Vertexing and Kinematic Fitting, Part III: Vertex Fitting

Lectures Given at SLAC
Aug. 5 – 7, 1998

Paul Avery
University of Florida

avery@phys.ufl.edu
http://www.phys.ufl.edu/~avery

Overview of plan

- **3rd of several lectures on kinematic fitting**
- **Focus in this lecture on vertex fitting theory**
- **Plan of lectures**
  - Lecture 1: Basic theory
  - Lecture 2: Introduction to the KWFIT fitting package
  - Lecture 3: Vertex fitting
  - Lecture 4: Building virtual particles
- **References**
  - **KWFIT**
    http://www.phys.ufl.edu/~avery/kwfit/ or
    http://w4.lns.cornell.edu/~avery/kwfit/
  - Several write-ups on fitting theory and constraints
    http://www.phys.ufl.edu/~avery/fitting.html

Theory

Equations of motion in solenoidal field

Written as function of arc length *s*, the path of the particle is a helix:

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

where $a = -0.299792458BQ$ and $\rho = a / p$.

Creating the constraint equations

Suppose we want to force n tracks to come from a common space point $\mathbf{x}$. Assume further that the vertex has some "prior information", i.e., it is a beam spot at $\mathbf{x}_0$ with covariance matrix $\mathbf{V}_{\mathbf{x}0}$. (If the vertex is completely unknown, we can just set the diagonal elements to large values.)

The condition that track i pass through the vertex generates 2 constraints: (1) r–ϕ and (2) z

$$0 = p_{xi}\Delta y_i - p_{yi}\Delta x_i - \frac{a_i}{2}\left(\Delta x_i^2 + \Delta y_i^2\right)$$

$$0 = \Delta z_i - \frac{p_{zi}}{a_i}\sin^{-1}\left[a_i\left(p_{xi}\Delta x_i + p_{yi}\Delta y_i\right) / \left(p_{xi}^2 + p_{yi}^2\right)\right]$$

where $a_i = -0.299792458 B Q_i$

Q_i = charge

$\Delta x_i = x_x - x_i$, etc.

For n tracks, there are $7n$ parameters and $2n$ constraints

The χ^2 includes contributions from the track parameters $\boldsymbol{\alpha}$**,** vertex parameters $\mathbf{x}$ and the constraints $\mathbf{H}(\boldsymbol{\alpha},\mathbf{x})=0$:

$$\begin{aligned}\chi^2 = &(\boldsymbol{\alpha}-\boldsymbol{\alpha}_0)^T \mathbf{V}_{\alpha 0}^{-1}(\boldsymbol{\alpha}-\boldsymbol{\alpha}_0) && \longleftarrow \boxed{\text{Tracks}} \\ &+(\mathbf{x}-\mathbf{x}_0)^T \mathbf{V}_{\mathbf{x}0}^{-1}(\mathbf{x}-\mathbf{x}_0) && \longleftarrow \boxed{\text{Vertex}} \\ &+2\boldsymbol{\lambda}^T(\mathbf{D}\delta\boldsymbol{\alpha}+\mathbf{E}\delta\mathbf{x}+\mathbf{d}) && \longleftarrow \boxed{\text{Constraints}}\end{aligned}$$

$\boldsymbol{\alpha}$ has length $7n$
$\mathbf{x}$ has length 3
$\boldsymbol{\lambda}$ has length $2n$

$\delta\boldsymbol{\alpha}=\boldsymbol{\alpha}-\boldsymbol{\alpha}_A$ (expanded around $\boldsymbol{\alpha}_A$)
$\delta\mathbf{x}=\mathbf{x}-\mathbf{x}_A$ (expanded around $\mathbf{x}_A$)

$\mathbf{D}$ is $2n\times 7n$ (coefficient of track parameters $D_{ij}=\partial H_i/\partial\alpha_j$)
$\mathbf{E}$ is $2n\times 3$ (coefficient of vertex parameters $E_{ij}=\partial H_i/\partial x_j$)
$\mathbf{d}$ is $2n\times 1$ (constant term $d_i=H_i(\boldsymbol{\alpha}_A,\mathbf{x}_A)$)

The crucial fact about the vertex constraint is that the constraints *do not mix tracks*.

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

This fact vastly speeds up the calculation for the solution because the matrices that need to be inverted can be reduced to block diagonal form.

$$\mathbf{V}_D = \left(\mathbf{D}\mathbf{V}_{\alpha 0}\mathbf{D}^T\right)^{-1} = \begin{pmatrix} D_1 V_{10} D_1^T & & & \\ & D_2 V_{20} D_2^T & & \\ & & \ddots & \\ & & & D_n V_{n0} D_n^T \end{pmatrix}^{-1}$$

2×2

The solution for α and $\mathbf{x}$ can be written (see CBX 98–37 for a detailed discussion)

$$\begin{aligned}
\delta\alpha &= \delta\alpha_0 - \mathbf{V}_{\alpha 0}\mathbf{D}^T\lambda \\
\delta\mathbf{x} &= \delta\mathbf{x}_0 - \mathbf{V}_{\mathbf{x}0}\mathbf{E}^T\lambda = \mathbf{V}_{\mathbf{x}}\mathbf{V}_{\mathbf{x}0}^{-1}\delta\mathbf{x}_0 - \mathbf{V}_{\mathbf{x}}\mathbf{E}^T\lambda_0 \\
\lambda &= \lambda_0 + \mathbf{V}_D\mathbf{E}\,\delta\mathbf{x} \\
\lambda_0 &= \mathbf{V}_D(\mathbf{D}\,\delta\alpha_0 + \mathbf{d}) \\
\mathbf{V}_D &= \left(\mathbf{D}\mathbf{V}_{\alpha 0}\mathbf{D}^T\right)^{-1}
\end{aligned}$$

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

The covariance matrices are

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

and the χ^2 is given by

$$\chi^2 = \lambda^T\left(\mathbf{V}_D^{-1} + \mathbf{E}\mathbf{V}_{\mathbf{x}0}^{-1}\mathbf{E}^T\right)\lambda = \lambda^T(\mathbf{D}\,\delta\alpha_0 + \mathbf{E}\,\delta\mathbf{x}_0 + \mathbf{d})$$

This solution requires that we invert the following

$\mathbf{V}_D = \left(\mathbf{D}\mathbf{V}_{\alpha 0}\mathbf{D}^T\right)^{-1}$ n 2×2 matrices

$\mathbf{V}_\mathbf{x} = \left(\mathbf{V}_{\mathbf{x}0} + \mathbf{E}^T\mathbf{V}_D\mathbf{E}\right)^{-1}$ a 3×3 matrix

$$\mathbf{V}_D = \left(\mathbf{D}\mathbf{V}_{\alpha 0}\mathbf{D}^T\right)^{-1} = \begin{pmatrix} D_1 V_{10} D_1^T & & & \\ & D_2 V_{20} D_2^T & & \\ & & \ddots & \\ & & & D_n V_{n0} D_n^T \end{pmatrix}^{-1}$$

$$= \begin{pmatrix} V_{D1} & & & \\ & V_{D2} & & \\ & & \ddots & \\ & & & V_{Dn} \end{pmatrix}$$

So we have found an efficient solution.

Comments About Solution

Vertex covariance matrix

The vertex covariance matrix $\mathbf{V_x}$ is an average of original covariance matrix and a sum over track info

$$\mathbf{V_x} = \left(\mathbf{V}_{\mathbf{x}0}^{-1} + \mathbf{E}^T \mathbf{V}_D \mathbf{E}\right)^{-1}$$

$$\mathbf{E} = \begin{pmatrix} E_1 \\ E_2 \\ \vdots \\ E_n \end{pmatrix} \qquad \mathbf{V}_D = \begin{pmatrix} V_{D1} & & & \\ & V_{D2} & & \\ & & \ddots & \\ & & & V_{Dn} \end{pmatrix}$$

$$\mathbf{V_x} = \left(\mathbf{V}_{\mathbf{x}0}^{-1} + \sum_i E_i^T V_{Di} E_i\right)^{-1}$$

Note that when the initial vertex is totally unknown, we make $\mathbf{V}_{\mathbf{x}0}$ as large as possible, so that $\mathbf{V}_{\mathbf{x}0}^{-1} \to 0$. In that case

$$\mathbf{V_x} = \left(\sum_i E_i^T V_{Di} E_i\right)^{-1}$$

Track correlations

Note the structure of the updated track covariance matrix

$$\mathbf{V}_\alpha = \mathbf{V}_{\alpha 0} - \mathbf{V}_{\alpha 0}\mathbf{D}^T\mathbf{V}_D\mathbf{D}\mathbf{V}_{\alpha 0} + \mathbf{V}_{\alpha 0}\mathbf{D}^T\mathbf{V}_D\mathbf{E}\mathbf{V}_\mathbf{x}\mathbf{E}^T\mathbf{V}_D\mathbf{D}\mathbf{V}_{\alpha 0}$$

Improvement due to fit to fixed vertex point.

Worsening due to “wiggle” of vertex point.

Term 2 does not cause track - track correlations, since $\mathbf{V}_{\alpha 0}$, $\mathbf{D}$ and $\mathbf{V}_D$ are block diagonal. Equivalent to fitting each track separately though the same fixed space point.

Term 3 causes track-track correlations *only* through the “tiny” 3×3 $\mathbf{V}_\mathbf{x}$ matrix. Track-track correlations can thus be calculated by saving some of the intermediate matrices.

Vertex Fitting as a Kalman Problem

Track fitting problem

Want to find 5 helix parameters α and 5×5 covariance matrix V_α by adding in the information from n measurements.

Method is to move to measurement point, then average the measurement (usually a 1 dim. quantity) with the track to get improved track parameters and covariance matrix. This is repeated until all measurements are included.

- Let α_0, $V_{\alpha 0}$ be the old track parameters & covariance matrix.
- Let Δy, V_y be the measurement deviation & its covariance matrix.
- Let A represent the derivative of the measurement with each of the 5 track parameters.

Then the new track parameters and covariance matrix are

$$\delta\alpha = \delta\alpha_0 + V_\alpha A^T V_y^{-1} \Delta y$$

$$\begin{aligned} V_\alpha &= \left(V_{\alpha 0}^{-1} + A^T V_y^{-1} A\right)^{-1} \\ &= V_{\alpha 0} - V_{\alpha 0} A^T \left(V_y + A V_{\alpha 0} A^T\right)^{-1} A V_{\alpha 0} \end{aligned}$$

Note that only a 1×1 matrix has to be inverted at each step.

If the track does not have to be moved or modified between measurements, then all the measurements can be added at once, at the cost of inverting an $n \times n$ matrix.

Vertex fitting

Want to find 3 vertex parameters $\mathbf{x}$ and 3×3 covariance matrix $\mathbf{V_x}$ by adding information from n tracks.

Each track "measurement" consists of 7 parameters α_0 and a 7×7 covariance matrix $V_{\alpha 0}$

Method is to average the vertex with the track to get improved vertex parameters and covariance matrix. This is repeated until all tracks are included.

The only difference from the track fitting problem is that a constraint is used to average the track info with the vertex information.

Let the index i refer to track i. Assume we have a vertex $\mathbf{x}_0$ with initial covariance $\mathbf{V}_{\mathbf{x}0}$ to which we want to add track i. The following equations update the vertex parameters, covariance matrix and χ^2.

$$\delta\mathbf{x} = \mathbf{V}_{\mathbf{x}}\left(\mathbf{V}_{\mathbf{x}0}^{-1}\delta\mathbf{x}_0 - E_i^T V_{Di}\beta_{0i}\right)$$

$$\mathbf{V}_{\mathbf{x}} = \left(\mathbf{V}_{\mathbf{x}0}^{-1} + E_i^T V_{Di} E_i\right)^{-1}$$

$$\Delta\chi_i^2 = \left(\beta_{0i} + E_i\delta\mathbf{x}\right)^T V_{Di}\left(\beta_{0i} + E\delta\mathbf{x}_0\right)$$

where $\beta_{0i} = D_i\delta\alpha_{0i} + d_i$. Repeat until all tracks are added.

It is also possible to make a quick algorithm where tracks can be included or discarded. Only 3 sums needed:

1. Vector of length 3 $\quad \sum_i E_i^T V_{Di}\beta_{0i}$
2. 3×3 matrix $\quad \sum_i E_i^T V_{Di} E_i$
3. Scalar $\quad \sum_i \Delta\chi_i^2$ as described above

A track can be removed easily from the sums if its $\Delta\chi^2$ contribution is too large. This process is similar to that used for removing hits from a Kalman fit.