lambda_3 &= (\mathbf{V}_D)_{31}\left(p^0_{x1} + p^0_{x2}\right) + (\mathbf{V}_D)_{32}\left(p^0_{y1} + p^0_{y2}\right) + (\mathbf{V}_D)_{33}\left(p^0_{z1} + p^0_{z2}\right)
\end{aligned} \tag{11}$$

The updated momentum components are calculated using $\boldsymbol{\alpha} = \boldsymbol{\alpha}_0 - \mathbf{V}_{\alpha_0}\mathbf{D}^T\boldsymbol{\lambda}$:

$$\begin{aligned}
p_{x1(2)} &= p^0_{x1(2)} - \left(\mathbf{V}_{1_0} + \mathbf{V}_{2_0}\right)_{11}\lambda_1 - \left(\mathbf{V}_{1_0} + \mathbf{V}_{2_0}\right)_{12}\lambda_2 - \left(\mathbf{V}_{1_0} + \mathbf{V}_{2}\right)_{13}\lambda_3 \\
p_{y1(2)} &= p^0_{y1(2)} - \left(\mathbf{V}_{1_0} + \mathbf{V}_{2_0}\right)_{21}\lambda_1 - \left(\mathbf{V}_{1_0} + \mathbf{V}_{2_0}\right)_{22}\lambda_2 - \left(\mathbf{V}_{1_0} + \mathbf{V}_{2_0}\right)_{23}\lambda_3 \\
p_{z1(2)} &= p^0_{z1(2)} - \left(\mathbf{V}_{1_0} + \mathbf{V}_{2_0}\right)_{31}\lambda_1 - \left(\mathbf{V}_{1_0} + \mathbf{V}_{2_0}\right)_{32}\lambda_2 - \left(\mathbf{V}_{1_0} + \mathbf{V}_{2_0}\right)_{33}\lambda_3
\end{aligned} \tag{12}$$

Similar equations can be written for the energy and position components by using the appropriate track covariance indices. The χ^2 for the solution is

$$\begin{aligned}
\chi^2 &= \boldsymbol{\lambda}^T\left(\mathbf{D}\delta\boldsymbol{\alpha}_0 + \mathbf{d}\right) \\
&= \lambda_1\left(p^0_{x1} + p^0_{x2}\right) + \lambda_2\left(p^0_{y1} + p^0_{y2}\right) + \lambda_3\left(p^0_{z1} + p^0_{z2}\right)
\end{aligned} \tag{13}$$

The updated covariance matrix can be computed from $\mathbf{V}_\alpha = \mathbf{V}_{\alpha_0} - \mathbf{V}_{\alpha_0}\mathbf{D}^T\mathbf{V}_D\mathbf{D}\mathbf{V}_{\alpha_0}$. This gives for the momentum components

$$\begin{aligned}
(\mathbf{V}_1)_{ij} &= \left(\mathbf{V}_{1_0}\right)_{ij} - \sum_{k,l}\left[\left(\mathbf{V}_{1_0}\right)_{ik}(\mathbf{V}_D)_{kl}\left(\mathbf{V}_{1_0}\right)_{jl}\right] \\
(\mathbf{V}_2)_{ij} &= \left(\mathbf{V}_{2_0}\right)_{ij} - \sum_{k,l}\left[\left(\mathbf{V}_{2_0}\right)_{ik}(\mathbf{V}_D)_{kl}\left(\mathbf{V}_{2_0}\right)_{jl}\right] \\
(\mathbf{V}_{12})_{ij} &= -\sum_{k,l}\left[\left(\mathbf{V}_{1_0}\right)_{ik}(\mathbf{V}_D)_{kl}\left(\mathbf{V}_{2_0}\right)_{jl}\right]
\end{aligned} \tag{14}$$

where all indices run from 1 to 3. The effect of the fit is to reduce the errors of the individual tracks. Note that the second term in the general formula for $\mathbf{V}_\alpha$ mixes the tracks together so they are correlated after the fit.

V Non-vertex constraints

In this section I compute the explicit form of the **D** and **d** matrices for constraints commonly encountered in high energy physics (vertex constraints are discussed in the next section). Once these matrices are known the tracks can be kinematically fit using the procedure described in the previous section. If multiple constraints are desired then one just extends the matrices by adding rows to them, one row per constraint. This allows many constraints to be used simultaneously in the fit. In all cases the tracks are expanded about their current values $\left(p_x, p_y, p_z, E, x, y, z\right)$, so the solution should be seen as changes from these initial values.

V.1 Invariant mass constraint

The constraint equation which forces a track to have an invariant mass m_c is

$$E^2 - p_x^2 - p_y^2 - p_z^2 - m_c^2 = 0, \tag{15}$$

Expanding about the initial parameters $\left(p_x, p_y, p_z, E, x, y, z\right)$, we get for **D** and **d**:

$$\begin{aligned} \mathbf{D} &= \begin{pmatrix} -2p_x & -2p_y & -2p_z & 2E & 0 & 0 & 0 \end{pmatrix} \\ \mathbf{d} &= E^2 - p_x^2 - p_y^2 - p_z^2 - m_c^2 \end{aligned} \tag{16}$$

V.2 Total energy constraint

The constraint that a track must have a total energy E_c can be written $E - E_c = 0$. The **D** and **d** matrices are trivially computed to be

$$\begin{aligned} \mathbf{D} &= \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \\ \mathbf{d} &= E - E_c \end{aligned} \tag{17}$$

V.3 Total momentum constraint

The constraint that a track must have a total momentum p_c can be written

$$\sqrt{p_x^2 + p_y^2 + p_z^2} - p_c = 0, \tag{18}$$

Expanding about an initial set of parameters $\left(p_x, p_y, p_z, E, x, y, z\right)$, we compute the **D** and **d** matrices to be

$$\begin{aligned} \mathbf{D} &= \begin{pmatrix} p_x / p & p_y / p & p_z / p & 0 & 0 & 0 & 0 \end{pmatrix} \\ \mathbf{d} &= \sqrt{p_x^2 + p_y^2 + p_z^2} - p_c \end{aligned} \tag{19}$$

V.4 Total 3-vector constraint

This is a set of three constraints that can be written

$$p_i - p_{ci} = 0 \tag{20}$$

where p_{ci} represents the components of the total 3-momentum (i = 1,2,3). Computing **D** and **d** yields

$$\mathbf{D}=\begin{pmatrix}1&0&0&0&0&0&0\\0&1&0&0&0&0&0\\0&0&1&0&0&0&0\end{pmatrix}$$
$$\mathbf{d}=\begin{pmatrix}p_x-p_{xc}\\p_y-p_{yc}\\p_z-p_{zc}\end{pmatrix}\tag{21}$$

V.5 Total 4-vector constraint

This is a set of four constraints that can be written

$$p^{\mu}-p_c^{\mu}=0,\tag{22}$$

where p_c^{μ} represents the components of the total 4-momentum ($\mu=0,1,2,3$). Computing **D** and **d** yields

$$\mathbf{D}=\begin{pmatrix}1&0&0&0&0&0&0\\0&1&0&0&0&0&0\\0&0&1&0&0&0&0\\0&0&0&1&0&0&0\end{pmatrix}$$
$$\mathbf{d}=\begin{pmatrix}p_x-p_{xc}\\p_y-p_{yc}\\p_z-p_{zc}\\E-E_c\end{pmatrix}\tag{23}$$

V.6 Particle lies in a plane

This constraint, applied to a single track, is useful for cases when one has built a virtual particle and wants to further constrain its vertex location. It is easier in this case to build the virtual particle using a standard vertex fit [6] and then apply the additional requirement to the virtual particle.

The constraint is described by the equation of a plane: $\boldsymbol{\eta}\cdot\left(\mathbf{x}-\mathbf{x}_p\right)=0$, where $\boldsymbol{\eta}$ is a unit vector normal to the plane and $\mathbf{x}_p$ is a point in the plane. The **D** and **d** matrices are given by

$$\begin{aligned}\mathbf{D}&=\begin{pmatrix}0&0&0&0&\eta_x&\eta_y&\eta_z\end{pmatrix}\\\mathbf{d}&=-\boldsymbol{\eta}\cdot\mathbf{x}_p=-\eta_x x_p-\eta_y y_p-\eta_z z_p\end{aligned}\tag{24}$$

V.7 Particle lies on a line

The comments about the usefulness of this constraint from the last subsection apply here. The constraint is described by the equation of a line: $\mathbf{x}=\mathbf{x}_p+\boldsymbol{\eta}s$, where $\mathbf{x}_p$ is some point on the line,

η is a unit vector along the direction of the line and *s* is the length along the line. Eliminating the distance *s* we get for the equations

$$\begin{aligned} x - x_p - (z - z_p)\eta_x / \eta_z = 0 \\ y - y_p - (z - z_p)\eta_y / \eta_z = 0 \end{aligned} \tag{25}$$

The **D** and **d** matrices are then

$$\begin{aligned} \mathbf{D} &= \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & -\eta_x / \eta_z \\ 0 & 0 & 0 & 0 & 0 & 1 & -\eta_y / \eta_z \end{pmatrix} \\ \mathbf{d} &= \begin{pmatrix} -x_p + z_p \eta_x / \eta_z \\ -y_p + z_p \eta_y / \eta_z \end{pmatrix} \end{aligned} \tag{26}$$

If η_z is close to zero we can choose either one of the other coordinates as the independent variable.

V.8 Back to back constraint

This is the one constraint which is actually more difficult to express in our standard *W* representation than in the *C* representation, defined in Appendix I. It expresses the fact that for the decay of a particle at rest into two oppositely charged particles, the three momentum of the daughter particles is equal and opposite and the two particles must originate at the same point, a total of 5 conditions.

The constraints can be written as follows (this parallels the way they would be expressed in the *C* representation):

$$\begin{aligned} p_{\perp 1} - p_{\perp 2} &= 0 \\ \phi_{01} - \phi_{02} &= \pi \\ d_{01} + d_{02} &= 0 \\ p_{z1} + p_{z2} &= 0 \\ z_{01} - z_{02} &= 0 \end{aligned} \tag{27}$$

We can expand these equations in terms of the *W* track parameters. For simplicity I will assume a solenoidal *B* field. This yields the following equations expressing the constraints

$$
\begin{aligned}
&\sqrt{p_{x1}^2 + p_{y1}^2} - \sqrt{p_{x2}^2 + p_{y2}^2} = 0 \\
&\left(p_{x1} + a_1 y_1\right)\left(p_{y2} - a_2 x_2\right) - \left(p_{x2} + a_2 y_2\right)\left(p_{y1} - a_1 x_1\right) = 0 \\
&\left(T_1 - p_{\perp 1}\right) / a_1 + \left(T_2 - p_{\perp 2}\right) / a_2 = 0 \\
&p_{z1} + p_{z2} = 0 \\
z_1 - \frac{p_{z1}}{a_1} \tan^{-1} &\left[\frac{a_1\left(x_1 p_{x1} + y_1 p_{y1}\right)}{p_{\perp 1}^2 - a_1\left(x_1 p_{y1} - y_1 p_{x1}\right)} \right] \\
&-z_2 + \frac{p_{z2}}{a_2} \tan^{-1} \left[\frac{a_2\left(x_2 p_{x2} + y_2 p_{y2}\right)}{p_{\perp 2}^2 - a_2\left(x_2 p_{y2} - y_2 p_{x2}\right)} \right] = 0
\end{aligned}
\tag{28}
$$

where $T_i = \sqrt{\left(p_{xi} + a_i y_i\right)^2 + \left(p_{yi} - a_i x_i\right)^2}$, $a_i = -0.299792458 B Q_i$, Q_i is the charge, and B is the magnetic field strength. Note that the second, third and fifth constraints are expressed at the point of closest approach (in the bend plane) to some point, which I choose to be the origin.

The corresponding **D** matrices are obtained by taking the derivatives of these 5 equations with respect to the 14 track parameters, yielding a 5 × 14 matrix. Unfortunately, I don’t have the energy to do this at the moment, although the constraints have been implemented in **KWFIT**. I will put the derivatives in a later update to the document.

VI Vertex constraints

VI.1 General solution

I will develop some common properties of vertex constraints here before specializing in the rest of this section. Consider a set of *n* tracks forced to pass through a common point $\mathbf{x} = (x, y, z)$. Since the covariance matrix of the vertex may be known in advance (what’s known as a “prior” covariance matrix), we write the overall χ^2 condition generally as

$$
\chi^2 = (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0)^T \mathbf{V}_{\alpha 0}^{-1} (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0) + (\mathbf{x} - \mathbf{x}_0)^T \mathbf{V}_{\mathbf{x}0}^{-1} (\mathbf{x} - \mathbf{x}_0) + 2\boldsymbol{\lambda}^T (\mathbf{D}\delta\boldsymbol{\alpha} + \mathbf{E}\delta\mathbf{x} + \mathbf{d}) \tag{29}
$$

where the terms represent, respectively, the contribution to the χ^2 from the tracks, vertex and the constraints. Note that $\mathbf{x}_0$ and $\mathbf{V}_{\mathbf{x}0}$ represent the initial vertex position and its covariance matrix, while $\delta\boldsymbol{\alpha} = \boldsymbol{\alpha} - \boldsymbol{\alpha}_A$ and $\delta\mathbf{x} = \mathbf{x} - \mathbf{x}_A$ are the departures of the variables from their expansion points.

For each track *i* there are two constraint equations, corresponding to the bend and non-bend planes, respectively. The equations are obtained by eliminating the arc length *s* from the *x*, *y*, *z* equations of motion in Appendix II. For example, in a solenoidal *B* field:

$$\begin{aligned} 0 &= p_{xi}\Delta y_i - p_{yi}\Delta x_i - \frac{a_i}{2}\left(\Delta x_i^2 + \Delta y_i^2\right) \\ 0 &= \Delta z_i - \frac{p_{zi}}{a_i}\sin^{-1}\left[a_i\left(p_{xi}\Delta x_i + p_{yi}\Delta y_i\right)/\, p_{Ti}^2\right] \end{aligned} \tag{30}$$

where $\Delta x_i = x_x - x_i$, etc., $a_i = -0.299792458BQ_i$, Q_i is the charge, B is the magnetic field strength and p_{Ti} is the transverse momentum. A generalization of this formula can be derived for B fields oriented along arbitrary directions using the equations in Appendix II:

$$\begin{aligned} 0 &= \left(\mathbf{p}_i \times \Delta\mathbf{x}_i\right)\cdot\hat{\mathbf{h}} - \frac{a_i}{2}\left(\Delta\mathbf{x}_i^2 - \left(\Delta\mathbf{x}_i\cdot\hat{\mathbf{h}}\right)^2\right) \\ 0 &= \Delta\mathbf{x}_i\cdot\hat{\mathbf{h}} - \frac{\mathbf{p}_i\cdot\hat{\mathbf{h}}}{a_i}\sin^{-1}\left[a_i\left(\mathbf{p}_i\cdot\Delta\mathbf{x}_i - \left(\mathbf{p}_i\cdot\hat{\mathbf{h}}\right)\left(\Delta\mathbf{x}_i\cdot\hat{\mathbf{h}}\right)\right)/\left|\mathbf{p}_i\times\hat{\mathbf{h}}\right|^2\right] \end{aligned} \tag{31}$$

where $\hat{\mathbf{h}}$ is a unit vector in the direction of the magnetic field. The **E** and **D** matrices thus have the simple form

$$\mathbf{E} = \begin{pmatrix} \mathbf{E}_1 \\ \mathbf{E}_2 \\ \vdots \\ \mathbf{E}_n \end{pmatrix} \qquad \mathbf{D} = \begin{pmatrix} \mathbf{D}_1 & & & \\ & \mathbf{D}_2 & & \\ & & \ddots & \\ & & & \mathbf{D}_n \end{pmatrix} \tag{32}$$

where $\mathbf{E}_i$ is a 2×3 matrix and $\mathbf{D}_i$ is a 2×7 matrix. **E** and **D** have this particular block diagonal form because the vertex constraints for each track only involve the parameters for that track. The fact that the constraints do not mix tracks greatly simplifies the solution (if the tracks are initially uncorrelated) because the inversion of the $2n\times 2n$ matrix $\mathbf{V}_D$ can be factored into n 2×2 inversions. In a solenoidal field, the matrices $\mathbf{D}_i$, $\mathbf{E}_i$, and $\mathbf{d}_i$ for each track are given by

$$\begin{aligned} \mathbf{D}_i &= \begin{pmatrix} \Delta y_i & -\Delta x_i & 0 & 0 & p_{yi} + a_i\Delta x_i & -p_{xi} + a_i\Delta y_i & 0 \\ -p_{zi}S_iR_{xi} & -p_{zi}S_iR_{yi} & -\dfrac{\sin^{-1} J_i}{a_i} & 0 & p_{xi}p_{zi}S_i & p_{yi}p_{zi}S_i & -1 \end{pmatrix} \\[2ex] \mathbf{E}_i &= \begin{pmatrix} -\left(p_{yi} + a_i\Delta x_i\right) & p_{xi} - a_i\Delta y_i & 0 \\ -p_{xi}p_{zi}S_i & -p_{yi}p_{zi}S_i & 1 \end{pmatrix} \\[2ex] \mathbf{d}_i &= \begin{pmatrix} \Delta y_i p_{xi} - \Delta x_i p_{yi} - \frac{1}{2}a_i\left(\Delta x_i^2 + \Delta y_i^2\right) \\ \Delta z_i - \dfrac{p_{zi}}{a_i}\sin^{-1} J_i \end{pmatrix} \end{aligned} \tag{33}$$

where the auxiliary quantities J, R_x, R_y, and S are defined as follows:

$$
\begin{aligned}
J &= a\left(\Delta x\, p_x + \Delta y\, p_y\right) / \, p_T^2 \\
R_{x(y)} &= \Delta x - 2 p_{x(y)}\left(\Delta x\, p_x + \Delta y\, p_y\right) / \, p_T^2 \\
S &= \frac{1}{p_T^2 \sqrt{1 - J^2}}
\end{aligned} \tag{34}
$$

The solution to this problem is straightforward, but algebraically tedious, and I leave the details to Appendix III. The solution is:

$$
\begin{aligned}
\delta\boldsymbol{\alpha} &= \delta\boldsymbol{\alpha}_0 - \mathbf{V}_0 \mathbf{D}^T \boldsymbol{\lambda} \\
\delta\mathbf{x} &= \delta\mathbf{x}_0 - \mathbf{V}_{\mathbf{x}0} \mathbf{E}^T \boldsymbol{\lambda} = \mathbf{V}_{\mathbf{x}} \mathbf{V}_{\mathbf{x}0}^{-1} \delta\mathbf{x}_0 - \mathbf{V}_{\mathbf{x}} \mathbf{E}^T \boldsymbol{\lambda}_0 \\
\boldsymbol{\lambda} &= \mathbf{V}_{\tilde{D}}\left(\mathbf{D}\delta\boldsymbol{\alpha}_0 + \mathbf{E}\delta\mathbf{x}_0 + \mathbf{d}\right) = \boldsymbol{\lambda}_0 + \mathbf{V}_D \mathbf{E}\delta\mathbf{x} \\
\boldsymbol{\lambda}_0 &= \mathbf{V}_D\left(\mathbf{D}\delta\boldsymbol{\alpha}_0 + \mathbf{d}\right) \\
\mathbf{V}_{\tilde{D}} &= \mathbf{V}_D - \mathbf{V}_D \mathbf{E} \mathbf{V}_{\mathbf{x}} \mathbf{E}^T \mathbf{V}_D
\end{aligned} \tag{35}
$$

where $\delta\boldsymbol{\alpha} = \boldsymbol{\alpha} - \boldsymbol{\alpha}_A$ and $\delta\mathbf{x} = \mathbf{x} - \mathbf{x}_A$ are the deviations of the parameters from their expansion points. The covariance matrices are

$$
\begin{aligned}
\mathbf{V}_{\mathbf{x}} &= \left(\mathbf{V}_{\mathbf{x}0}^{-1} + \mathbf{E}^T \mathbf{V}_D \mathbf{E}\right)^{-1} \\
\mathbf{V}_{\boldsymbol{\alpha}} &= \mathbf{V}_{\boldsymbol{\alpha}0} - \mathbf{V}_{\boldsymbol{\alpha}0} \mathbf{D}^T \mathbf{V}_D \mathbf{D} \mathbf{V}_{\boldsymbol{\alpha}0} + \mathbf{V}_{\boldsymbol{\alpha}0} \mathbf{D}^T \mathbf{V}_D \mathbf{E} \mathbf{V}_{\mathbf{x}} \mathbf{E}^T \mathbf{V}_D \mathbf{D} \mathbf{V}_{\boldsymbol{\alpha}0} \\
\operatorname{cov}(\boldsymbol{\alpha}, \mathbf{x}) &= -\mathbf{V}_{\boldsymbol{\alpha}0} \mathbf{D}^T \mathbf{V}_D \mathbf{E} \mathbf{V}_{\mathbf{x}}
\end{aligned} \tag{36}
$$

and the χ^2 is given by

$$
\begin{aligned}
\chi^2 &= \boldsymbol{\lambda}^T \mathbf{V}_{\tilde{D}}^{-1} \boldsymbol{\lambda} = \boldsymbol{\lambda}^T \left(\mathbf{V}_D^{-1} + \mathbf{E} \mathbf{V}_{\mathbf{x}0} \mathbf{E}^T\right) \boldsymbol{\lambda} \\
&= \boldsymbol{\lambda}^T \left(\mathbf{D}\delta\boldsymbol{\alpha}_0 + \mathbf{E}\delta\mathbf{x}_0 + \mathbf{d}\right)
\end{aligned} \tag{37}
$$

The physical meaning of the covariance matrices can be explained as follows. The vertex error matrix $\mathbf{V}_{\mathbf{x}}$ is the weighted average of its initial covariance matrix and the errors determined from the tracks. The track error matrix $\mathbf{V}_{\boldsymbol{\alpha}}$ has an initial piece that is decreased by the constraints applied per track and is increased by the wiggle of the vertex itself. Note only this last term correlates the tracks with one another.

VI.2 Vertex constraint to a fixed position

In this case, the vertex position **x** is fixed and the solution can be obtained by setting $\delta\mathbf{x}_0 = 0$, **E** = 0 and $\mathbf{V}_{\mathbf{x}0} = 0$. The solution factors into *n* pieces, one per track *i*:

$$
\begin{aligned}
\alpha_i &= \alpha_{0i} - \mathbf{V}_{\alpha_{0i}} \mathbf{D}_i^T \lambda_i \\
\lambda_i &= \mathbf{V}_{Di} \left(\mathbf{D}_i \delta\alpha_{0i} + \mathbf{d}_i \right) \\
\mathbf{V}_{Di} &= \left(\mathbf{D}_i \mathbf{V}_{\alpha_{0i}} \mathbf{D}_i^T \right)^{-1} \\
\mathbf{V}_{\alpha_i} &= \mathbf{V}_{\alpha_{0i}} - \mathbf{V}_{\alpha_{0i}} \mathbf{D}_i^T \mathbf{V}_{Di} \mathbf{D}_i \mathbf{V}_{\alpha_{0i}} \\
\chi^2 &= \sum \lambda_i^T \mathbf{V}_{Di}^{-1} \lambda_i = \sum \lambda_i^T \left(\mathbf{D}_i \delta\alpha_{0i} + \mathbf{d}_i \right)
\end{aligned}
\tag{38}
$$

The tracks remain uncorrelated after the fit because each track is fit separately to the fixed point.

VI.3 Vertex constraint to an unknown position

In this case the vertex position $\mathbf{x}$ must be determined from the constraints. The simplest approach is to assign large values to $\mathbf{V}_{\mathbf{x}0}$, the initial vertex covariance matrix, and apply the method from the first subsection. This method of assigning a large prior covariance matrix to a set of parameters is discussed in more detail in ref. [3]. Note that when using this technique one has an effective number of degrees of freedom equal to $2n-3$ because the fit causes the three vertex parameters to move an insignificant number of standard deviations.

VII Vertex constraint with the vertex position satisfying other conditions

Suppose one wanted to determine a vertex whose position satisfies some other constraint. For example, in e^+e^- collisions the vertical spread of the beam may be much smaller than the resolution so that the vertex effectively lies in a horizontal plane (1 constraint), or the vertex might lie on the trajectory of another particle (2 constraints).

The fit can be carried out in basically two ways. One method is to incorporate the extra information directly in the vertex fit. This leads to a somewhat more complicated but solvable problem, except that several vertex fitting routines need to be written, one for each variation.

A better method is to do the fit in two steps, first determining the vertex using the algorithm from Section VI and then applying the additional constraints. This two step procedure is mathematically identical to applying all the constraints simultaneously, as I have proven elsewhere [3]. The final χ^2 is obtained by adding together the χ^2 values for the two steps.

Note, however, that the input tracks are not updated by the two step procedure. To update the track parameters using the extra information, we would have to update the track covariance matrices after the initial vertex constraint, a process that would introduce correlations between the tracks. This is not particularly difficult to work out, but it does not seem necessary at this time to keep track of the updated input tracks.

VIII Miscellaneous vertex manipulations

VIII.1 Averaging two vertices

Suppose we have two estimates of a vertex, each with a different covariance matrix. How do we combine them using the full information? The best value, which I call **z**, can be found by turning the problem into a χ^2 minimization:

$$\chi^2_{\text{ave}} = (\mathbf{z}-\mathbf{x})^T \mathbf{V}_x^{-1}(\mathbf{z}-\mathbf{x}) + (\mathbf{z}-\mathbf{y})^T \mathbf{V}_y^{-1}(\mathbf{z}-\mathbf{y}) \tag{39}$$

The “best” **z** is the one that minimizes this χ^2. A straightforward procedure gives the solution

$$\begin{aligned}\mathbf{z} &= \left(\mathbf{V}_x^{-1}+\mathbf{V}_y^{-1}\right)^{-1}\left(\mathbf{V}_x^{-1}\mathbf{x}+\mathbf{V}_y^{-1}\mathbf{y}\right)\\ &= \mathbf{x}+\mathbf{V}_x\left(\mathbf{V}_x+\mathbf{V}_y\right)^{-1}(\mathbf{y}-\mathbf{x})\end{aligned} \tag{40}$$

with $\chi^2_{\text{ave}} = (\mathbf{y}-\mathbf{x})^T\left(\mathbf{V}_x+\mathbf{V}_y\right)^{-1}(\mathbf{y}-\mathbf{x})$. Note that this solution is the generalization of the well-known average of two quantities having different errors (it is trivial to generalize to any number of vertices). Moreover, the above formulation only requires the inverse of a single matrix, $\left(\mathbf{V}_x+\mathbf{V}_y\right)^{-1}$, so $\mathbf{V}_x$ and $\mathbf{V}_y$ can individually be singular.

The averaging procedure offers an interesting way to “factor” the vertex fitting problem. Suppose you have *n* tracks whose common vertex you are trying to find. The tracks can be broken up into two disjoint sets with n_1 and n_2 tracks, respectively, and separate vertices $\mathbf{v}_1$ and $\mathbf{v}_2$ can be found. The overall vertex position is just the weighted average of the two vertices and the total χ^2 is equal to $\chi_1^2+\chi_2^2+\chi^2_{\text{ave}}$. This result provides a useful way of testing vertex fitting algorithms. As described in the last Section, however, the averaging procedure provides no way to update the input tracks going into the vertex.

VIII.2 Testing for equality of two vertices

It is useful occasionally to test whether two vertices are identical. The problem can be cast into a constraint problem, and the χ^2 determined in this way turns out to be the same as in the previous subsection, as a moment's reflection tells us it should. If the vertices are denoted **x** and **y**, having covariance matrices $\mathbf{V}_x$ and $\mathbf{V}_y$, respectively, the χ^2 for equality is given by

$$\chi^2_{\text{equal}} = (\mathbf{y}-\mathbf{x})^T\left(\mathbf{V}_x+\mathbf{V}_y\right)^{-1}(\mathbf{y}-\mathbf{x}) \tag{41}$$

The value of χ^2_{equal} can be converted to a confidence level using the fact that there are three degrees of freedom.

IX Building virtual particles

A virtual particle is made by combining a set of measured tracks into a single one. Common examples in high energy physics are the π^0, K_s and Λ, all of which decay into two specific kinds of particles and have had techniques developed especially for finding them. We would like to generalize this process so that, for example, one could combine the daughter products in $D^0 \to K^- \pi^+ \pi^0$ and make the parent D^0. Once constructed with a set of parameters and a covariance matrix, the virtual particle indistinguishable from any other measured particle and can be used in further kinematic fits or used to build still other particles.

The process of creating virtual particle offers an alternative way of kinematically fitting a decay sequence. Consider the cascade decay described at the beginning of this note:

$$\begin{aligned} &\overline{B}^0 \to D^{*+}\pi^- \\ &\qquad D^{*+} \to D^0 \pi^+ \\ &\qquad\qquad D^0 \to K^- \pi^+ \pi^0 \end{aligned} \tag{42}$$

The full fit proceeds as follows. First one builds the D^0 from the $K^- \pi^+ \pi^0$ tracks and calculates a χ^2. Then the D^0 is forced to satisfy a mass constraint, generating another χ^2. This process is repeated until the $\overline{B}^0$ is constructed; the total χ^2 for the sequence is the sum of the individual χ^2's. The values for the track parameters and the total χ^2 are exactly what would have been obtained if the entire sequence had been fit simultaneously. This is a special case of a theorem I proved in Ref. [3], which says that the results of a fit do not depend on whether constraints are used sequentially or simultaneously. The mathematical details of how virtual particles are constructed are described in ref. [6].

Appendix I

Converting charged track information to the *W* format

The *C* representation is used in charged track fitting in CLEO and several other HEP experiments. The 5 parameters $\alpha_C = (c, \phi_0, d_0, \lambda, z_0)$, measured relative to the reference point (x_r, y_r, z_r), describe the entire shape of the helix, where $c = 1/2R$ (*R* is the radius of curvature), ϕ_0 is the ϕ of the track momentum vector at the point of closest approach to the reference point, d_0 is the signed distance of closest approach to the reference point in the *x*–*y* plane, $\lambda = \cot\theta$, where θ is the polar angle measured from the +*z* axis, and z_0 is the *z* position of the track at the point of closest approach to the reference point in the *x*–*y* plane. I assume that the *B* field is oriented along the +*z* direction.

The seven *W* parameters at the point of closest approach to the reference point can then be calculated in terms of the five canonical parameters, as shown below:

$$
\begin{aligned}
p_x &= \frac{a}{2c}\cos\phi_0 \\
p_x &= \frac{a}{2c}\sin\phi_0 \\
p_x &= \frac{a}{2c}\lambda \\
E &= \sqrt{(a/2c)^2\left(1+\lambda^2\right)+m^2} \\
x &= x_r - d_0\sin\phi_0 \\
y &= y_r + d_0\cos\phi_0 \\
z &= z_r + z_0
\end{aligned}
\tag{43}
$$

where $a = -0.299792458Bq$, *B* is the magnetic field in Tesla and *q* is the charge in units of *e*. The values of $\left(p_x, p_y, p_z, E\right)$ defined here are used by default in CLEO analyses.

The 7 × 7 covariance matrix for the *W* parameters can be computed by taking the differentials of the above equations:

$$
\begin{aligned}
\delta p_x &= -\frac{p_x}{c}\delta c - p_y\,\delta\phi_0 \\
\delta p_y &= -\frac{p_y}{c}\delta c + p_x\,\delta\phi_0 \\
\delta p_z &= -\frac{p_z}{c}\delta c + p_\perp\,\delta\lambda \\
\delta E &= -\frac{p^2}{Ec}\delta c + \frac{\lambda p_\perp^2}{E}\delta\lambda \\
\delta x &= -\frac{p_y}{p_\perp}\delta d_0 - y\,\delta\phi_0 \\
\delta y &= +\frac{p_y}{p_\perp}\delta d_0 + x\,\delta\phi_0 \\
\delta z &= \delta z_0
\end{aligned}
\tag{44}
$$

V_W can be computed from V_C using the above formulas and propagation of errors. For example,

$$
\begin{aligned}
(V_W)_{11} &= \frac{p_x^2}{c^2}(V_C)_{11} + \frac{2p_x p_y}{c}(V_C)_{12} + p_y^2(V_C)_{22} \\
(V_W)_{12} &= \frac{p_x p_y}{c^2}(V_C)_{11} + \frac{\left(p_y^2 - p_x^2\right)}{c}(V_C)_{12} - p_x p_y(V_C)_{22} \\
(V_W)_{13} &= \frac{p_x p_z}{c^2}(V_C)_{11} - \frac{p_x p_\perp}{c}(V_C)_{14} - p_y p_\perp(V_C)_{24}
\end{aligned}
\tag{45}
$$

These follows from the definition $(V_W)_{11} \equiv \mathrm{cov}(p_x, p_x) = \langle \delta p_x\,\delta p_x \rangle$, etc.

Appendix II

Motion of a charged particle in a magnetic field

It can easily be shown by solving the Lorentz force equation $d\mathbf{p} / dt = q\mathbf{v} \times \mathbf{B}$ that a charged particle traces out a helix in space, with the axis of the helix lying along the magnetic field direction. For a solenoidal field aligned along the *z* direction, as encountered in most colliding beam experiments, particles move in circles in the $r-\phi$ plane with a radius of curvature $R = p_T / a$, where $a = -0.299792458Bq$, *B* is the magnetic field strength in Tesla, *q* is the charge and p_T is the momentum transverse to the *z* axis. The equations of motion as a function of the path length *s* are then:

$$\begin{aligned}
p_x &= p_{0x} \cos\rho s - p_{0y} \sin\rho s \\
p_y &= p_{0y} \cos\rho s + p_{0x} \sin\rho s \\
p_z &= p_{0z} \\
x &= x_0 + \frac{p_{0x}}{a} \sin\rho s - \frac{p_{0y}}{a}(1-\cos\rho s) \\
y &= y_0 + \frac{p_{0y}}{a} \sin\rho s + \frac{p_{0x}}{a}(1-\cos\rho s) \\
z &= z_0 + \frac{p_{0z}}{p} s
\end{aligned} \tag{47}$$

where $\rho = a / p$, (x_0, y_0, z_0) is a known point on the helix and (p_{0x}, p_{0y}, p_{0z}) is its 3-momentum there. They are functions of *s*, the arc length measured from the point (x_0, y_0, z_0) to (x, y, z) (the radius of curvature is $R = p_T / a$). Eliminating *s* we can express momentum in terms of position:

$$\begin{aligned}
p_x &= p_{x0} - a(y - y_0) \\
p_y &= p_{y0} + a(x - x_0) \\
p_z &= p_{z0}
\end{aligned} \tag{48}$$

When the *B* field is not aligned along the *z* axis, it is useful to write the equations in a rotationally invariant form. We define a right-handed coordinate system based on these vectors:

$$\begin{gathered}
-(\mathbf{p}_0 \times \hat{\mathbf{h}}) \times \hat{\mathbf{h}} \\
-(\mathbf{p}_0 \times \hat{\mathbf{h}}) \\
\hat{\mathbf{h}}
\end{gathered} \tag{49}$$

where $\hat{\mathbf{h}}$ is a unit vector pointing along the direction of the magnetic field. By identifying the components from the solenoidal case with the general case, the momentum and position components can be written (these can be derived directly from the equations of motion under the Lorentz force)

$$\begin{aligned}
\mathbf{p} &= -\left(\mathbf{p}_0 \times \hat{\mathbf{h}}\right) \times \hat{\mathbf{h}} \cos \rho s - \mathbf{p}_0 \times \hat{\mathbf{h}} \sin \rho s + \left(\mathbf{p}_0 \cdot \hat{\mathbf{h}}\right)\hat{\mathbf{h}} \\
\mathbf{r} &= \mathbf{r}_0 - \frac{\left(\mathbf{p}_0 \times \hat{\mathbf{h}}\right) \times \hat{\mathbf{h}}}{a} \sin \rho s - \frac{\mathbf{p}_0 \times \hat{\mathbf{h}}}{a}(1 - \cos \rho s) + \frac{\mathbf{p}_0 \cdot \hat{\mathbf{h}}}{p} \hat{\mathbf{h}} s \\
\mathbf{p} &= \mathbf{p}_0 - a\left(\mathbf{r} - \mathbf{r}_0\right) \times \hat{\mathbf{h}}
\end{aligned} \tag{50}$$

Appendix III

Derivation of the Vertex Constraint Solution

We write the overall χ^2 condition for a vertex constraint as

$$\chi^2 = (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0)^T \mathbf{V}_{\boldsymbol{\alpha}0}^{-1} (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0) + (\mathbf{x} - \mathbf{x}_0)^T \mathbf{V}_{\mathbf{x}0}^{-1} (\mathbf{x} - \mathbf{x}_0) + 2\boldsymbol{\lambda}^T (\mathbf{D}\delta\boldsymbol{\alpha} + \mathbf{E}\delta\mathbf{x} + \mathbf{d}) \tag{51}$$

where the terms represent, respectively, the contribution to the χ^2 from the tracks, the vertex and the vertex constraint. Note that $\mathbf{x}_0$ and $\mathbf{V}_{\mathbf{x}0}$ represent the initial vertex position and its covariance matrix, while **D**, **E** and **d** are defined in Section VI. Note that $\delta\boldsymbol{\alpha} = \boldsymbol{\alpha} - \boldsymbol{\alpha}_A$ and $\delta\mathbf{x} = \mathbf{x} - \mathbf{x}_A$ are the deviations of the track and vertex coordinates about their expansion points.

To cast the χ^2 equation into a familiar form, we write it the using “extended” matrices combining information from the vertex and the tracks:

$$\tilde{\boldsymbol{\alpha}} = \begin{pmatrix} \mathbf{x} \\ \boldsymbol{\alpha}_1 \\ \vdots \\ \boldsymbol{\alpha}_n \end{pmatrix} \quad \mathbf{V}_{\tilde{\boldsymbol{\alpha}}0} = \begin{pmatrix} \mathbf{V}_{\mathbf{x}0} & & & \\ & \mathbf{V}_{01} & & \\ & & \ddots & \\ & & & \mathbf{V}_{0n} \end{pmatrix} \quad \tilde{\mathbf{D}} = \begin{pmatrix} \mathbf{E}_1 & \mathbf{D}_1 & & & \\ \mathbf{E}_2 & & \mathbf{D}_2 & & \\ \vdots & & & \ddots & \\ \mathbf{E}_n & & & & \mathbf{D}_n \end{pmatrix} \tag{52}$$

giving the new equation

$$\chi^2 = (\tilde{\boldsymbol{\alpha}} - \tilde{\boldsymbol{\alpha}}_0)^T \mathbf{V}_{\tilde{\boldsymbol{\alpha}}0}^{-1} (\tilde{\boldsymbol{\alpha}} - \tilde{\boldsymbol{\alpha}}_0) + 2\boldsymbol{\lambda}^T \left(\tilde{\mathbf{D}}\delta\tilde{\boldsymbol{\alpha}} + \mathbf{d}\right) \tag{53}$$

The minimization of this χ^2 with respect to all the variables has the standard solution [3]:

$$\begin{aligned} \delta\tilde{\boldsymbol{\alpha}} &= \delta\tilde{\boldsymbol{\alpha}}_0 - \mathbf{V}_{\tilde{\boldsymbol{\alpha}}_0} \tilde{\mathbf{D}}^T \boldsymbol{\lambda} \\ \boldsymbol{\lambda} &= \mathbf{V}_{\tilde{D}} \left(\tilde{\mathbf{D}}\delta\tilde{\boldsymbol{\alpha}}_0 + \mathbf{d}\right) \\ \mathbf{V}_{\tilde{D}} &= \left(\tilde{\mathbf{D}}\mathbf{V}_{\tilde{\boldsymbol{\alpha}}0}\tilde{\mathbf{D}}^T\right)^{-1} \\ \mathbf{V}_{\tilde{\boldsymbol{\alpha}}} &= \mathbf{V}_{\tilde{\boldsymbol{\alpha}}0} - \mathbf{V}_{\tilde{\boldsymbol{\alpha}}0}\tilde{\mathbf{D}}^T \mathbf{V}_{\tilde{D}} \tilde{\mathbf{D}} \mathbf{V}_{\tilde{\boldsymbol{\alpha}}0} \\ \chi^2 &= \boldsymbol{\lambda}^T \mathbf{V}_{\tilde{D}}^{-1} \boldsymbol{\lambda} = \boldsymbol{\lambda}^T \left(\tilde{\mathbf{D}}\delta\tilde{\boldsymbol{\alpha}}_0 + \mathbf{d}\right) \end{aligned} \tag{54}$$

where $\delta\tilde{\boldsymbol{\alpha}}_0 = \tilde{\boldsymbol{\alpha}}_0 - \tilde{\boldsymbol{\alpha}}_A$. The vertex and track part of the solution can be separated out to give:

$$\begin{aligned} \delta\boldsymbol{\alpha} &= \delta\boldsymbol{\alpha}_0 - \mathbf{V}_{\boldsymbol{\alpha}0}\mathbf{D}^T\boldsymbol{\lambda} \\ \delta\mathbf{x} &= \delta\mathbf{x}_0 - \mathbf{V}_{\mathbf{x}0}\mathbf{E}^T\boldsymbol{\lambda} \\ \boldsymbol{\lambda} &= \mathbf{V}_{\tilde{D}}(\mathbf{D}\delta\boldsymbol{\alpha}_0 + \mathbf{E}\delta\mathbf{x}_0 + \mathbf{d}) = \boldsymbol{\lambda}_0 + \mathbf{V}_D\mathbf{E}\delta\mathbf{x} \\ \boldsymbol{\lambda}_0 &= \mathbf{V}_D(\mathbf{D}\delta\boldsymbol{\alpha}_0 + \mathbf{d}) \\ \mathbf{V}_{\tilde{D}} &= \left(\mathbf{D}\mathbf{V}_{\boldsymbol{\alpha}0}\mathbf{D}^T + \mathbf{E}\mathbf{V}_{\mathbf{x}0}\mathbf{E}^T\right)^{-1} \end{aligned} \tag{55}$$

with covariance matrices

$$\begin{aligned}\mathbf{V}_{\boldsymbol{\alpha}} &= \mathbf{V}_{\boldsymbol{\alpha}0} - \mathbf{V}_{\boldsymbol{\alpha}0}\mathbf{D}^T\mathbf{V}_{\tilde{D}}\mathbf{D}\mathbf{V}_{\boldsymbol{\alpha}0} \\ \mathbf{V}_{\mathbf{x}} &= \mathbf{V}_{\mathbf{x}0} - \mathbf{V}_{\mathbf{x}0}\mathbf{E}^T\mathbf{V}_{\tilde{D}}\mathbf{E}\mathbf{V}_{\mathbf{x}0} \\ \mathrm{cov}(\boldsymbol{\alpha},\mathbf{x}) &= -\mathbf{V}_{\boldsymbol{\alpha}0}\mathbf{D}^T\mathbf{V}_{\tilde{D}}\mathbf{E}\mathbf{V}_{\mathbf{x}0}\end{aligned} \tag{56}$$

The only non-trivial part of the calculation involves the calculation of $\mathbf{V}_{\tilde{D}} = \left(\tilde{\mathbf{D}}\mathbf{V}_{\tilde{\boldsymbol{\alpha}}_0}\tilde{\mathbf{D}}^T\right)^{-1}$, which is a $2n \times 2n$ matrix. This is where the particular form of the **D** matrix comes into play. We write

$$\mathbf{V}_{\tilde{D}} = \left(\tilde{\mathbf{D}}\mathbf{V}_{\tilde{\boldsymbol{\alpha}}0}\tilde{\mathbf{D}}^T\right)^{-1} = \left(\mathbf{E}\mathbf{V}_{\mathbf{x}0}\mathbf{E}^T + \mathbf{D}\mathbf{V}_{\boldsymbol{\alpha}0}\mathbf{D}^T\right)^{-1} \tag{57}$$

We invert this using the Woodbury theorem, $(A+UV)^{-1} = A^{-1} - A^{-1}U\left(VA^{-1}U\right)^{-1}VA^{-1}$, where A is an invertible matrix. U and V should be selected to make the matrix in parentheses have small dimensions. Choosing $U = \mathbf{E}\mathbf{V}_{\mathbf{x}0}$ and $V = \mathbf{E}^T$ yields the inverse

$$\begin{aligned}\mathbf{V}_{\tilde{D}} &= \mathbf{V}_D - \mathbf{V}_D\mathbf{E}\mathbf{V}_{\mathbf{x}0}\left(1+\mathbf{E}^T\mathbf{V}_D\mathbf{E}\mathbf{V}_{\mathbf{x}0}\right)^{-1}\mathbf{E}^T\mathbf{V}_D \\ &= \mathbf{V}_D - \mathbf{V}_D\mathbf{E}\left(\mathbf{V}_{\mathbf{x}0}^{-1} + \mathbf{E}^T\mathbf{V}_D\mathbf{E}\right)^{-1}\mathbf{E}^T\mathbf{V}_D\end{aligned} \tag{58}$$

The quantity in parentheses is of size 3×3. $\mathbf{V}_D$ is the block diagonal matrix

$$\mathbf{V}_D = \begin{pmatrix}\mathbf{V}_{D1} & & & \\ & \mathbf{V}_{D2} & & \\ & & \ddots & \\ & & & \mathbf{V}_{Dn}\end{pmatrix} \qquad \mathbf{V}_{Di} = \left(\mathbf{D}_i\mathbf{V}_{0i}\mathbf{D}_i^T\right)^{-1} \tag{59}$$

and each $\mathbf{V}_{Di}$ is a 2×2 matrix. Thus the problem reduces to inverting n 2×2 matrices and one 3×3 matrix and doing a lot of matrix multiplication. The full solution reduces to:

$$\begin{aligned}\delta\boldsymbol{\alpha} &= \delta\boldsymbol{\alpha}_0 - \mathbf{V}_{\boldsymbol{\alpha}0}\mathbf{D}^T\boldsymbol{\lambda} \\ \delta\mathbf{x} &= \delta\mathbf{x}_0 - \mathbf{V}_{\mathbf{x}0}\mathbf{E}^T\boldsymbol{\lambda} = \mathbf{V}_{\mathbf{x}}\mathbf{V}_{\mathbf{x}0}^{-1}\delta\mathbf{x}_0 - \mathbf{V}_{\mathbf{x}}\mathbf{E}^T\boldsymbol{\lambda}_0 \\ \boldsymbol{\lambda} &= \mathbf{V}_{\tilde{D}}\left(\mathbf{D}\delta\boldsymbol{\alpha}_0 + \mathbf{E}\delta\mathbf{x}_0 + \mathbf{d}\right) = \boldsymbol{\lambda}_0 + \mathbf{V}_D\mathbf{E}\delta\mathbf{x} \\ \boldsymbol{\lambda}_0 &= \mathbf{V}_D\left(\mathbf{D}\delta\boldsymbol{\alpha}_0 + \mathbf{d}\right) \\ \mathbf{V}_{\tilde{D}} &= \mathbf{V}_D - \mathbf{V}_D\mathbf{E}\mathbf{V}_{\mathbf{x}}\mathbf{E}^T\mathbf{V}_D\end{aligned} \tag{60}$$

with covariance matrices

$$\begin{aligned}
\mathbf{V}_{\mathbf{x}} &= \left(\mathbf{V}_{\mathbf{x}0}^{-1} + \mathbf{E}^T \mathbf{V}_D \mathbf{E}\right)^{-1} \\
\mathbf{V}_{\alpha} &= \mathbf{V}_{\alpha 0} - \mathbf{V}_{\alpha 0}\mathbf{D}^T \mathbf{V}_D \mathbf{D} \mathbf{V}_{\alpha 0} + \mathbf{V}_{\alpha 0}\mathbf{D}^T \mathbf{V}_D \mathbf{E} \mathbf{V}_{\mathbf{x}} \mathbf{E}^T \mathbf{V}_D \mathbf{D} \mathbf{V}_{\alpha 0} \\
\mathrm{cov}(\alpha, \mathbf{x}) &= -\mathbf{V}_{\alpha 0}\mathbf{D}^T \mathbf{V}_D \mathbf{E} \mathbf{V}_{\mathbf{x}}
\end{aligned} \tag{61}$$

and the χ^2 is given by

$$\begin{aligned}
\chi^2 &= \lambda^T \mathbf{V}_{\tilde{D}}^{-1} \lambda = \lambda^T \left(\mathbf{V}_D^{-1} + \mathbf{E}\mathbf{V}_{\mathbf{x}0}\mathbf{E}^T\right)\lambda \\
&= \lambda^T \left(\mathbf{D}\delta\alpha_0 + \mathbf{E}\delta\mathbf{x}_0 + \mathbf{d}\right)
\end{aligned} \tag{62}$$

References

1. Paul Avery, “Data Analysis and Kinematic Fitting with the **KWFIT** Library”, CBX 98–, www.phys.ufl.edu/~avery/kwfit or www.lns.cornell.edu/~avery/kwfit.
2. Bill Ford, “The VCFIT Vertex Fitting Program”, CSN 94–329.
3. Paul Avery, “Applied Fitting Theory I: General Least Squares Theory”, CBX 91–72, www.phys.ufl.edu/~avery/fitting.html.
4. Paul Avery, “Applied Fitting Theory IV: Formulas for Track Fitting”, CBX 92–45, www.phys.ufl.edu/~avery/fitting.html.
5. Paul Avery, “Applied Fitting Theory V: Track Fitting Using the Kalman Filter”, CBX 92–39, www.phys.ufl.edu/~avery/fitting.html.
6. Paul Avery, “Applied Fitting Theory VII: Building Virtual Particles”, CBX 97–, www.phys.ufl.edu/~avery/fitting.html.