

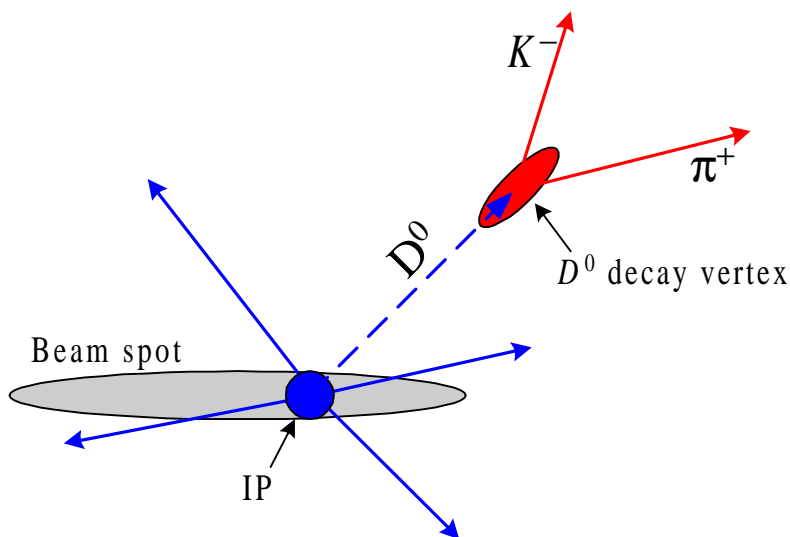
## Directly Determining Lifetime Using a 3-D Fit

I derive in this note a fitting algorithm that directly determines the proper lifetime and error of a single particle decay using all track and beam information, not just the information in the y coordinate. The method works for charged or neutral particles of arbitrarily long lifetime and has been implemented in **KWFIT**. I demonstrate through high statistics Monte Carlo studies that the method yields correct results and is unbiased to a high degree.

I do not address in this note the problem of obtaining an accurate starting point when tracks from a recoiling long-lived particle (i.e., anti-charm) are included in the fit. This problem and its potential resolution will be discussed in a separate note.

### I Introduction

To introduce the subject of lifetime determination, consider the diagram below, which shows a  $D^0$  decaying to  $K^-\pi^+$  at a secondary vertex after being produced in the beam interaction region.



In this example the  $D^0$  is produced at a point near the beam center and travels a certain distance (the “flight distance”) before decaying into its  $K^-\pi^+$  daughters. These daughters are subsequently measured by the tracking system.

An accurate determination of the lifetime requires that both the beginning and endpoint of the  $D^0$  flight vector be determined accurately. The endpoint is determined by vertex fitting with a package like **KWFIT**, and its measurement accuracy is determined purely by the tracking errors of the daughter particles. The beginning point is determined by the beam spot size augmented perhaps, as shown above, by other tracks produced in association with the  $D^0$ . The lifetime as determined in CLEO typically only uses the  $y$  portion of the flight vector because the CESR vertex profile ( $330 \mu\text{m} \times 7 \mu\text{m} \times 1.7 \text{ cm}$ ) provides an extremely small uncertainty in  $y$ . Although the  $x$  and  $z$  measurements can be improved by including extra tracks emerging from the collision point, the presence of tracks from anti-charm particles (which also travel a finite distance) can cause a bias in the starting point location so they are frequently not included in high precision lifetime determinations. The 4-momentum errors of the  $D^0$  are ignored, generally because they are felt to play a negligible role for short lifetimes. I address this point later in the paper.

## II Proper lifetime determination for simple cases

### *Calculation of proper lifetime with one coordinate*

Let's see how the lifetime can be determined from the  $y$  information for the  $D^0 \rightarrow K^-\pi^+$  case. Since the flight path is a straight line, the equation of motion of the particle is

$$y_p = y_d - sp_y / p, \quad (1)$$

where  $y_p$  and  $y_d$  are the  $y$  coordinates of the production and decay points, respectively,  $p$  and  $p_y$  are the total and  $y$  component of momentum at the decay point, respectively, and  $s$  is the flight distance. Since we are trying to determine the proper time and not the flight distance, we use the relativistic formula  $s = \beta ct = \gamma \beta c\tau = (p/m)c\tau$ , where  $c\tau$  is the proper time measured in units of distance and  $m$  is the mass. Substituting this expression into eqn. (1) yields  $y_p = y_d - c\tau p_y / m$ , or

$$c\tau = \frac{m}{p_y} (y_d - y_p) \quad (2)$$

The error in  $c\tau$  is easily calculated by simple propagation of errors to be

$$\sigma_{c\tau} = \frac{m}{p_y} \sigma_{\Delta y} \quad (3)$$

where  $\sigma_{\Delta y}^2 = \sigma_{y_d}^2 + \sigma_{y_p}^2$ , assuming that  $y_d$  and  $y_p$  are independent and neglecting contributions from momentum and mass uncertainties.

### *Proper lifetime determination with two uncorrelated coordinates*

Suppose we now include the information from both the  $x$  and  $y$  coordinates in the calculation of the lifetime. In this case the equations relating the production point and decay point, using obvious notation, are

$$\begin{aligned}x_p &= x_d - c\tau p_x / m \\y_p &= y_d - c\tau p_y / m\end{aligned}\tag{4}$$

We can solve for  $c\tau$  naively by multiply the  $x$  and  $y$  equations by  $p_x$  and  $p_y$ , respectively, and adding. This yields

$$c\tau = \frac{m}{p_{\perp}^2} \left[ p_x (x_d - x_p) + p_y (y_d - y_p) \right]\tag{5}$$

with error

$$\sigma_{c\tau} = \frac{m}{p_{\perp}^2} \sqrt{p_x^2 \sigma_{\Delta x}^2 + p_y^2 \sigma_{\Delta y}^2}\tag{6}$$

where  $\sigma_{\Delta x}^2 = \sigma_{x_d}^2 + \sigma_{x_p}^2$ , again assuming that the measurements of the production and decay points are not correlated.

Equation (6) is utterly, gloriously *wrong*! Well, not exactly wrong, but not optimal. Compare equation (3) with equation (6). If  $\sigma_{\Delta x}$  happens to be huge, it will dominate the error in  $c\tau$  irrespective of the  $y$  contribution. This makes no sense from the point of view of information theory, since adding new information (the  $x$  coordinate) should yield an error no larger than what was obtained by ignoring it altogether.

The reason why the error calculation in eqn. (6) is not optimal is that eqn. (5) is not the only solution for  $c\tau$  in eqns. (4). The general unbiased solution (unbiased means that averaging over an infinite number of experiments gives the correct answer) is

$$c\tau = Am \frac{(x_d - x_p)}{p_x} + (1 - A)m \frac{(y_d - y_p)}{p_y}\tag{7}$$

where  $A$  is not determined. The standard way of specifying  $A$  is to choose a value that minimizes the error on  $c\tau$ . This procedure is straightforward and yields

$$A = \frac{(p_x^2 / m^2) / \sigma_{\Delta x}^2}{(p_x^2 / m^2) / \sigma_{\Delta x}^2 + (p_y^2 / m^2) / \sigma_{\Delta y}^2}\tag{8}$$

where  $\sigma_{\Delta x}$  and  $\sigma_{\Delta y}$  are defined as before. The error for  $c\tau$  is now easily determined to be

$$\sigma_{c\tau}^2 = \frac{1}{(p_x^2 / m^2) / \sigma_{\Delta x}^2 + (p_y^2 / m^2) / \sigma_{\Delta y}^2}\tag{9}$$

The same result is obtained by using a more general analysis discussed in the next section.

Equation (9) includes all the information in an optimal way, and is easily seen to have the correct limiting behavior. For example, ignoring the  $x$  information in eqn. (9) ( $\sigma_{\Delta x} = \infty$ ) reproduces eqn. (3).

### Proper lifetime lifetime with three uncorrelated coordinates

Extending the above result to three dimensions (assuming that the  $x$ ,  $y$  and  $z$  variables are uncorrelated and that the momentum errors can be neglected) is trivial and yields the solution

$$c\tau = \frac{p_x}{m} \frac{\sigma_{c\tau}^2}{\sigma_{\Delta x}^2} (x_d - x_p) + \frac{p_y}{m} \frac{\sigma_{c\tau}^2}{\sigma_{\Delta y}^2} (y_d - y_p) + \frac{p_z}{m} \frac{\sigma_{c\tau}^2}{\sigma_{\Delta z}^2} (z_d - z_p)$$

$$\sigma_{c\tau}^2 = \frac{1}{\left(p_x^2/m^2\right)/\sigma_{\Delta x}^2 + \left(p_y^2/m^2\right)/\sigma_{\Delta y}^2 + \left(p_z^2/m^2\right)/\sigma_{\Delta z}^2}$$
(10)

In CLEO II.V and CLEO III, the resolution of the  $D^0$  decay point is of the order of  $60 \mu\text{m}$  in each coordinate. However, if only beam spot information is used at the starting point, then the  $x$  and  $z$  errors are very large,  $330 \mu\text{m}$  and  $1.7 \text{ cm}$ , respectively, and therefore contribute little to the answer. On the other hand, if tracks produced in association with the  $D^0$  are included, they can be used to determine the starting point in the other coordinates with roughly the same resolution as those of the decay point, significantly improving the lifetime error. While including these tracks we must be mindful of the previously stated caveat that tracks from anti-charm particles must not be allowed to bias the measurement.

### III Calculation of proper lifetime for the general case

Note: you may proceed directly to the next section if you are not interested in this general derivation and only want to find out how to use the **KWFIT** implementation.

Having worked out the simple case of a neutral particle with uncorrelated measurements and no momentum errors, I now turn to the general problem of a charged or neutral particle moving through a fixed magnetic field before decaying.

For simplicity, I only show here the equations for a solenoidal field, although the implementation in **KWFIT** includes the more general case of an arbitrary, but fixed, orientation. The equations relating the production and decay points in a solenoidal field are [1]

$$x_p = x_d - \frac{p_x}{a} \sin \rho s - \frac{p_y}{a} (1 - \cos \rho s)$$

$$y_p = y_d - \frac{p_y}{a} \sin \rho s + \frac{p_x}{a} (1 - \cos \rho s)$$

$$z_p = z_d - \frac{p_z}{p} s$$
(11)

where  $a = -0.299792458 Bq$ ,  $B$  is the field strength in Tesla,  $q$  is the charge,  $\rho = a/p$ ,  $(x_p, y_p, z_p)$  is the production point,  $(x_d, y_d, z_d)$  is the decay point and  $(p_x, p_y, p_z)$  is its 3-momentum there. They are functions of  $s$ , the flight distance measured from the point

$(x_p, y_p, z_p)$  to  $(x_d, y_d, z_d)$ . We eliminate  $s$  in favor of the proper lifetime  $c\tau$  using  $s = \beta ct = \gamma\beta c\tau = (p/m)c\tau$ , yielding the new equations

$$\begin{aligned}x_p &= x_d - \frac{p_x}{a} \sin(ac\tau/m) - \frac{p_y}{a} (1 - \cos(ac\tau/m)) \\y_p &= y_d - \frac{p_y}{a} \sin(ac\tau/m) + \frac{p_x}{a} (1 - \cos(ac\tau/m)) \\z_p &= z_d - \frac{p_z}{m} c\tau\end{aligned}\tag{12}$$

The lifetime  $c\tau$  is determined by recognizing that eqns. (12) represent constraint conditions. Since there are three equations and one unknown ( $c\tau$ ), there are a total of two degrees of freedom (astute readers will recognize that these are the same equations one starts with when deriving the vertex constraint, except that  $c\tau$  is usually eliminated to yield two equations).

We can apply the machinery in reference [1] to solve for  $c\tau$  and its errors, while at the same time improving the track parameters and the starting point. First we separate the variables which are known within errors ( $\alpha$ ) from those which are not known ( $c\tau$ ). The vector  $\alpha$  contains 10 variables, the 7 track parameters at the decay point and the production point  $(x_p, y_p, z_p)$ .

$$\alpha = \begin{pmatrix} p_x \\ p_y \\ p_z \\ E \\ x_d \\ y_d \\ z_d \\ x_p \\ y_p \\ z_p \end{pmatrix}\tag{13}$$

The 3 equations describing the constraints can be written generally as  $\mathbf{H}(\alpha, c\tau) = \mathbf{0}$ , where

$$\mathbf{H}(\alpha, c\tau) = \begin{pmatrix} x_p - x_d + \frac{p_x}{a} \sin(ac\tau/m) + \frac{p_y}{a} [1 - \cos(ac\tau/m)] \\ y_p - y_d + \frac{p_y}{a} \sin(ac\tau/m) - \frac{p_x}{a} [1 - \cos(ac\tau/m)] \\ z_p - z_d + \frac{p_z}{m} c\tau \end{pmatrix}\tag{14}$$

Expanding around a convenient point  $(\alpha_A, c\tau_A)$  yields the linearized constraint equations

$$\mathbf{0} = \mathbf{D}\delta\boldsymbol{\alpha} + \mathbf{E}\delta c\tau + \mathbf{d}, \quad (15)$$

where  $\mathbf{D}$  and  $\mathbf{E}$  are the derivatives of  $\mathbf{H}$  with respect to  $\boldsymbol{\alpha}$  and  $c\tau$ , respectively,  $\delta\boldsymbol{\alpha} = \boldsymbol{\alpha} - \boldsymbol{\alpha}_A$  and  $\delta c\tau = c\tau - c\tau_A$ . The constraints are incorporated using the method of Lagrange multipliers in which the  $\chi^2$  is written as a sum of two terms, e.g.

$$\chi^2 = (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0)^T \mathbf{V}_{\alpha 0}^{-1} (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0) + 2\boldsymbol{\lambda}^T (\mathbf{D}\delta\boldsymbol{\alpha} + \mathbf{E}\delta c\tau + \mathbf{d}), \quad (16)$$

where  $\boldsymbol{\lambda}$  is a vector of 3 unknowns, the Lagrange multipliers, and there is one unknown,  $c\tau$ . Note that  $\delta\boldsymbol{\alpha} = \boldsymbol{\alpha} - \boldsymbol{\alpha}_A$  and  $\delta c\tau = c\tau - c\tau_A$  are the departures of the variables from their respective expansion points. Minimizing the  $\chi^2$  with respect to  $\boldsymbol{\alpha}$ ,  $c\tau$  and  $\boldsymbol{\lambda}$  yields three vector equations which can be solved for  $\boldsymbol{\alpha}$ ,  $c\tau$  and their covariance matrices:

$$\begin{aligned} \mathbf{V}_{\alpha 0}^{-1} (\delta\boldsymbol{\alpha} - \delta\boldsymbol{\alpha}_0) + \mathbf{D}^T \boldsymbol{\lambda} &= \mathbf{0} \\ \mathbf{E}^T \boldsymbol{\lambda} &= \mathbf{0} \\ \mathbf{D}\delta\boldsymbol{\alpha} + \mathbf{E}\delta c\tau + \mathbf{d} &= \mathbf{0} \end{aligned} \quad (17)$$

The last equation demonstrates clearly that the solution satisfies the constraints. The solution for  $c\tau$  and its covariance matrix is straightforward [1], though the matrix algebra is a bit messy:

$$\begin{aligned} \delta c\tau &= -\mathbf{V}_{c\tau} \mathbf{E}^T \boldsymbol{\lambda}_0 \\ \mathbf{V}_{c\tau} &= (\mathbf{E}^T \mathbf{V}_D \mathbf{E})^{-1} \end{aligned} \quad (18)$$

The auxiliary quantities  $\boldsymbol{\lambda}_0$  and  $\mathbf{V}_D$  are defined as

$$\begin{aligned} \boldsymbol{\lambda}_0 &= \mathbf{V}_D (\mathbf{D}\delta\boldsymbol{\alpha}_0 + \mathbf{d}) \\ \mathbf{V}_D &= (\mathbf{D}\mathbf{V}_{\alpha 0}\mathbf{D}^T)^{-1} \end{aligned} \quad (19)$$

where  $\delta\boldsymbol{\alpha}_0 = \boldsymbol{\alpha}_0 - \boldsymbol{\alpha}_A$ . The same machinery also yields the updated  $\boldsymbol{\alpha}$  measurements

$$\begin{aligned} \boldsymbol{\alpha} &= \boldsymbol{\alpha}_0 - \mathbf{V}_{\alpha 0} \mathbf{D}^T \boldsymbol{\lambda} \\ \boldsymbol{\lambda} &= \boldsymbol{\lambda}_0 + \mathbf{V}_D \mathbf{E} \delta c\tau \end{aligned} \quad (20)$$

as well as their covariance matrix and correlation with  $c\tau$ :

$$\begin{aligned} \mathbf{V}_{\alpha} &= \mathbf{V}_{\alpha 0} - \mathbf{V}_{\alpha 0} \mathbf{D}^T \mathbf{V}_D \mathbf{D} \mathbf{V}_{\alpha 0} + \mathbf{V}_{\alpha 0} \mathbf{D}^T \mathbf{V}_D \mathbf{E} \mathbf{V}_{c\tau} \mathbf{E}^T \mathbf{V}_D \mathbf{D} \mathbf{V}_{\alpha 0} \\ \text{cov}(\boldsymbol{\alpha}, c\tau) &= -\mathbf{V}_{\alpha 0} \mathbf{D}^T \mathbf{V}_D \mathbf{E} \mathbf{V}_{c\tau} \end{aligned} \quad (21)$$

The  $\chi^2$  is given by

$$\chi^2 = \boldsymbol{\lambda}^T (\mathbf{D}\delta\boldsymbol{\alpha}_0 + \mathbf{d}) \quad (22)$$

Let's check the algorithm by deriving eqn. (10). We assume that the charge is zero (straight line motion) and that the momentum and mass are constant, i.e., their errors are zero. The equations relating the production and decay points are written in the standard constraint form

$$\begin{aligned}
x_p - x_d + \frac{p_x}{m} c\tau &= 0 \\
y_p - y_d + \frac{p_y}{m} c\tau &= 0 \\
z_p - z_d + \frac{p_z}{m} c\tau &= 0
\end{aligned} \tag{23}$$

First set the expansion points  $\boldsymbol{\alpha}_A^T = (p_x, p_y, p_z, E, x_d, y_d, z_d, x_p, y_p, z_p)$  and  $c\tau_A = 0$ .

Furthermore, set  $\boldsymbol{\alpha}_0 = \boldsymbol{\alpha}_A$  for simplicity. The  $\mathbf{D}$ ,  $\mathbf{d}$  and  $\mathbf{E}$  matrices are calculated by taking derivatives of eqns. (23) with respect to the track parameters and  $c\tau$ , noting that the momenta and mass can be treated as constants. This yields

$$\mathbf{D} = \begin{pmatrix} 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \end{pmatrix} \tag{24}$$

$$\mathbf{E} = \begin{pmatrix} p_x / m \\ p_y / m \\ p_z / m \end{pmatrix} \quad \mathbf{d} = \begin{pmatrix} x_p - x_d \\ y_p - y_d \\ z_p - z_d \end{pmatrix}$$

The covariance matrix, assuming diagonal errors since the errors are supposed to be uncorrelated in this simple example, is

$$\mathbf{V}_{\alpha 0} = \begin{pmatrix} 0 & & & & & & & & & \\ & 0 & & & & & & & & \\ & & 0 & & & & & & & \\ & & & 0 & & & & & & \\ & & & & \sigma_{x_d}^2 & & & & & \\ & & & & & \sigma_{y_d}^2 & & & & \\ & & & & & & \sigma_{z_d}^2 & & & \\ & & & & & & & 0 & & \\ & & & & & & & & 0 & \\ & & & & & & & & & 0 \end{pmatrix} \tag{25}$$

For simplicity I set the errors for the starting point equal to zero. These can be put back in with only a little more complication. The matrix  $\mathbf{V}_D = (\mathbf{D}\mathbf{V}_{\alpha 0}\mathbf{D}^T)^{-1}$  is easily calculated:

$$\mathbf{V}_D = \begin{pmatrix} \sigma_{x_d}^2 & & \\ & \sigma_{y_d}^2 & \\ & & \sigma_{z_d}^2 \end{pmatrix}^{-1} = \begin{pmatrix} 1/\sigma_{x_d}^2 & & \\ & 1/\sigma_{y_d}^2 & \\ & & 1/\sigma_{z_d}^2 \end{pmatrix} \quad (26)$$

Finally, we calculate the lifetime error  $\mathbf{V}_{c\tau} = \sigma_{c\tau}^2 = (\mathbf{E}^T \mathbf{V}_D \mathbf{E})^{-1}$ :

$$\begin{aligned} \mathbf{V}_{c\tau} &= \begin{pmatrix} p_x/m & p_y/m & p_z/m \end{pmatrix} \begin{pmatrix} 1/\sigma_{x_d}^2 & & \\ & 1/\sigma_{y_d}^2 & \\ & & 1/\sigma_{z_d}^2 \end{pmatrix} \begin{pmatrix} p_x/m \\ p_y/m \\ p_z/m \end{pmatrix}^{-1} \\ &= \left( \frac{p_x^2}{m^2} \frac{1}{\sigma_{x_d}^2} + \frac{p_y^2}{m^2} \frac{1}{\sigma_{y_d}^2} + \frac{p_z^2}{m^2} \frac{1}{\sigma_{z_d}^2} \right)^{-1} \end{aligned} \quad (27)$$

The solution for  $c\tau$  is found from  $c\tau = -\mathbf{V}_{c\tau} \mathbf{E}^T \mathbf{V}_D (\mathbf{D} \delta \alpha_0 + \mathbf{d})$ .

$$\begin{aligned} c\tau &= -\mathbf{V}_{c\tau} \begin{pmatrix} p_x/m & p_y/m & p_z/m \end{pmatrix} \begin{pmatrix} 1/\sigma_{x_d}^2 & & \\ & 1/\sigma_{y_d}^2 & \\ & & 1/\sigma_{z_d}^2 \end{pmatrix} \begin{pmatrix} x_p - x_d \\ y_p - y_d \\ z_p - z_d \end{pmatrix} \\ &= \frac{p_x}{m} \frac{\sigma_{c\tau}^2}{\sigma_{x_d}^2} (x_d - x_p) + \frac{p_y}{m} \frac{\sigma_{c\tau}^2}{\sigma_{y_d}^2} (y_d - y_p) + \frac{p_z}{m} \frac{\sigma_{c\tau}^2}{\sigma_{z_d}^2} (z_d - z_p) \end{aligned} \quad (28)$$

Equations (28) are identical to eqns. (10) except that we assumed the production point errors to be zero here. Putting in these errors is trivial and is left to the reader as an exercise [2].

The constraint equation approach outlined in this section has the following advantages when determining the proper lifetime. First, it automatically uses all available information, including all coordinate and momentum measurements (momentum measurements can be important for long-lived particles like the  $K_s$  and  $\Lambda$ ). The method reduces to the y-only method used by standard CLEO analyses by making the  $x$  and  $z$  errors infinite and all momentum errors zero. Second, the algorithm works with either charged or neutral particles. Finally, the proper lifetime and improved values of the track parameters and initial starting value, including their errors and correlations, are directly computed.

In the next section I show how the technique is implemented in **KWFIT**. Monte Carlo studies based on this approach are discussed later.



## IV KWFIT routines for fitting lifetime

This algorithm derived in the previous section has been encoded in several **KWFIT** routines, listed briefly here. The full calling sequences can be found in the **KWFIT** web documentation [3]. Before calling these routines, you must first compute the track parameters at the decay point. The routine **kvir\_unknown** builds a virtual particle from its daughters. Calls to one of the following routines can then be made to determine the lifetime.

- **kvtx\_beam\_lifetime** This routine is used most frequently. You supply a **KWFIT** track at the decay point, and the routine takes the beam position and size as the initial production point information to determine  $c\tau$  and its covariance matrix, which are returned. You must have previously set the beam spot and size with a call to **kset\_beam\_position**.
- **kvtx\_known\_lifetime** This routine is more general than the previous one in that the initial production point estimate and its covariance matrix are set in the argument list. Again you supply a **KWFIT** track at the decay point and  $c\tau$  and its covariance matrix are returned.
- **kvtx\_fixed\_lifetime** You supply a **KWFIT** track and a fixed production point, and  $c\tau$  and its covariance matrix are returned. This routine is normally not used with real data, since the production vertex has a finite size, but it is useful for testing Monte Carlo samples.

After the fit is performed, you may call the routine **kget\_lifetime\_covar** to return the full 4×4 covariance matrix of the production point (elements 1-3) and  $c\tau$  (element 4). This extra information can be useful in situations requiring sensitive lifetime information.

## V Monte Carlo studies of lifetime algorithm

### *Simple Monte Carlo description*

I have tested the **KWFIT** routines described in the previous section using a standalone Monte Carlo which uses the MCFAST geometry structures and input routines [4]. This geometry is general enough to accurately describe most of CLEO 2's tracking system, including energy loss and multiple scattering. However, I transferred the information to a simplified geometry in which each silicon layer is represented by a cylinder having the correct thickness and average radius and each drift chamber layer is represented by a thin measuring layer having the correct stereo angle. I used the best numbers I could find for the resolution and efficiencies of the silicon, VD and DR detectors.

The Monte Carlo traces particles through the tracking system and intersects them at each surface, making hits that follow the resolutions and efficiencies of each layer. No scattering or energy loss is done during the tracing stage, but the tracks are later Kalman fit from the outside in to include the effects of multiple scattering. The Kalman fit only determines the covariance matrix of the track at the origin. The true track parameters are smeared by the covariance matrix to give the so-called “measured track parameters”. This fitting process guarantees that the

parameters are exactly Gaussian, though retaining the exact correlations from the covariance matrix. Such exactly Gaussian-smeared tracks are useful for determining any biases that may creep in from kinematic fitting.

The Kalman fit particles obtained from this procedure reproduce reasonably well the errors seen in the CLEO data and **cleog** Monte Carlo. For example, the mass distribution from the  $D^0 \rightarrow K^- \pi^+$  decay is about 20% narrower than in **cleog**. I have not made a systematic study comparing my Monte Carlo resolutions to those of **cleog**.

### *Monte Carlo fitting procedure*

I generated approximately 10M events of the form  $D^0 \rightarrow K^- \pi^+$ , with no other particles in the event. The steps I use to determine the lifetime are outlined below:

- (1) Generate the  $D^0$  according to a Peterson function with  $\varepsilon = 0.078$ ;
- (2) Propagate the  $D^0$  by a distance generated from its proper lifetime (124.4  $\mu\text{m}$ ) and momentum;
- (3) Decay the  $D^0$  into its daughters;
- (4) Trace each daughter through the detector and fit it using the procedure described previously;
- (5) Build the measured  $D^0$  using the **KWFIT** routine **kvir\_vertex\_unknown**;
- (6) Fit for the lifetime of the  $D^0$  using **kvtx\_beam\_lifetime**, where the starting point is the center of beam and the uncertainty is specified by the beam width;
- (7) Fit for the lifetime of the  $D^0$  using **kvtx\_fixed\_lifetime**, where the starting point is the true  $D^0$  production point.

To assess the quality of the fit, I first plotted the probability of the fit for both the fixed point and beam point method. These plots are shown below:

plotted the quantities  $\Delta = c\tau_{\text{meas}} - c\tau_{\text{true}}$ ,  $\Delta/\sigma = (c\tau_{\text{meas}} - c\tau_{\text{true}})/\sigma_{c\tau}$  and  $\sigma_{c\tau}$  as function of several variables.

## References

1. Paul Avery, “Applied Fitting Theory VI: Formulas for Kinematic Fitting”, CBX 98–37, <http://www.phys.ufl.edu/~avery/fitting.html>.
2. Admit it, you’re not going to do this, are you?
3. Paul Avery, “Data Analysis and Kinematic Fitting With the **KWFIT** Library”, CBX 98–35, <http://www.phys.ufl.edu/~avery/kwfit/>.
4. MCFast web site