Paul Avery
CBX 98–38
June 8, 1998
Apr. 17, 1999 (rev.)

Applied Fitting Theory VII

Building Virtual Particles

I Statement of the problem

In many physics analyses we encounter the problem of merging a set of particles into a single particle so that this new "virtual" particle can be used directly in the analysis. For example, the K_s parameters from the decay $K_s \to \pi^+\pi^-$ are commonly used, but we also need decays such as $D^+ \to K^-\pi^+\pi^+$ or even complete decay chains, e.g.

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

where we want to (1) combine $K^-\pi^+$ to form a D^0, (2) merge the D^0 with a π^+ to make a D^{*+} and (3) combine the D^{*+} with a π^- to make a $\overline{B}^0$. The basic idea is that once a virtual particle has been built, we can forget about the original particles that went into it. However, the procedure for building the new particle is complicated by the fact that the original particles have measured parameters with associated error (covariance) matrices. The merging process has to take this information into account in such a way that the merged parameters and their error matrix provide the best information possible about the reconstructed particle.

In this note[1] I will show how to calculate the track parameters and covariance matrix of a particle by applying a vertex and possibly other constraints to a set of particles and then adding together the 4–momenta of n "daughter" particles at the vertex point. The track parameters for particle i are denoted by α_i, where

$$\alpha_i = \begin{pmatrix} \mathbf{p}_i \\ \mathbf{x}_i \end{pmatrix} = \begin{pmatrix} p_{xi} \\ p_{yi} \\ p_{zi} \\ E_i \\ x_i \\ y_i \\ z_i \end{pmatrix} \tag{2}$$

[1] This document can be found on the web on my fitting page at www.phys.ufl.edu/~avery/fitting.html

i.e., a 4–momentum and a position at which the momentum is evaluated. The full set of *n* tracks is denoted by $\boldsymbol{\alpha}$, e.g.,

$$\boldsymbol{\alpha} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{pmatrix} \tag{3}$$

The initial covariance matrix for these particles is called $\mathbf{V}_{\boldsymbol{\alpha}0}$. This has the block diagonal form

$$\mathbf{V}_{\boldsymbol{\alpha}0} = \begin{pmatrix} \mathbf{V}_{01} & & & \\ & \mathbf{V}_{02} & & \\ & & \ddots & \\ & & & \mathbf{V}_{0n} \end{pmatrix} \tag{4}$$

expressing the fact that the particles are initially independent. The track parameters for the virtual particle are called $\alpha_V = \begin{pmatrix} \mathbf{p}_V \\ \mathbf{x}_V \end{pmatrix}$.

II Building a virtual particle with vertex constraints

Equations defining the virtual particle

The motion of a charged particle in an arbitrarily oriented, fixed magnetic field was described in Appendix II in ref. [1]. A particle starting out with momentum $\mathbf{p}_0$ at position $\mathbf{x}_0$ will have a momentum $\mathbf{p}$ at position $\mathbf{x}$ given by $\mathbf{p} = \mathbf{p}_0 + a(\mathbf{x} - \mathbf{x}_0) \times \hat{\mathbf{h}}$, where $a = -0.299792458BQ$, where B is the magnetic field in Tesla, Q is the charge of the particle and $\hat{\mathbf{h}}$ is a unit vector in the direction of magnetic field. The virtual particle parameters at the point $\mathbf{x} = (x_x, x_y, x_z)$ can then be calculated by adding the contributions from *n* tracks:

$$\begin{aligned} p_{Vx} &= \sum_i \left[p_{0ix} + a_{zi}(y_{0i} - x_y) - a_{yi}(z_{0i} - x_z) \right] \\ p_{Vy} &= \sum_i \left[p_{0iy} + a_{xi}(z_{0i} - x_z) - a_{zi}(x_{0i} - x_x) \right] \\ p_{Vz} &= \sum_i \left[p_{0iz} + a_{yi}(x_{0i} - x_x) - a_{xi}(y_{0i} - x_y) \right] \\ E_V &= \sum_i E_{0i} \\ \mathbf{x}_V &= \mathbf{x} \end{aligned} \tag{5}$$

where, e.g., $a_{xi} = -0.299792458 B_x Q_i$, Q_i is the charge on the particle and B_x is the *x* component of the *B* field.

Applying the vertex constraint

The following is a brief summary of the derivation in ref. [1]. We assume that the constraint equations can be linearized by expanding about convenient points $\boldsymbol{\alpha}_A$ and $\mathbf{x}_A$. Since the covariance matrix of the vertex may be known in advance (sometimes known as a “prior” covariance matrix), we write the overall χ^2 condition as

$$\chi^2 = (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0)^T \mathbf{V}_{\alpha 0}^{-1} (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0) + (\mathbf{x} - \mathbf{x}_0)^T \mathbf{V}_{x0}^{-1} (\mathbf{x} - \mathbf{x}_0) + 2\boldsymbol{\lambda}^T (\mathbf{D}\delta\boldsymbol{\alpha} + \mathbf{E}\delta\mathbf{x} + \mathbf{d}) \tag{6}$$

where $\delta\boldsymbol{\alpha} = \boldsymbol{\alpha} - \boldsymbol{\alpha}_A$ and $\delta\mathbf{x} = \mathbf{x} - \mathbf{x}_A$. The terms represent, respectively, the contribution to the χ^2 from the tracks, the vertex and the constraints. $\boldsymbol{\lambda}$ is the vector of Lagrange multipliers, one per constraint, while $\mathbf{x}_0$ and $\mathbf{V}_{\mathbf{x}0}$ are the initial vertex location and its covariance matrix. If the vertex location is unknown one can set its covariance matrix to a large value.

There are two constraint equations for each track. For example, in a solenoidal field

$$\begin{aligned} 0 &= p_{xi}\Delta y_i - p_{yi}\Delta x_i - \frac{a_{zi}}{2}\left(\Delta x_i^2 + \Delta y_i^2\right) \\ 0 &= \Delta z_i - \frac{p_{zi}}{a_{zi}} \sin^{-1}\left[a_{zi}\left(p_{xi}\Delta x_i + p_{yi}\Delta y_i\right) / \, p_{\perp i}^2\right] \end{aligned} \tag{7}$$

where $\Delta x_i = x_x - x_i$, etc. A more complicated pair of equations can be derived for *B* fields oriented along other directions [1]. Whatever the *B* field orientation, the **E** and **D** matrices have the simple form

$$\mathbf{E} = \begin{pmatrix} \mathbf{E}_1 \\ \mathbf{E}_2 \\ \vdots \\ \mathbf{E}_n \end{pmatrix} \qquad \mathbf{D} = \begin{pmatrix} \mathbf{D}_1 & & & \\ & \mathbf{D}_2 & & \\ & & \ddots & \\ & & & \mathbf{D}_n \end{pmatrix} \tag{8}$$

where each $\mathbf{E}_i$ is a 2×3 matrix and each $\mathbf{D}_i$ is a 2×7 matrix. Each row of these matrices corresponds to a single constraint.

To cast the χ^2 equation into a familiar form, we write it using the “extended” matrices

$$\tilde{\boldsymbol{\alpha}} = \begin{pmatrix} \mathbf{x} \\ \boldsymbol{\alpha}_1 \\ \vdots \\ \boldsymbol{\alpha}_n \end{pmatrix} \quad \mathbf{V}_{\tilde{\boldsymbol{\alpha}}0} = \begin{pmatrix} \mathbf{V}_{\mathbf{x}0} & & & \\ & \mathbf{V}_{01} & & \\ & & \ddots & \\ & & & \mathbf{V}_{0n} \end{pmatrix} \quad \tilde{\mathbf{D}} = \begin{pmatrix} \mathbf{E}_1 & \mathbf{D}_1 & & & \\ \mathbf{E}_2 & & \mathbf{D}_2 & & \\ \vdots & & & \ddots & \\ \mathbf{E}_n & & & & \mathbf{D}_n \end{pmatrix} \tag{9}$$

giving the new equation

$$\chi^2 = (\tilde{\boldsymbol{\alpha}} - \tilde{\boldsymbol{\alpha}}_0)^T \mathbf{V}_{\tilde{\alpha}0}^{-1} (\tilde{\boldsymbol{\alpha}} - \tilde{\boldsymbol{\alpha}}_0) + 2\boldsymbol{\lambda}^T \left(\tilde{\mathbf{D}}\delta\tilde{\boldsymbol{\alpha}} + \mathbf{d}\right) \tag{10}$$

The minimization of this χ^2 with respect to all the variables has the solution [1]:

$$
\begin{aligned}
\boldsymbol{\alpha} &= \boldsymbol{\alpha}_0 - \mathbf{V}_{\boldsymbol{\alpha}0}\mathbf{D}^T\boldsymbol{\lambda} \\
\mathbf{x} &= \mathbf{x}_0 - \mathbf{V}_{\mathbf{x}0}\mathbf{E}^T\boldsymbol{\lambda} \\
\boldsymbol{\lambda} &= \mathbf{V}_{\tilde{D}}\left(\mathbf{D}\delta\boldsymbol{\alpha}_0 + \mathbf{E}\delta\mathbf{x}_0 + \mathbf{d}\right) = \boldsymbol{\lambda}_0 + \mathbf{V}_D\mathbf{E}\delta\mathbf{x} \\
\boldsymbol{\lambda}_0 &= \mathbf{V}_D\left(\mathbf{D}\delta\boldsymbol{\alpha}_0 + \mathbf{d}\right) \\
\mathbf{V}_{\tilde{D}} &= \left(\mathbf{D}\mathbf{V}_{\boldsymbol{\alpha}0}\mathbf{D}^T + \mathbf{E}\mathbf{V}_{\mathbf{x}0}\mathbf{E}^T\right)^{-1} \\
&= \mathbf{V}_D - \mathbf{V}_D\mathbf{E}\mathbf{V}_{\mathbf{x}}\mathbf{E}^T\mathbf{V}_D
\end{aligned}
\tag{11}
$$

with the matrix $\mathbf{V}_D$ defined by

$$
\mathbf{V}_D = \begin{pmatrix} \mathbf{V}_{D1} & & & \\ & \mathbf{V}_{D2} & & \\ & & \ddots & \\ & & & \mathbf{V}_{Dn} \end{pmatrix} \qquad \mathbf{V}_{Di} = \left(\mathbf{D}_i\mathbf{V}_{0i}\mathbf{D}_i^T\right)^{-1}
\tag{12}
$$

The χ^2 is given by the simplex expression

$$
\begin{aligned}
\chi^2 &= \boldsymbol{\lambda}^T\mathbf{V}_{\tilde{D}}^{-1}\boldsymbol{\lambda} = \boldsymbol{\lambda}^T\left(\mathbf{V}_D^{-1} + \mathbf{E}\mathbf{V}_{\mathbf{x}0}\mathbf{E}^T\right)\boldsymbol{\lambda} \\
&= \boldsymbol{\lambda}^T\left(\mathbf{D}\delta\boldsymbol{\alpha}_0 + \mathbf{E}\delta\mathbf{x}_0 + \mathbf{d}\right)
\end{aligned}
\tag{13}
$$

The covariance matrices are

$$
\begin{aligned}
\mathbf{V}_{\mathbf{x}} &= \left(\mathbf{V}_{\mathbf{x}0}^{-1} + \mathbf{V}_E^{-1}\right)^{-1} \\
\mathbf{V}_{\boldsymbol{\alpha}} &= \mathbf{V}_{\boldsymbol{\alpha}0} - \mathbf{V}_{\boldsymbol{\alpha}0}\mathbf{D}^T\mathbf{V}_D\mathbf{D}\mathbf{V}_{\boldsymbol{\alpha}0} + \mathbf{V}_{\boldsymbol{\alpha}0}\mathbf{D}^T\mathbf{V}_D\mathbf{E}\mathbf{V}_{\mathbf{x}}\mathbf{E}^T\mathbf{V}_D\mathbf{D}\mathbf{V}_{\boldsymbol{\alpha}0} \\
\mathrm{cov}(\boldsymbol{\alpha},\mathbf{x}) &= -\mathbf{V}_{\boldsymbol{\alpha}0}\mathbf{D}^T\mathbf{V}_D\mathbf{E}\mathbf{V}_{\mathbf{x}}
\end{aligned}
\tag{14}
$$

Solution for the virtual particle parameters

We are now ready to construct the virtual particle parameters and covariance matrix. The parameters can be written in matrix form, following eqn. (5):

$$
\begin{aligned}
\mathbf{p}_V &= \mathbf{A}\boldsymbol{\alpha} + \mathbf{B}\mathbf{x} = \textstyle\sum_i \mathbf{A}_i\alpha_i + \mathbf{B}\mathbf{x} \\
\mathbf{x}_V &= \mathbf{x}
\end{aligned}
\tag{15}
$$

where

$$
\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 & \mathbf{A}_2 & \cdots & \mathbf{A}_n \end{pmatrix} \qquad \mathbf{B} = \begin{pmatrix} 0 & -\sum a_{zi} & \sum a_{yi} \\ \sum a_{zi} & 0 & -\sum a_{xi} \\ -\sum a_{yi} & \sum a_{xi} & 0 \\ 0 & 0 & 0 \end{pmatrix}
\tag{16}
$$

and

$$\mathbf{A}_i = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & a_{zi} & -a_{yi} \\ 0 & 1 & 0 & 0 & -a_{zi} & 0 & a_{xi} \\ 0 & 0 & 1 & 0 & a_{yi} & -a_{xi} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \tag{17}$$

This part is easy since the track parameters are determined from the vertex constraint above. The harder part is getting the covariance matrix right. It can be written as

$$\mathbf{V}_{\boldsymbol{\alpha}_V} = \begin{pmatrix} \mathbf{V}_{\mathbf{p}_V} & \operatorname{cov}(\mathbf{p}_V, \mathbf{x}_V) \\ \operatorname{cov}(\mathbf{x}_V, \mathbf{p}_V) & \mathbf{V}_{\mathbf{x}_V} \end{pmatrix} \tag{18}$$

From the definition of the virtual particle parameters in eqn. (15)

$$\begin{aligned} \mathbf{V}_{\mathbf{p}V} &= \mathbf{A}\mathbf{V}_{\boldsymbol{\alpha}}\mathbf{A}^T + \mathbf{A}\operatorname{cov}(\boldsymbol{\alpha}, \mathbf{x})\mathbf{B}^T + \mathbf{B}\operatorname{cov}(\mathbf{x}, \boldsymbol{\alpha})\mathbf{A}^T + \mathbf{B}\mathbf{V}_{\mathbf{x}}\mathbf{B}^T \\ \operatorname{cov}(\mathbf{p}_V, \mathbf{x}_V) &= \mathbf{A}\operatorname{cov}(\boldsymbol{\alpha}, \mathbf{x}) + \mathbf{B}\mathbf{V}_{\mathbf{x}} \\ \mathbf{V}_{\mathbf{x}_V} &= \mathbf{V}_{\mathbf{x}} \end{aligned} \tag{19}$$

Using the matrices in eqn. (14), we find the covariance matrix for the virtual particle

$$\mathbf{V}_{\boldsymbol{\alpha}_V} = \begin{pmatrix} \sum_i \left(\mathbf{A}_i \mathbf{V}_{0i} \mathbf{A}_i^T - \mathbf{S}_{1i} \mathbf{V}_{Di} \mathbf{S}_{1i}^T \right) + \mathbf{T}\mathbf{V}_{\mathbf{x}}\mathbf{T}^T & -\mathbf{T}\mathbf{V}_{\mathbf{x}} \\ -\mathbf{V}_{\mathbf{x}}\mathbf{T}^T & \mathbf{V}_{\mathbf{x}} \end{pmatrix} \tag{20}$$

where $\mathbf{S}_{1i} = \mathbf{A}_i \mathbf{V}_{0i} \mathbf{D}_i^T$, $\mathbf{S}_{3i} = \mathbf{A}_i \mathbf{V}_{0i} \mathbf{D}_i^T \mathbf{V}_{Di} \mathbf{E}_i$ and $\mathbf{T} = -\mathbf{B} + \sum_i \mathbf{S}_{3i}$.

The construction of the virtual particle, including track parameters and covariance matrix, is now complete. The particle can be used freely in subsequent fits with the understanding that when it is used in a fit, its daughters' track parameters will not be updated. To make it possible to update the daughter tracks when the virtual particle participates in a later fit, one would first have to use eqns. (11) and (14) to recalculate the daughter parameters and their covariance matrices, including the correlations between tracks introduced by the vertex constraint. This is not possible in the **KWFIT** package, which does not keep track of inter-track correlations.

III Building virtual particles when a subset of particles is verticized

Building virtual particles is more complicated than discussed above because we want the flexibility of merging different kinds of particles, some of which have poor or no position information and so cannot be easily used in a vertex constraint. I divide the particles into three classes:

1. Particles which are used to determine the vertex.
2. Particles which are forced to pass through the vertex determined by the first class, but which are not used to find the vertex. One might use soft pions from $D^{*+} \to D^0\pi^+$ in this way.

3. Particles which have no position information at all (such as π^0s and photons) which only contribute to the total 4-momentum.

It is necessary to re-derive the equations from the beginning to account for these changes. The overall χ^2 condition for the vertex is, as before,

$$\chi^2 = (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0)^T \mathbf{V}_{\alpha 0}^{-1} (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0) + (\mathbf{x} - \mathbf{x}_0)^T \mathbf{V}_{x0}^{-1} (\mathbf{x} - \mathbf{x}_0) + 2\boldsymbol{\lambda}^T (\mathbf{D}\delta\boldsymbol{\alpha} + \mathbf{E}\delta\mathbf{x} + \mathbf{d}) \qquad (21)$$

where this time only class 1 and class 2 particles are included. Care must be taken when taking partial derivatives, however. Classes 1 and 2 will both be used in the $\partial/\partial\boldsymbol{\alpha}_i$ partials, since both sets of particles must pass though the vertex, but only class 1 particles will participate in $\partial/\partial\mathbf{x}$ because they alone determine the vertex. Taking partials with respect to $\mathbf{x}$, $\boldsymbol{\alpha}$ and $\boldsymbol{\lambda}$ in this manner to minimize the χ^2 yields the following system of equations

$$\begin{aligned} \mathbf{V}_{\mathbf{x}0}^{-1}(\mathbf{x} - \mathbf{x}_0) + \mathbf{E}'^T\boldsymbol{\lambda} &= 0 \\ \mathbf{V}_{\boldsymbol{\alpha}0}^{-1}(\boldsymbol{\alpha} - \boldsymbol{\alpha}_0) + \mathbf{D}^T\boldsymbol{\lambda} &= 0 \\ \mathbf{D}\delta\boldsymbol{\alpha} + \mathbf{E}\delta\mathbf{x} + \mathbf{d} &= 0 \end{aligned} \qquad (22)$$

where $\mathbf{E}'_i = \mathbf{E}_i$ for class 1 particles and $\mathbf{E}'_i = 0$ for all others. Solving for the unknowns $\boldsymbol{\alpha}$ and $\mathbf{x}$ yields

$$\begin{aligned} \boldsymbol{\alpha} &= \boldsymbol{\alpha}_0 - \mathbf{V}_0\mathbf{D}^T\boldsymbol{\lambda} \\ \mathbf{x} &= \mathbf{x}_0 - \mathbf{V}_{\mathbf{x}0}\mathbf{E}'^T\boldsymbol{\lambda} \\ \boldsymbol{\lambda} &= \mathbf{V}_{\tilde{D}}(\mathbf{D}\delta\boldsymbol{\alpha}_0 + \mathbf{E}\delta\mathbf{x}_0 + \mathbf{d}) \\ \mathbf{V}_{\tilde{D}} &= \left(\mathbf{D}\mathbf{V}_{\boldsymbol{\alpha}0}\mathbf{D}^T + \mathbf{E}\mathbf{V}_{\mathbf{x}0}\mathbf{E}'^T\right)^{-1} \\ &= \mathbf{V}_D - \mathbf{V}_D\mathbf{E}\left(\mathbf{V}_{\mathbf{x}0}^{-1} + \mathbf{V}_{E'}^{-1}\right)^{-1}\mathbf{E}'^T\mathbf{V}_D \end{aligned} \qquad (23)$$

where $\mathbf{V}_{E'}^{-1} = \mathbf{E}'^T\mathbf{V}_D\mathbf{E} = \mathbf{E}'^T\mathbf{V}_D\mathbf{E}'$. This final equality follows from the definition of $\mathbf{E}'_i$. Also note that $\mathbf{V}_{\tilde{D}}$ *is no longer symmetric*, a fact which greatly complicates the matrix algebra.

After an absolutely incredible amount of matrix manipulation, the covariance matrices can be shown to reduce to the following forms

$$\begin{aligned} \mathbf{V}_{\mathbf{x}} &= \left(\mathbf{V}_{\mathbf{x}0}^{-1} + \mathbf{V}_{E'}^{-1}\right)^{-1} \\ \mathbf{V}_{\boldsymbol{\alpha}} &= \mathbf{V}_{\boldsymbol{\alpha}0} - \mathbf{V}_{\boldsymbol{\alpha}0}\mathbf{D}^T\mathbf{V}_D\mathbf{D}\mathbf{V}_{\boldsymbol{\alpha}0} + \mathbf{V}_{\boldsymbol{\alpha}0}\mathbf{D}^T\mathbf{V}_D\mathbf{E}\mathbf{V}_{\mathbf{x}}\mathbf{E}^T\mathbf{V}_D\mathbf{D}\mathbf{V}_{\boldsymbol{\alpha}0} \\ \mathrm{cov}(\boldsymbol{\alpha}, \mathbf{x}) &= -\mathbf{V}_{\boldsymbol{\alpha}0}\mathbf{D}^T\mathbf{V}_D\mathbf{E}\mathbf{V}_{\mathbf{x}} \end{aligned} \qquad (24)$$

which can be seen by inspection to be *identical* to those derived in the previous section, with the exception that $\mathbf{V}_{\mathbf{x}}$ is computed only from the class 1 tracks: $\mathbf{V}_{\mathbf{x}} = \left(\mathbf{V}_{\mathbf{x}0}^{-1} + \mathbf{V}_{E'}^{-1}\right)^{-1}$.

The calculation of the virtual particle's parameters and covariance matrix must include tracks from all classes, including those from class 3, which were not used in the vertex calculation above. We define the virtual particle parameters as before

$$\begin{aligned} \mathbf{p}_V &= \mathbf{A}\boldsymbol{\alpha} + \mathbf{B}\mathbf{x} = \sum\nolimits_i \mathbf{A}_i \alpha_i + \mathbf{B}\mathbf{x} \\ \mathbf{x}_V &= \mathbf{x} \end{aligned} \qquad (25)$$

where the sums now run over *all* particles and **A** and **B** are defined as before to be

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 & \mathbf{A}_2 & \cdots & \mathbf{A}_n \end{pmatrix} \qquad \mathbf{B} = \begin{pmatrix} 0 & -\sum a_{zi} & \sum a_{yi} \\ \sum a_{zi} & 0 & -\sum a_{xi} \\ -\sum a_{yi} & \sum a_{xi} & 0 \\ 0 & 0 & 0 \end{pmatrix} \qquad (26)$$

with

$$\mathbf{A}_i = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & a_{zi} & -a_{yi} \\ 0 & 1 & 0 & 0 & -a_{zi} & 0 & a_{xi} \\ 0 & 0 & 1 & 0 & a_{yi} & -a_{xi} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \qquad (27)$$

The covariance matrix is, by definition,

$$\mathbf{V}_{\boldsymbol{\alpha}_V} = \begin{pmatrix} \mathbf{V}_{\mathbf{p}_V} & \mathrm{cov}(\mathbf{p}_V, \mathbf{x}_V) \\ \mathrm{cov}(\mathbf{x}_V, \mathbf{p}_V) & \mathbf{V}_{\mathbf{x}_V} \end{pmatrix} \qquad (28)$$

where, as before,

$$\begin{aligned} \mathbf{V}_{\mathbf{p}V} &= \mathbf{A}\mathbf{V}_{\boldsymbol{\alpha}}\mathbf{A}^T + \mathbf{A}\,\mathrm{cov}(\boldsymbol{\alpha}, \mathbf{x})\mathbf{B}^T + \mathbf{B}\,\mathrm{cov}(\mathbf{x}, \boldsymbol{\alpha})\mathbf{A}^T + \mathbf{B}\mathbf{V}_{\mathbf{x}}\mathbf{B}^T \\ \mathrm{cov}(\mathbf{p}_V, \mathbf{x}_V) &= \mathbf{A}\,\mathrm{cov}(\boldsymbol{\alpha}, \mathbf{x}) + \mathbf{B}\mathbf{V}_{\mathbf{x}} \\ \mathbf{V}_{\mathbf{x}_V} &= \mathbf{V}_{\mathbf{x}} \end{aligned} \qquad (29)$$

Using (24) and collecting terms with the understanding that $\mathbf{D}_i = 0$ and $\mathbf{V}_{Di} = 0$ for class 3 particles, we find the covariance matrix for the virtual particle:

$$\mathbf{V}_{\boldsymbol{\alpha}_V} = \begin{pmatrix} \sum_i \left(\mathbf{A}_i \mathbf{V}_{0i} \mathbf{A}_i^T - \mathbf{S}_{1i} \mathbf{V}_{Di} \mathbf{S}_{1i}^T \right) + \mathbf{T}\mathbf{V}_{\mathbf{x}}\mathbf{T}^T & -\mathbf{T}\mathbf{V}_{\mathbf{x}} \\ -\mathbf{V}_{\mathbf{x}}\mathbf{T}^T & \mathbf{V}_{\mathbf{x}} \end{pmatrix} \qquad (30)$$

where $\mathbf{S}_{1i} = \mathbf{A}_i \mathbf{V}_{0i} \mathbf{D}_i^T$, $\mathbf{S}_{3i} = \mathbf{A}_i \mathbf{V}_{0i} \mathbf{D}_i^T \mathbf{V}_{Di} \mathbf{E}_i$ and $\mathbf{T} = -\mathbf{B} + \sum_i \mathbf{S}_{3i}$. This is the identical result as before, provided we use the proper values for the matrices. This algorithm is implemented in **KWFIT**.

IV Building virtual particles incorporating additional constraints

Frequently we want to build virtual particles with the vertex constraint augmented by, say, a mass constraint. The incorporation of additional constraints is not difficult at a fundamental level. However, the equations derived in the last section do not work because the extra constraints do not have the convenient block diagonal structure. Although one can include the extra constraints simultaneously in the fit (see Appendix), it is far easier to adopt the two step method described in Section V in ref. [1].

The strategy is to first build a virtual particle, using the standard vertex fit, and compute the χ^2. The additional constraints are then applied to the virtual particle itself, using one of the constraints described in ref. [3]. The track parameters and covariance matrix are updated and a new χ^2 is computed, which is added to the first one to form the total χ^2. If the additional constraints are such that they apply to the virtual particle parameters as a whole, then this two step procedure is *mathematically identical* to utilizing all the constraints in one simultaneous fit, as proven in ref. [1].

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.

Appendix

Directly Incorporating Additional Constraints in Vertex Fitting

We frequently want to build a virtual particle where a mass constraint is imposed in addition to the vertex conditions. I argued in Section IV that the best way to handle this case is to build the virtual particle first with only the vertex constraint and add the additional constraint afterwards. This method is identical to putting all the constraints together and turning the crank, and offers the additional advantage of returning two χ^2 values. However, for reasons of mathematical completeness I decided to include here an algorithm for simultaneously (as opposed to consecutively) adding additional constraints when building a virtual particle.

There is no fundamental problem in adding *m* additional constraints to the vertex fitting problem. The straightforward solution was worked out in ref. [2] using the matrix **D** to express the constraints. For *n* tracks, we have a problem with 2*n* + *m* total constraints, thus a $(2n+m)\times(2n+m)$ matrix must be inverted. However, I show here an alternative method which is computationally faster and only involves the inversion of an additional $m \times m$ matrix.

Again we expand the constraint equations about convenient values $\boldsymbol{\alpha}_A$ and $\mathbf{x}_A$. As in the vertex only case, the constraint conditions are denoted $\mathbf{H}(\boldsymbol{\alpha},\mathbf{x}) = \mathbf{0}$, where **H** is a vector of length $m+2n$, and the non-vertex constraints are assumed to come first.

The χ^2 condition including all the constraints can be written in the standard linearized form:

$$\chi^2 = (\boldsymbol{\alpha}-\boldsymbol{\alpha}_0)^T \mathbf{V}_{\alpha 0}^{-1}(\boldsymbol{\alpha}-\boldsymbol{\alpha}_0) + (\mathbf{x}-\mathbf{x}_0)^T \mathbf{V}_{x0}^{-1}(\mathbf{x}-\mathbf{x}_0) + 2\boldsymbol{\lambda}^T(\mathbf{G}\delta\boldsymbol{\alpha} + \mathbf{F}\delta\mathbf{x} + \mathbf{g}) \qquad (31)$$

where $\delta\boldsymbol{\alpha} = \boldsymbol{\alpha} - \boldsymbol{\alpha}_A$, $\delta\mathbf{x} = \mathbf{x} - \mathbf{x}_A$. **F** and **G** are matrices representing the partial derivatives of **H** with respect to the vertex and track parameters, respectively, while $\mathbf{g} = \mathbf{H}(\boldsymbol{\alpha}_A, \mathbf{x}_A)$ is a column vector representing the values of the constraints at the expansion point. These matrices can be written

$$\mathbf{F} = \begin{pmatrix} \mathbf{N}_x \\ \mathbf{E}_1 \\ \mathbf{E}_2 \\ \vdots \\ \mathbf{E}_n \end{pmatrix} \qquad \mathbf{G} = \begin{pmatrix} \mathbf{N}_{\alpha 1} & \mathbf{N}_{\alpha 2} & \cdots & \mathbf{N}_{\alpha n} \\ \mathbf{D}_1 & & & \\ & \mathbf{D}_2 & & \\ & & \ddots & \\ & & & \mathbf{D}_n \end{pmatrix} \qquad (32)$$

The **D** and **E** matrices are the partials of **H** (the vertex constraint portion) with respect to the track and vertex parameters, respectively, as described in Section II. The **N** matrices are the partial derivatives of **H** for the non-vertex constraints. Note that I include an $\mathbf{N}_x$ term, allowing the extra constraints to depend on the vertex position. The dimensions of these matrices are tabulated below.

$\mathbf{E}_i$	2×3
$\mathbf{D}_i$	2×7

$\mathbf{N}_x$	$m \times 3$
$\mathbf{N}_{\alpha i}$	$m \times 7$
$\mathbf{g}$	$(m + 2n) \times 1$
$\mathbf{F}$	$(m + 2n) \times 3$
$\mathbf{G}$	$(m + 2n) \times 7n$

Again we proceed by defining extended matrices to incorporate vertex and track information, e.g.,

$$\tilde{\boldsymbol{\alpha}} = \begin{pmatrix} \mathbf{x} \\ \boldsymbol{\alpha}_1 \\ \vdots \\ \boldsymbol{\alpha}_n \end{pmatrix} \quad \mathbf{V}_{\tilde{\alpha}0} = \begin{pmatrix} \mathbf{V}_{\mathbf{x}0} & & & \\ & \mathbf{V}_{01} & & \\ & & \ddots & \\ & & & \mathbf{V}_{0n} \end{pmatrix} \quad \tilde{\mathbf{G}} = \begin{pmatrix} \mathbf{N}_x & \mathbf{N}_{\alpha 1} & \mathbf{N}_{\alpha 2} & \cdots & \mathbf{N}_{\alpha n} \\ \mathbf{E}_1 & \mathbf{D}_1 & & & \\ \mathbf{E}_2 & & \mathbf{D}_2 & & \\ \vdots & & & \ddots & \\ \mathbf{E}_n & & & & \mathbf{D}_n \end{pmatrix} \tag{33}$$

The new χ^2 equation can be written in these external variables as

$$\chi^2 = (\tilde{\boldsymbol{\alpha}} - \tilde{\boldsymbol{\alpha}}_0)^T \mathbf{V}_{\tilde{\alpha}0}^{-1} (\tilde{\boldsymbol{\alpha}} - \tilde{\boldsymbol{\alpha}}_0) + 2\boldsymbol{\lambda}^T \left(\tilde{\mathbf{G}} \delta \tilde{\boldsymbol{\alpha}} + \mathbf{g} \right) \tag{34}$$

Minimizing this with respect to $\tilde{\boldsymbol{\alpha}}$ and $\boldsymbol{\lambda}$ yields the solution [1]:

$$\begin{aligned} \tilde{\boldsymbol{\alpha}} &= \tilde{\boldsymbol{\alpha}}_0 - \mathbf{V}_{\tilde{\alpha}_0} \tilde{\mathbf{G}}^T \boldsymbol{\lambda} \\ \boldsymbol{\lambda} &= \mathbf{V}_{\tilde{G}} \left(\tilde{\mathbf{G}} \delta \tilde{\boldsymbol{\alpha}}_0 + \mathbf{d} \right) \\ \mathbf{V}_{\tilde{G}} &= \left(\tilde{\mathbf{G}} \mathbf{V}_{\tilde{\alpha}_0} \tilde{\mathbf{G}}^T \right)^{-1} \\ \mathbf{V}_{\tilde{\alpha}} &= \mathbf{V}_{\tilde{\alpha}0} - \mathbf{V}_{\tilde{\alpha}0} \tilde{\mathbf{G}}^T \mathbf{V}_{\tilde{G}} \tilde{\mathbf{G}} \mathbf{V}_{\tilde{\alpha}0} \\ \chi^2 &= \boldsymbol{\lambda}^T \mathbf{V}_{\tilde{G}}^{-1} \boldsymbol{\lambda} = \boldsymbol{\lambda}^T \left(\tilde{\mathbf{G}} \delta \tilde{\boldsymbol{\alpha}}_0 + \mathbf{g} \right) \end{aligned} \tag{35}$$

Implementing this solution requires that we invert a matrix of size $(2n + m) \times (2n + m)$. For 3 tracks with a mass constraint, the matrix is 7×7, which is starting to get a bit large. However, it is possible to simplify the problem. We compute $\mathbf{V}_{\tilde{G}}$ to be

$$\begin{aligned} \mathbf{V}_{\tilde{G}} &= \begin{pmatrix} \mathbf{N}_x \mathbf{V}_{\mathbf{x}0} \mathbf{N}_x^T + \sum_i \mathbf{N}_{\alpha i} \mathbf{V}_{0i} \mathbf{N}_{\alpha i}^T & \mathbf{N}_x \mathbf{V}_{\mathbf{x}0} \mathbf{E}_1^T + \mathbf{N}_{\alpha 1} \mathbf{V}_{01} \mathbf{D}_1^T & \cdots & \mathbf{N}_x \mathbf{V}_{\mathbf{x}0} \mathbf{E}_n^T + \mathbf{N}_{\alpha n} \mathbf{V}_{0n} \mathbf{D}_n^T \\ \mathbf{E}_1 \mathbf{V}_{\mathbf{x}0} \mathbf{N}_x^T + \mathbf{D}_1 \mathbf{V}_{01} \mathbf{N}_{\alpha 1}^T & \mathbf{E}_1 \mathbf{V}_{\mathbf{x}0} \mathbf{E}_1^T + \mathbf{V}_{D1}^{-1} & \cdots & \mathbf{E}_1 \mathbf{V}_{\mathbf{x}0} \mathbf{E}_n^T \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{E}_n \mathbf{V}_{\mathbf{x}0} \mathbf{N}_0^T + \mathbf{D}_n \mathbf{V}_{0n} \mathbf{N}_{\alpha n}^T & \mathbf{E}_n \mathbf{V}_{\mathbf{x}0} \mathbf{E}_1^T & \cdots & \mathbf{E}_n \mathbf{V}_{\mathbf{x}0} \mathbf{E}_n^T + \mathbf{V}_{Dn}^{-1} \end{pmatrix}^{-1} \\ &= \begin{pmatrix} \mathbf{N} \mathbf{V}_{\tilde{\alpha}0} \mathbf{N}^T & \mathbf{N} \mathbf{V}_{\tilde{\alpha}0} \tilde{\mathbf{D}}^T \\ \tilde{\mathbf{D}} \mathbf{V}_{\tilde{\alpha}0} \mathbf{N}^T & \mathbf{V}_{\tilde{D}}^{-1} \end{pmatrix}^{-1} \end{aligned} \tag{36}$$

where $\mathbf{V}_{Di}^{-1} = \mathbf{D}_i \mathbf{V}_{0i} \mathbf{D}_i^T$ are 2×2 matrices and $\mathbf{N} = \begin{pmatrix} \mathbf{N}_x & \mathbf{N}_{\alpha 1} & \mathbf{N}_{\alpha 2} & \cdots & \mathbf{N}_{\alpha n} \end{pmatrix}$.

Step aside and watch me work. To compute $\mathbf{V}_{\tilde{G}}$ I use a matrix inversion technique derived in the Appendix in ref. [2]. Consider a large matrix **V** which is split into a 2 × 2 block. Then it is easy to prove that

$$\mathbf{V}^{-1} = \begin{pmatrix} V_{11} & V_{12} \\ V_{12}^T & V_{22} \end{pmatrix}^{-1} = \begin{pmatrix} S^{-1} & -S^{-1}V_{12}V_{22}^{-1} \\ -V_{22}^{-1}V_{12}^T S^{-1} & V_{22}^{-1} + V_{22}^{-1}V_{12}^T S^{-1} V_{12} V_{22}^{-1} \end{pmatrix} \tag{37}$$

where $S = V_{11} - V_{12}V_{22}^{-1}V_{12}^T$ and V_{22} is invertible. You can verify this solution by multiplying the matrices together. The trick is to make the dimensions of V_{11} as small as possible while keeping V_{22} invertible.

The lower block of $\mathbf{V}_{\tilde{G}}$ is just $\mathbf{V}_{\tilde{D}}^{-1}$, which we inverted in eqn. (11), so we only need to calculate an additional $m \times m$ matrix inversion. This yields, using the above theorem,

$$\mathbf{V}_{\tilde{G}} = \begin{pmatrix} \mathbf{V}_N & -\mathbf{V}_N \mathbf{N} \mathbf{V}_{\tilde{\alpha}0} \tilde{\mathbf{D}}^T \mathbf{V}_{\tilde{D}} \\ -\mathbf{V}_{\tilde{D}} \tilde{\mathbf{D}} \mathbf{V}_{\tilde{\alpha}0} \mathbf{N}^T \mathbf{V}_N & \mathbf{V}_{\tilde{D}} + \mathbf{V}_{\tilde{D}} \tilde{\mathbf{D}} \mathbf{V}_{\tilde{\alpha}0} \mathbf{N}^T \mathbf{V}_N \mathbf{N} \mathbf{V}_{\tilde{\alpha}0} \tilde{\mathbf{D}}^T \mathbf{V}_{\tilde{D}} \end{pmatrix} \tag{38}$$

where $\mathbf{V}_N^{-1} = \mathbf{N}\mathbf{V}_{\tilde{\alpha}0}\mathbf{N}^T - \mathbf{N}\mathbf{V}_{\tilde{\alpha}0}\tilde{\mathbf{D}}^T\mathbf{V}_{\tilde{D}}\tilde{\mathbf{D}}\mathbf{V}_{\tilde{\alpha}0}\mathbf{N}^T$ is the new $m \times m$ matrix.

Equations (34) and (37) constitute the solution to the vertex problem plus additional constraints. It requires only one $m \times m$ matrix inverse in addition to what is required for the vertex constraints. However, the calculation of the covariance matrix for the new particle (see eqns. 17 and 18) is much more complicated than before, mostly because $\mathbf{V}_{\tilde{G}}$ cannot be expressed in a simple way. For now, I see no way of easily computing the covariance matrix, even if the form of **N** is simplified (say by containing no vertex component). For this reason I vastly prefer using the two step approach outlined in Section IV.

References

1. Paul Avery, “Applied Fitting Theory VI: Formulas for Kinematic Fitting”, CBX 97–, http://www.phys.ufl.edu/~avery/fitting.html.
2. Paul Avery, “Applied Fitting Theory I: General Least Squares Theory”, CBX 91–72, http://www.phys.ufl.edu/~avery/fitting.html.