Paul Avery
CBX 91 − 72
October 18, 1991

Applied Fitting Theory I
General Least Squares Theory

1. General Statements

This paper is the first of four documents I am writing on general fitting theory, particularly as applied to high energy physics experiments. I have several reasons for doing this. First, I want to provide a general introduction to the subject using matrix notation for people who have not done much fitting, or who are not familiar with the very elegant matrix formulation of least squares fitting theory. Second, I want to demonstrate how constraints, even those that introduce new variables, can be incorporated into the fit (using Lagrange multipliers) and how they affect the error calculation. Third, I want to show some new techniques I have developed that greatly extend the usefulness of least squares theory. Finally, I wanted to provide a single reference for people who use least squares techniques frequently but who are unable to find practical algorithms such as the ones covered here in standard books and articles on probability and statistics.

This document replaces an earlier note, CBX 87-38, that I released in 1987. The older note served as an introduction to least squares theory, especially in the use of constraints, and was supposed to be the first of a three part series. Unfortunately, it didn't work out that way since the followup notes were never released (although they had been written).

Since that time I have developed some new factorization algorithms for applying constraints and inverting large matrices that I felt are important enough to warrant inclusion. These methods, and the ones discussed in the original paper, have particular relevance to kinematic fitting, a favorite topic of mine. One important consequence is that it should be possible to speed up the process of kinematic fitting so that it becomes a more practical tool for high energy physics analysis. In fact, I am writing a kinematic fitting package that uses the algorithms discussed in this article. It is discussed in a separate writeup.

My plan is to provide in in the first paper some general formulas for fitting, with and without constraints. These formulas are used extensively in fitting data to histograms, determining track parameters from hit information and kinematic fitting. In the second note, I will discuss how to exploit least squares fitting to determine systematic detector effects, which are either too small to be determined in a single event by fitting, or too intertwined with other effects to be reliably estimated. In the third paper, I will present an analysis of how track parameter errors due to multiple scattering can be treated exactly, although the best possible parameters are not obtained. The technique, which I call Non-optimal Least Squares fitting, can be applied to any problem in which correlations among large numbers of data points prevents a direct attack by traditional methods. Finally, in the fourth article I will discuss fitting techniques specific to kinematic fitting, including ways to factorize the fits and improve execution speed.

I have no intention of making this series of notes a complete reference on fitting theory. For the most part I assume Gaussian errors and do not worry about fitting low statistics histograms, which requires special techniques. Consequently, I omit discussion of how confidence intervals, upper limits and asymmetric errors are calculated.

A word on notation. Throughout this series lowercase bold letters ($\mathbf{x}$, $\mathbf{y}$) always refer to vector quantities and uppercase letters ($\mathbf{A}$, $\mathbf{B}$) represent matrix quantities. The symbol $\mathbf{V}$ always refers to a covariance matrix.

2. **Simple Least Squares Fitting**

Let's start off with an easy problem in least squares fitting that demonstrates the basic principle and which also illustrates the compactness and power of matrix notation. Suppose we have n independently measured data values y_l which we believe are functions of m unknown variables α_i, where $m \leq n$. In other words, we are hypothesizing the relation $y_l = f_l(\alpha_i)$. Note that the functions f_l depend on the data point in question, as can be seen easily by thinking of the index l as tracking the "independent variable" x at each point.

Since each y_l is a measured quantity with a standard deviation σ_l, the equation $y_l = f_l(\alpha_i)$ cannot be satisfied exactly if $m < n$. However, we can require that the equation be satisfied "as closely as possible" by defining the χ^2 statistic, where

$$\chi^2 = \sum_l \frac{(y_l - f_l(\alpha))^2}{\sigma_l^2},$$

and demand that the values of α_i be chosen so as to minimize χ^2. Note that χ^2 is merely the sum of the squares of the "misses" weighted by the uncertainty σ_l of each data item (each of which is, for now, independent of all others) so small values are better than large ones.

Let me expand a little on what is going on here. We assume that each measurement y_l has a measurment error σ_l. If the errors are Gaussian and the expected value for the measurement is $f_l(\alpha)$ then the probability density function (pdf) for the measurements can be written

$$g(y_l) = \frac{1}{\sqrt{2\pi}\sigma_l} \exp\left[-(y_l - f_l)^2/2\sigma_l^2\right],$$

where the normalization is chosen to make $\int_{-\infty}^{\infty} g_l(y_l)dy_l = 1$. The joint pdf for n measurements is then

$$\begin{aligned} g(y_1 \ldots y_n) &= \prod_l \frac{1}{\sqrt{2\pi}\sigma_l} \exp\left[-\sum_l (y_l - f_l)^2/2\sigma_l^2\right] \\ &= \prod_l \frac{1}{\sqrt{2\pi}\sigma_l} \exp\left(-\chi^2/2\right) \end{aligned}$$

It is clear that maximizing this pdf is equivalent to minimizing χ^2. The adjustment of the parameters α to achieve the maximization of the pdf is called the "maximum likelihood" method, and can be carried out even when the pdf is non-Gaussian, e.g., Poisson statistics.

The requirement that the χ^2 function be at a minimum is usually guaranteed by the m relations

$$\frac{\partial \chi^2}{\partial \alpha_i} = 0.$$

Unfortunately, no general method exists for solving these m equations since f_l and its first derivatives can in general be highly nonlinear functions of the parameters α_i. Often, however, the functions vary slowly enough so that one can expand to first order about an approximate solution, viz.

$$f_l(\alpha) = f_l(\alpha_A) + \sum_i (\alpha_i - \alpha_{iA}) \frac{\partial f_l(\alpha)}{\partial \alpha_i}\bigg|_{\alpha_A} \equiv f_{lA} + \sum_i A_{li}\eta_i.$$

Under this approximation, the χ^2 function can be written

$$\chi^2 = \sum_l \frac{(y_l - f_{lA} - \sum_i A_{li}\eta_i)^2}{\sigma_l^2} \equiv \sum_l \frac{(\Delta y_l - \sum_i A_{li}\eta_i)^2}{\sigma_l^2}.$$

Under very general conditions it can be proved that the parameters $\eta_i = \alpha_i - \alpha_{iA}$ obtained by minimizing this linearized version of the χ^2 function are *unbiased* and have *minimum* variance $\sigma^2_{\alpha_i}$, i.e., one cannot find a better set of parameters.

The minimization conditions, $\partial\chi^2/\partial\alpha_i = 0$, can be written in this linear approximation as

$$\sum_l \frac{A_{li}A_{lj}}{\sigma_l^2}\eta_j = \sum_l \frac{A_{li}}{\sigma_l^2}\Delta y_l,$$

where $1 \le i, j \le m$. Defining the symmetric matrix on the LHS as

$$V_{A\,ij} = \left(\sum_l A_{li}A_{lj}/\sigma_l^2\right)^{-1},$$

we get the solution

$$\eta_i = \sum_{j,l} V_{A\,ij}\frac{A_{lj}\Delta y_l}{\sigma_l^2}.$$

Surely, you say, there must be a better way of writing these equations without worrying about all the indices. To see how this might be done, let's go back to the linearized χ^2

equation and write it in matrix form:

$$\chi^2 = (\mathbf{\Delta y} - \mathbf{A}\eta)^t \mathbf{V}_y^{-1} (\mathbf{\Delta y} - \mathbf{A}\eta)$$

where $\mathbf{A}$ is the matrix of coefficients defined above and $\mathbf{V}_y^{-1}$, η and $\mathbf{y}$ are given by the expressions

$$\mathbf{V}_y^{-1} = \begin{pmatrix} 1/\sigma_1^2 & 0 & \cdots & 0 \\ 0 & 1/\sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1/\sigma_n^2 \end{pmatrix}, \quad \eta = \begin{pmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_m \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}.$$

$\mathbf{V}_y^{-1}$, the inverse of the measurement covariance matrix, is called the "weight matrix".

Taking the partial derivative with respect to the parameters η we obtain the m equations

$$\mathbf{A}^t \mathbf{V}_y^{-1} (\mathbf{\Delta y} - \mathbf{A}\eta) = \mathbf{0},$$

which can be written

$$\mathbf{A}^t \mathbf{V}_y^{-1} \mathbf{A}\eta = \mathbf{A}^t \mathbf{V}_y^{-1} \mathbf{\Delta y}.$$

Defining as before $\mathbf{V}_A = (\mathbf{A}^t \mathbf{V}_y^{-1} \mathbf{A})^{-1}$, we finally obtain the m equations

$$\eta = \mathbf{V}_A \mathbf{A}^t \mathbf{V}_y^{-1} \mathbf{\Delta y},$$

which is a much more understandable and meaningful expression than the previous one. The parameters α can be determined from $\alpha = \alpha_A + \eta$. Note that $\mathbf{V}_y^{-1}$ has dimensions of $n \times n$, $\mathbf{A}$ is of size $n \times m$, and $\mathbf{V}_A$ is of size $m \times m$. The advantage of the matrix notation is that we can easily put in the indices by inspection after the equation is written in compact form.

Sometimes it is useful to calculate the "residuals" $\mathbf{r} = \mathbf{\Delta y} - \mathbf{A}\eta = \mathbf{y} - \mathbf{f}_A(\alpha_A) - \mathbf{A}\eta$, which are merely the differences of the measurements and predicted values. The χ^2 can then be written $\chi^2 = \mathbf{r}^t \mathbf{V}_y^{-1} \mathbf{r} = \sum_l r_l^2/\sigma_l^2$.

3. **Calculation of Errors: The Covariance Matrix**

Obtaining a solution for a set of parameters is not enough: we also want to know the errors of the parameters and their correlations with one another. This information is contained in the so-called *covariance* matrix, which is defined as

$$\mathrm{cov}(x_i, x_j) \equiv V_{x\,ij} = \langle (x_i - \bar{x}_i)(x_j - \bar{x}_j) \rangle \equiv \langle \delta x_i \delta x_j \rangle,$$

where the symbol $\langle\rangle$ refers to a weighted average taken over all possible values of the enclosed expression. Note that the i^{th} diagonal entry in the covariance matrix is merely $\sigma^2_{x_i}$, the square of the standard deviation of x_i. The covariance matrix can be written in matrix notation as $\mathbf{V_x} = \langle \delta\mathbf{x} \delta\mathbf{x}^t \rangle$. For example, the covariance matrix for the (independent) data values y_l is

$$\mathbf{V}_y = \langle \delta\mathbf{y} \delta\mathbf{y}^t \rangle = \mathrm{diag}(\sigma_i^2).$$

This explains how I chose the notation for the weight matrix $\mathbf{V}_y^{-1}$ in the previous section. In general the covariance matrix $\mathbf{V}$ is related to another matrix called the *correlation* matrix ρ. The elements ρ_{ij} are computed from

$$\rho_{ij} = \frac{V_{ij}}{\sqrt{V_{ii} V_{jj}}} = \frac{V_{ij}}{\sigma_i \sigma_j}.$$

The element ρ_{ij} $(i \neq j)$ is called the ij correlation coefficient and satisfies the inequality $-1 \leq \rho_{ij} \leq 1$.

Finding the errors in the parameters η is now straightforward. Starting with the solution $\eta = \mathbf{V}_A \mathbf{A}^t \mathbf{V}_y^{-1} \mathbf{\Delta y}$ we obtain

$$\mathbf{V}_\alpha = \mathbf{V}_\eta = \langle \delta\eta \delta\eta^t \rangle = \mathbf{V}_A \mathbf{A}^t \mathbf{V}_y^{-1} \langle \delta\mathbf{y} \delta\mathbf{y}^t \rangle \mathbf{V}_y^{-1} \mathbf{A} \mathbf{V}_A = \mathbf{V}_A \mathbf{A}^t \mathbf{V}_y^{-1} \mathbf{V}_y \mathbf{V}_y^{-1} \mathbf{A} \mathbf{V}_A = \mathbf{V}_A,$$

and the errors are $\sigma_{\eta_i} = \sqrt{V_{\eta\,ii}} = \sqrt{V_{A\,ii}}$. I have used $\delta\mathbf{\Delta y} = \delta\mathbf{y}$ and the definition $\mathbf{V}_A = (\mathbf{A}^t \mathbf{V}_y^{-1} \mathbf{A})^{-1}$. The fact that $\mathbf{V}_\eta = \mathbf{V}_A$ might seem surprising, but it turns out that the least squares technique, with or without constraints, always yields an equation of the form $\mathbf{V}^{-1}\eta = \mathbf{C y}$ and gives $\mathbf{V}_\eta = \mathbf{V}$.

Once a parameter error is known, it is easy to figure out for Gaussian errors the probability that the parameter will fall within $\pm 1\sigma$, $\pm 2\sigma$ etc. The following table gives some typical values.

# standard deviations	Probability to fall in $\pm n\sigma$
0.5	0.3830
1.0	0.6826
1.5	0.8664
2.0	0.9554
2.5	0.9876
3.0	0.9974
3.5	0.9996

Of course, in real life measurement errors are rarely distributed like Gaussians and usually have some significant non-Gaussian tail. However, the Central Limit Theorem usually guarantees that the estimated parameters will be more Gaussian than the estimated measurements, provided that there are a sufficient number of measurements,

Given a set of parameters and their estimated errors, what assurance do we have that the fit is any good? To answer this question, we note that the entire least squares procedure is concerned with minimizing the χ^2, so the final value of χ^2 provides a measure of how well the fit worked. Traditionally, one defines a "confidence level" which is the probability that a fit will have a χ^2 greater than or equal to the value we found from the fit. Mathematically, we write

$$\text{C.L.} = \int_{\chi^2}^{\infty} f(x)dx,$$

where $f(\chi^2)$ is the χ^2 probability distribution function.

For Gaussian errors we can calculate $f(\chi^2)$ straightforwardly. The χ^2 can be written in the compact form

$$\chi^2 = \mathbf{\Delta y}^t \mathbf{V}_y^{-1} \mathbf{\Delta y} - \eta^t \mathbf{V}_\eta^{-1} \eta,$$

which is the difference of two positive definite forms having n and m terms, respectively. It

is straightforward to show that if the $\mathbf{y}$ are distributed as Gaussians (correlated), they can be rotated and rescaled in such a way that the χ^2 can be written in the manifestly positive form $\chi^2 = w_1^2 + w_2^2 + \cdots + w_\nu^2$, where the w_i are independent Gaussians having unit error and zero mean. $\nu = n - m$ is called the number of "degrees of freedom" of the fit. In that case, applying the Jacobian transformation to the joint pdf with $\sqrt{\chi^2}$ as the "radius" and $\nu - 1$ angular variables over which we integrate, we find the χ^2 probability distribution to be

$$f(\chi^2) = \frac{(\chi^2)^{\frac{\nu-2}{2}} e^{-\frac{1}{2}\chi^2}}{2^{\frac{\nu}{2}}\Gamma(\nu/2)}.$$

This distribution has $\overline{\chi^2} = \nu$ and $\sigma_{\chi^2} = \sqrt{2\nu}$. For large ν, the distribution of χ^2/ν is itself approximately Gaussian with mean 1 and standard deviation $\sqrt{2/\nu}$, so that one can estimate the confidence level from the number of standard deviations χ^2/ν exceeds 1.

4. **Fitting with Correlated Errors**

It frequently happens that the data measurements y_l are not independent of each other, so that movement of a particular measurement will cause the simultaneous adjustment of other measurements. A particularly odious example of this is multiple scattering, in which a single scattering event at a particular point affects the drift distance measurements at all subsequent points. In this case the deviation of the measurements from the original track will have an independent component due to random measurement errors and a correlated component due to the presence of the multiple scattering event.

Correlated data measurements can be accounted for in a simple way in the least squares algorithm. The χ^2 function, obtained from section 2, can be written as

$$\chi^2 = (\mathbf{\Delta y} - \mathbf{A}\eta)^t \mathbf{V}_y^{-1} (\mathbf{\Delta y} - \mathbf{A}\eta),$$

where the "weight" matrix $\mathbf{V}_y^{-1}$ should now be considered to be the inverse of the covariance matrix $\mathbf{V}_y = \langle \delta\mathbf{y}\delta\mathbf{y}^t \rangle$, and reduces to the familiar diagonal form for independent measurements. The Gauss-Markov theorem guarantees that the parameters η obtained by minimizing this χ^2 function are both unbiased and have minimum variance, just as in the

case of independent measurements. The solution for the parameters $\eta = \alpha - \alpha_A$ and their covariance matrix is then exactly the same as before, $\eta = \mathbf{V}_A \mathbf{A}^t \mathbf{V}_y^{-1} \mathbf{\Delta y}$, $\mathbf{V}_\eta = \mathbf{V}_A$. The only difficulty is practicality: it is frequently too time consuming to invert $\mathbf{V}_y$ to get the matrix $\mathbf{V}_y^{-1}$. In the third paper in this series I will show a way around this difficulty.

5. **Fitting with Constraints**

Many times it is useful to fit a set of data points to parameters whose values must satisfy constraint equations. In general, one uses constraints to impose "external knowledge"of a physical process to govern the behavior of the fit, thereby forcing the fit to conform to physical principles which are not known by the internal variables of the fit.

For example, forcing 2 tracks to come from a common vertex or to be back- to-back are common uses of constraints. In the first case, the external knowledge is the fact that the tracks had to emerge from a single space-time point (disregarding secondary vertices). In the second, knowledge of the kinematics of two body decays constrains the behavior of the fit.

When constraints are applied, the effective number of unknowns in the fit is reduced by the number of constraints. This can be seen in a trivial way by writing the chisquare equation (to be minimized) with m parameters together with the r constraint equations. By substituting the r constraints in the χ^2 equation one is left with an expression having $m - r$ unknowns. In the second example above, for instance, the effective number of parameters is reduced from 10 to $10 - 5 = 5$ because of the 5 back-to-back constraints.

Although the elimination of unknowns by direct substitution is popular and fast, there are certain tradeoffs. First of all, the substitution can only be performed when one has relatively simple constraints. In many cases the substitution is impossible because the equations cannot be solved analytically for the variables to be substituted. Secondly, the resulting covariance matrix is calculated only for the reduced set of variables and it is sometimes difficult to extend it to the more meaningful set of original parameters by propagation of errors. In short, the somewhat reduced execution times of substitution algorithms is offset by the special computer code that must be written for each case. Moreover, the addition of new constraints is accompanied by even more program rewrites, which inevitably result in lost time and increased errors.

The method of Lagrange multipliers solves these problems by the rather counterintuitive notion of introducing more variables into the fit. Let the r constraint equations be written $\mathbf{H}(\alpha) = \mathbf{0}$. Expanding about α_A as before, we obtain the linearized equations $\mathbf{H}(\alpha_A) + (\alpha - \alpha_A)\partial\mathbf{H}(\alpha_A)/\partial\alpha \equiv \mathbf{D}\eta + \mathbf{d} = \mathbf{0}$, using obvious notation, where $\mathbf{D}$ is a $r \times m$ matrix and $\mathbf{d}$ is a vector of length r. The constraints are imposed by adding a new term to the χ^2 equation,

$$\chi^2 = (\mathbf{\Delta y} - \mathbf{A}\eta)^t \mathbf{V}_y^{-1} (\mathbf{\Delta y} - \mathbf{A}\eta) + 2\lambda^t (\mathbf{D}\eta + \mathbf{d}),$$

where λ, a vector of length r, is the set of Lagrange multipliers. The solution for η is obtained by setting the partial derivatives with respect to η and λ to 0. The $\partial/\partial\lambda = \mathbf{0}$ equation generates the constraint conditions. The solution to the equations shows that η is equal to η_0 plus a term proportional to λ, i.e. the constraints "pull" the parameters η away from their unconstrained values η_0. This result suggests that the solution can be "factored" into two pieces: (1) solving the unconstrained equations for η_0 and (2) applying the constraints to solve for η in terms of η_0. Since all the uncertainties in the data measurements have been absorbed into $\mathbf{V}_{\eta_0}$, we don't expect to see any explicit reference to $\mathbf{A}$ or $\mathbf{V}_A$ when constraints are applied. Our intuition is correct as can be seen from the fact that the full χ^2 can be written (after a little manipulation)

$$\chi^2 = (\mathbf{\Delta y} - \mathbf{A}\eta_0)^t \mathbf{V}_y^{-1} (\mathbf{\Delta y} - \mathbf{A}\eta_0) + (\eta - \eta_0)^t \mathbf{V}_{\eta_0}^{-1} (\eta - \eta_0) + \lambda^t (\mathbf{D}\eta + \mathbf{d}).$$

The first term is constant and is just the χ^2 for the initial unconstrained fit. Our preferred method then is to perform the fit initially *without* constraints and then add the constraints after the fact using the new χ^2

$$\chi^2 = (\eta - \eta_0)^t \mathbf{V}_{\eta_0}^{-1} (\eta - \eta_0) + 2\lambda^t (\mathbf{D}\eta + \mathbf{d}).$$

The minimization conditions $\partial\chi^2/\partial\eta = \mathbf{0}$ and $\partial\chi^2/\partial\lambda = \mathbf{0}$ yield the equations

$$\begin{aligned} \mathbf{V}_{\eta_0}^{-1}(\eta - \eta_0) + \mathbf{D}^t\lambda &= \mathbf{0} \\ \mathbf{D}\eta + \mathbf{d} &= \mathbf{0}, \end{aligned}$$

which, when solved, yield

$$\begin{aligned}
\lambda &= \mathbf{V}_D(\mathbf{D}\eta_0 + \mathbf{d}) \equiv \mathbf{V}_D\alpha \qquad (\alpha = \mathbf{D}\eta_0 + \mathbf{d}) \\
\eta &= \eta_0 - \mathbf{V}_{\eta_0}\mathbf{D}^t\lambda \\
\mathbf{V}_D &= (\mathbf{D}\mathbf{V}_{\eta_0}\mathbf{D}^t)^{-1} \\
\mathbf{V}_\lambda &= \mathbf{V}_D \\
\mathbf{V}_\eta &= \mathbf{V}_{\eta_0} - \mathbf{V}_{\eta_0}\mathbf{D}^t\mathbf{V}_\lambda\mathbf{D}\mathbf{V}_{\eta_0}. \\
\chi^2 &= \lambda^t\mathbf{V}_D^{-1}\lambda = \alpha^t\mathbf{V}_D\alpha = \lambda^t\alpha = \alpha^t\lambda
\end{aligned}$$

The expression for the χ^2 shows quite clearly that there are r degrees of freedom, one per constraint.

This factorization process of applying constraints to previously fitted parameters (after all the uncertainities have been absorbed in the parameter covariance matrix) will be used frequently in this series of papers. I will show later in this document that one can even factorize the constraints in the sense that the parameters fitted using one set of constraints can be used as input to a second set to obtain the same overall solution as would have been obtained from using the constraints all at once. This has application to kinematic fitting.

6. **Solving for Unknown Parameters in the Constraint Equations**

1. Direct solution

The constrained fitting technique derived in the previous section can be generalized to the case in which the constraint equations involve new unknown parameters which must be calculated in addition to those that appear directly in the χ^2 function. A common example is the fitting of several tracks under the constraint that they intersect at an unknown space point.

Instead of applying the Lagrange multiplier technique directly on the original χ^2 function, we shall use the two step procedure developed in the last section. We first perform the fit without constraints and get the solution $\eta_0 = \mathbf{V}_A\mathbf{A}^t\mathbf{V}_y^{-1}\mathbf{\Delta y}$ with the covariance matrix

$\mathbf{V}_{\eta_0} = \mathbf{V}_A$. The new χ^2 equation written in terms of η and including the constraints is then

$$\chi^2 = (\eta - \eta_0)^t \mathbf{V}_{\eta_0}^{-1} (\eta - \eta_0) + 2\lambda^t (\mathbf{D}\eta + \mathbf{E}\mathbf{z} + \mathbf{d}),$$

where $\mathbf{z}$ contains the undetermined parameters (q of them) and $\mathbf{E}$ is a $r \times q$ matrix. Minimizing this function with respect to η, λ and $\mathbf{z}$ yields the equations

$$\begin{aligned} \mathbf{V}_{\eta_0}^{-1}(\eta - \eta_0) + \mathbf{D}^t\lambda &= \mathbf{0} \\ \mathbf{D}\eta + \mathbf{E}\mathbf{z} + \mathbf{d} &= \mathbf{0} \\ \mathbf{E}^t\lambda &= \mathbf{0}. \end{aligned}$$

Solving these equations is somewhat more tedious than in the previous section since more matrix manipulation is involved. It can be written

$$\begin{aligned} \lambda_0 &= \mathbf{V}_D(\mathbf{D}\eta_0 + \mathbf{d}) \equiv \mathbf{V}_D\alpha_0 \qquad (\alpha_0 = \mathbf{D}\eta_0 + d) \\ \mathbf{z} &= -\mathbf{V}_E\mathbf{E}^t\lambda_0 \\ \lambda &= \lambda_0 + \mathbf{V}_D\mathbf{E}\mathbf{z} = \mathbf{V}_D(\mathbf{D}\eta_0 + \mathbf{E}\mathbf{z} + \mathbf{d}) \equiv \mathbf{V}_D\alpha \\ \eta &= \eta_0 - \mathbf{V}_{\eta_0}\mathbf{D}^t\lambda \\ \chi^2 &= \lambda^t\mathbf{V}_D^{-1}\lambda = \lambda_0^t\mathbf{V}_D^{-1}\lambda_0 - \mathbf{z}^t\mathbf{V}_{\mathbf{z}}^{-1}\mathbf{z} \\ &= \alpha^t\mathbf{V}_\lambda\alpha = \lambda^t\alpha = \alpha^t\lambda = \lambda_0^t\alpha = \lambda^t\alpha_0 \end{aligned}$$

where the auxiliary matrices are defined as follows

$$\begin{aligned} \mathbf{V}_D &= (\mathbf{D}\mathbf{V}_{\eta_0}\mathbf{D}^t)^{-1} \\ \mathbf{V}_{\mathbf{z}} &= \mathbf{V}_E = (\mathbf{E}^t\mathbf{V}_D\mathbf{E})^{-1} \\ \mathbf{V}_\lambda &= \mathbf{V}_D - \mathbf{V}_D\mathbf{E}\mathbf{V}_E\mathbf{E}^t\mathbf{V}_D \\ \mathbf{V}_\eta &= \mathbf{V}_{\eta_0} - \mathbf{V}_{\eta_0}\mathbf{D}^t\mathbf{V}_\lambda\mathbf{D}\mathbf{V}_{\eta_0} \\ \mathrm{cov}(\mathbf{z}, \eta) &= -\mathbf{V}_E\mathbf{E}^t\mathbf{V}_D\mathbf{D}\mathbf{V}_{\eta_0} \end{aligned}$$

The operations should be performed in the following order (assuming that η_0 and $\mathbf{V}_{\eta_0}$ are available from a previous unconstrained fit):

1. Calculate $\mathbf{V}_D$ and $\mathbf{V}_E$

2. Calculate λ_0

3. Calculate $\mathbf{z}$

4. Calculate λ

5. Calculate η

The number of degrees of freedom is clearly $r - q$, the number of constraints minus the number of unknown parameters which have to be determined.

2. Huge error method

There is a clear parallel between the above equations and those derived in the previous section without the new variables $\mathbf{z}$. The only difference is that $\mathbf{z}$ have no prior covariance matrix and so must be determined by a special procedure. However, it is clear that we do not change the problem by adding to the χ^2 the term $\mathbf{z}^t L^{-1}\mathbf{z}$, where L corresponds to huge errors in $\mathbf{z}$. Since both η and $\mathbf{z}$ have a prior covariance matrix, the simple algorithm derived in the last section can be used to solve for η and $\mathbf{z}$. This must give the same answer as the method used in this section in the limit $L \to \infty$.

We introduce the "huge error" algorithm by adding a $\mathbf{z}$ covariance term (corresponding to huge errors) to the χ^2 and then rewrite the χ^2 in terms of new variables $\mathbf{w}$:

$$\begin{aligned}\chi^2 &= (\eta - \eta_0)^t \mathbf{V}_{\eta_0}^{-1}(\eta - \eta_0) + \mathbf{z}^t L^{-1}\mathbf{z} + 2\lambda^t(\mathbf{D}\eta + \mathbf{E}\mathbf{z} + \mathbf{d}) \\ &\equiv (\mathbf{w} - \mathbf{w}_0)^t \mathbf{V}_{\mathbf{w}_0}^{-1}(\mathbf{w} - \mathbf{w}_0) + 2\lambda^t(\mathbf{D}'\mathbf{w} + \mathbf{d}),\end{aligned}$$

where

$$\mathbf{w} = \begin{pmatrix}\eta \\ \mathbf{z}\end{pmatrix}, \quad \mathbf{D}' = (\,\mathbf{D} \quad \mathbf{E}\,) \quad \text{and} \quad \mathbf{V}_{\mathbf{w}_0} = \begin{pmatrix}\mathbf{V}_{\eta_0} & 0 \\ 0 & L\end{pmatrix},$$

and $\mathbf{w}_0 = \begin{pmatrix}\eta_0 \\ \mathbf{0}\end{pmatrix}$. Since the expanded variables $\mathbf{w}$ have an initial covariance matrix the solution for $\mathbf{w}$ and $\mathbf{V}_{\mathbf{w}}$ can be taken straight from Section 5:

$$\begin{aligned}\lambda' &= \mathbf{V}_{D'}(\mathbf{D}'\eta_0 + \mathbf{d}) \equiv \mathbf{V}_{D'}\alpha_0 \qquad (\alpha_0 = \mathbf{D}\eta_0 + \mathbf{d}) \\ \mathbf{w} &= \mathbf{w}_0 - \mathbf{V}_{\mathbf{w}_0}\mathbf{D}'^t\lambda' \\ \mathbf{V}_{D'} &= (\mathbf{D}'\mathbf{V}_{\mathbf{w}_0}\mathbf{D}'^t)^{-1} = (\mathbf{D}\mathbf{V}_D\mathbf{D}^t + \mathbf{E}L\mathbf{E}^t)^{-1} \\ \mathbf{V}_{\mathbf{w}} &= \mathbf{V}_{\mathbf{w}_0} - \mathbf{V}_{\mathbf{w}_0}\mathbf{D}'^t\mathbf{V}_{D'}\mathbf{D}'\mathbf{V}_{\mathbf{w}_0} \\ \chi^2 &= \lambda'^t\mathbf{V}_{D'}^{-1}\lambda' = \alpha_0^t\mathbf{V}_{D'}\alpha_0 = \lambda'^t\alpha_0 = \alpha_0^t\lambda'.\end{aligned}$$

The solutions for η, $\mathbf{z}$, $\mathbf{V}_\eta$, $\mathrm{cov}(\mathbf{z},\eta)$ and $\mathbf{V}_\mathbf{z}$ can either be extracted from the full solution above using

$$\mathbf{w} = \begin{pmatrix} \eta \\ \mathbf{z} \end{pmatrix}, \quad \mathbf{V}_\mathbf{w} = \begin{pmatrix} \mathbf{V}_\eta & \mathrm{cov}(\mathbf{z},\eta) \\ \mathrm{cov}(\eta,\mathbf{z}) & \mathbf{V}_\mathbf{z} \end{pmatrix},$$

or by direct calculation:

$$\begin{aligned} \eta &= \eta_0 - \mathbf{V}_{\eta_0}\mathbf{D}^t\lambda' \\ \mathbf{z} &= -L\mathbf{E}^t\lambda' \\ \mathbf{V}_\eta &= \mathbf{V}_{\eta_0} - \mathbf{V}_{\eta_0}D^t\mathbf{V}_{D'}D\mathbf{V}_{\eta_0} \\ V_\mathbf{z} &= L - L\mathbf{E}^t\mathbf{V}_{D'}\mathbf{E}L \\ \mathrm{cov}(\mathbf{z},\eta) &= -L\mathbf{E}^t\mathbf{V}_{D'}\mathbf{D}\mathbf{V}_{\eta_0}. \end{aligned}$$

It can be shown by expanding $\mathbf{V}_{D'}$ using the Woodbury inversion algorithm (see Appendix) that the huge error method in the limit $L \to \infty$ gives exactly the same solution as the direct technique. Moreover, it is simpler to implement on a computer and it reuses the same algorithm derived for the case where no new variables were introduced in the constraints. One must be careful not to make the errors L too large, because the parameters and their covariance matrix are obtained by subtracting very large numbers, causing a significant loss of precision. One should almost certainly use double precision when applyng this method.

7. **Factorization of Constraints**

Earlier in this paper I showed that when constraints are applied to a fit, it is possible to factor the problem into two parts: (1) obtain the unconstrained parameters by solving the fit without constraints and (2) apply the constraints to these unconstrained parameters to get the final parameters. In this section I will show that the idea of factorization can be generalized so that constraints themselves can be applied sequentially.

The theorem to be proved can be stated as follows. Suppose we have a set of m parameters η_0 with covariance matrix $\mathbf{V}_{\eta_0}$ and a set of r constraints that we want to apply (see Section 5). We divide the r constraints into disjoint subsets containing r_1, r_2,... r_n elements. If each of these constraint subsets are applied sequentially, using the parameters and covariance matrix of the previous step and generating a new set of parameters and covariance

matrix for the next step, then the final set of parameters η and its covariance matrix $\mathbf{V}_\eta$ is the same as would have been obtained by applying simultaneously all $r = r_1 + r_2 + \cdots + r_n$ constraints to the initial parameters.

In a sense, this theorem *must* be true because the final answer shouldn't depend on the fact that we applied the constraints sequentially rather than simultaneously. The proof is somewhat interesting, however, so I'll go through it.

First, I'll divide the constraints into two pieces and apply them sequentially. The proof for two pieces is sufficient since each piece can be subdivided as needed. Write the χ^2 as (see Section 5)

$$\chi^2 = (\eta - \eta_0)^t \mathbf{V}_{\eta_0}^{-1} (\eta - \eta_0) + 2\lambda^t (\mathbf{D}\eta + \mathbf{d}).$$

where

$$\mathbf{D} = \begin{pmatrix} \mathbf{D}_1 \\ \mathbf{D}_2 \end{pmatrix}, \quad \mathbf{d} = \begin{pmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \end{pmatrix} \quad \text{and} \quad \lambda = \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix}.$$

Let's first find the intermediate parameters η_1 by applying the constraints $\mathbf{D}_1\eta_1 + \mathbf{d}_1 = \mathbf{0}$ to

$$\chi_1^2 = (\eta - \eta_0)^t \mathbf{V}_{\eta_0}^{-1} (\eta - \eta_0) + 2\lambda'^t (\mathbf{D}_1\eta + \mathbf{d}_1).$$

The solution from Section 5 is

$$\begin{aligned}
\lambda_1' &= \mathbf{V}_{D_1}(\mathbf{D}_1\eta_0 + \mathbf{d}_1) \equiv \mathbf{V}_{D_1}\alpha_1 \\
\eta_1 &= \eta_0 - \mathbf{V}_{\eta_0}\mathbf{D}_1^t\lambda_1' \\
\mathbf{V}_{D_1} &= (\mathbf{D}_1\mathbf{V}_{\eta_0}\mathbf{D}_1^t)^{-1} \\
\mathbf{V}_{\eta_1} &= \mathbf{V}_{\eta_0} - \mathbf{V}_{\eta_0}\mathbf{D}_1^t\mathbf{V}_{D_1}\mathbf{D}_1\mathbf{V}_{\eta_0} \\
\chi_1^2 &= \lambda_1'^t\mathbf{V}_{D_1}^{-1}\lambda_1' = \lambda_1'^t\alpha_1.
\end{aligned}$$

The parameters η_1 and covariance matrix $\mathbf{V}_{\eta_1}$ are then used as input for the constraints

$\mathbf{D}_2\eta_2 + \mathbf{d}_2 = \mathbf{0}$. The solution is similar to the one above, i.e.,

$$\begin{aligned}
\lambda_2' &= \mathbf{V}_{D_2}(\mathbf{D}_2\eta_1 + \mathbf{d}_2) \equiv \mathbf{V}_{D_2}\alpha_2' \\
\alpha_2' &= \alpha_2 - \mathbf{V}_{\eta_0}\mathbf{D}_1^t\mathbf{V}_{D_1}\alpha_1 \qquad (\alpha_2 = \mathbf{D}_2\eta_0 + \mathbf{d}_2) \\
\eta_2 &= \eta_1 - \mathbf{V}_{\eta_1}\mathbf{D}_2^t\lambda_2', \\
\mathbf{V}_{D_2} &= (\mathbf{D}_2\mathbf{V}_{\eta_1}\mathbf{D}_2^t)^{-1} \\
\mathbf{V}_{\eta_2} &= \mathbf{V}_{\eta_1} - \mathbf{V}_{\eta_1}\mathbf{D}_2^t\mathbf{V}_{D_2}\mathbf{D}_2\mathbf{V}_{\eta_1} \\
\chi_2^2 &= \lambda_2'^t\mathbf{V}_{D_2}^{-1}\lambda_2' = \lambda_2'^t\alpha_2'.
\end{aligned}$$

Using the above two sets of equations we can express the solution for η_2 directly in terms of η_0 and $\mathbf{V}_{\eta_0}$:

$$\begin{aligned}
&\eta_2 = \eta_0 - \mathbf{V}_{\eta_0}\mathbf{D}_1^t\mathbf{V}_{D_1}\alpha_1 - (\mathbf{V}_{\eta_0} - \mathbf{V}_{\eta_0}\mathbf{D}_1^t\mathbf{V}_{D_1}\mathbf{D}_1\mathbf{V}_{\eta_0})\mathbf{D}_2^t\mathbf{V}_{D_2}(\alpha_2 - \mathbf{V}_{\eta_0}\mathbf{D}_1^t\mathbf{V}_{D_1}\alpha_1) \\
&\chi^2 = \chi_1^2 + \chi_2^2,
\end{aligned}$$

or, using the vector $\alpha = \begin{pmatrix}\alpha_1\\ \alpha_2\end{pmatrix}$

$$\eta_2 = \eta_0 - \mathbf{V}_{\eta_0}\begin{pmatrix}\mathbf{D}_1^t & \mathbf{D}_2^t\end{pmatrix}\begin{pmatrix}\mathbf{V}_{D_1} + \mathbf{V}_{D_1}\mathbf{D}_1\mathbf{V}_{\eta_0}\mathbf{D}_2^t\mathbf{V}_{D_2}\mathbf{V}_{\eta_0}\mathbf{D}_1^t\mathbf{V}_{D_1} & -\mathbf{V}_{D_1}\mathbf{D}_1\mathbf{V}_{\eta_0}\mathbf{D}_2^t\mathbf{V}_{D_2} \\ -\mathbf{V}_{D_2}\mathbf{D}_2\mathbf{V}_{\eta_0}\mathbf{D}_1^t\mathbf{V}_{D_1} & \mathbf{V}_{D_2}\end{pmatrix}\begin{pmatrix}\alpha_1\\ \alpha_2\end{pmatrix}$$

Compare this equation to the one derived in Section 5 for the constraints applied simultaneously. The solution can be written as

$$\begin{aligned}
\eta &= \eta_0 - \mathbf{V}_{\eta_0}\mathbf{D}^t\mathbf{V}_D\alpha \\
&= \eta_0 - \mathbf{V}_{\eta_0}\begin{pmatrix}\mathbf{D}_1^t & \mathbf{D}_2^t\end{pmatrix}\begin{pmatrix}\mathbf{D}_1\mathbf{V}_{\eta_0}\mathbf{D}_1^t & \mathbf{D}_1\mathbf{V}_{\eta_0}\mathbf{D}_2^t \\ \mathbf{D}_2\mathbf{V}_{\eta_0}\mathbf{D}_1^t & \mathbf{D}_2\mathbf{V}_{\eta_0}\mathbf{D}_2^t\end{pmatrix}^{-1}\begin{pmatrix}\alpha_1\\ \alpha_2\end{pmatrix}.
\end{aligned}$$

However, the inverse of the matrix in parentheses is just the large matrix in the sequential solution, as can be verified by the factorization matrix inversion algorithm discussed in the Appendix. This proof even works when there are unknown parameters $\mathbf{z}$ in the constraint equations, since, as we showed in Section 6, the presence of new parameters can be cast into a form without the variables using an appropriate limit process.

The fact that constraints can be applied one after the other is very useful for kinematic fitting. Consider the decay sequence

$$\begin{aligned} B^- &\to D^{*0}\pi^-, \\ D^{*0} &\to D^0\pi^0, \\ D^0 &\to K^-\pi^+, \\ \pi^0 &\to \gamma\gamma, \end{aligned}$$

and assume that we want to apply mass constraints to the B^-, D^{*0}, D^0 and π^0 and the beam energy constraint to the B^-. The four decay processes can be fit simultaneously to give an overall χ^2. A more sensible approach, however, would be to fit, in order, the π^0, D^0 and D^{*0} and then use these particles in the fit for the B^- with the remaining mass and beam energy constraint. The final χ^2 is just the sum of the χ^2s for the individual fits. This allows us to reuse the particles in other fits and to develop standard procedures for fitting lower level particles.

8. **Simple Application of Unconstrained Track Fitting**

In this section I will apply some of the ideas discussed so far to a "toy" track fitting problem. Consider a set of n planar drift chambers arrayed along the x axis in a magnetic field. If the momentum is not too small, the equation of motion is $y = a + bx + cx^2$, where a, b and c are, respectively, the offset, slope and half the curvature of the track. The equation of motion is linear in these 3 parameters.

Since the problem is linear, we can set the parameters to 0 initially ($\alpha_A = \mathbf{0}$, $\mathbf{f}_A = \mathbf{0}$) because convergence is guaranteed in a single iteration. The solution is given by $\eta = \mathbf{V}_A\mathbf{A}^t\mathbf{V}_y^{-1}\mathbf{y}$. The matrices are calculated as follows.

$$\mathbf{A} = \begin{pmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 \end{pmatrix}$$

$$\eta = \begin{pmatrix} a \\ b \\ c \end{pmatrix}, \qquad \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

$$\mathbf{V}_y^{-1} = \begin{pmatrix} 1/\sigma_1^2 & 0 & \cdots & 0 \\ 0 & 1/\sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1/\sigma_n^2 \end{pmatrix}$$

$$\mathbf{V}_A = (\mathbf{A}^t \mathbf{V}_y^{-1} \mathbf{A})^{-1} = \begin{pmatrix} \sum_l 1/\sigma_l^2 & \sum_l x_l/\sigma_l^2 & \sum_l x_l^2/\sigma_l^2 \\ \sum_l x_l/\sigma_l^2 & \sum_l x_l^2/\sigma_l^2 & \sum_l x_l^3/\sigma_l^2 \\ \sum_l x_l^2/\sigma_l^2 & \sum_l x_l^3/\sigma_l^2 & \sum_l x_l^4/\sigma_l^2 \end{pmatrix}^{-1}$$

$$\eta = \mathbf{V}_A \begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \\ x_1^2 & x_2^2 & \cdots & x_n^2 \end{pmatrix} \begin{pmatrix} y_1/\sigma_1^2 \\ y_2/\sigma_2^2 \\ \vdots \\ y_n/\sigma_n^2 \end{pmatrix} = \mathbf{V}_A \begin{pmatrix} \sum_l y_1/\sigma_l^2 \\ \sum_l x_l y_l/\sigma_l^2 \\ \sum_l x_l^2 y_l/\sigma_l^2 \end{pmatrix}$$

To get an idea what the covariance matrix looks like, let's assume further that the measurement errors are identical ($\sigma_i = \sigma$) and that the n points are evenly distributed on the interval $0 \le x \le L$. We will also assume that n is large so that the difference between n, $n+1$, $n+2$, etc. can be ignored. In the approximation of large n, the sums in $\mathbf{V}_A^{-1}$ can be approximately evaluated as integrals to yield

$$\mathbf{V}_A = \frac{\sigma^2}{n} \begin{pmatrix} 1 & \frac{1}{2}L & \frac{1}{3}L^2 \\ \frac{1}{2}L & \frac{1}{3}L^2 & \frac{1}{4}L^3 \\ \frac{1}{3}L^2 & \frac{1}{4}L^3 & \frac{1}{5}L^4 \end{pmatrix}^{-1} = \frac{\sigma^2}{n} \begin{pmatrix} 9 & -36/L & 30/L^2 \\ -36/L & 192/L^2 & -180/L^3 \\ 30/L^2 & -180/L^3 & 180/L^4 \end{pmatrix}.$$

In this case the errors can be obtained from the diagonal portion of $\mathbf{V}_A$ to yield:

$$\sigma_a = \sqrt{\frac{9}{n}}\sigma$$
$$\sigma_b = \sqrt{\frac{192}{n}}\frac{\sigma}{L}$$
$$\sigma_c = \sqrt{\frac{180}{n}}\frac{\sigma}{L^2}.$$

These results were first obtained by Gluckstern in 1964 using a somewhat different approach.

9. **Simple Application of Constrained Track Fitting**

Let's take the simple model from the last section and apply constraints to it. An interesting set of constraints is to require that two tracks emerge back to back from a common interaction point. The subscripts 1 and 2 will denote the track being considered. Then η, the vector of unknowns, is 6 dimensional and has the form

$$\eta = \begin{pmatrix} a_1 \\ b_1 \\ c_1 \\ a_2 \\ b_2 \\ c_2 \end{pmatrix}$$

where we want to enforce the constraints $a_2 = a_1$, $b_2 = b_1$ and $c_2 = c_1$. We demand that n equally spaced measurements be taken of track 1 in the interval $0 \leq x \leq L$ and the same number for track 2 be taken for $-L \leq x \leq 0$ (that's right, folks, we're faking a back to back 2 track event in a solenoidal detector).

The covariance matrix for the two tracks is initially block diagonal because the tracks were fit independently. We can write it as

$$\mathbf{V}_{\eta_0} = \frac{\sigma^2}{n} \begin{pmatrix} 9 & -36/L & 30/L^2 & 0 & 0 & 0 \\ -36/L & 192/L^2 & -180/L^3 & 0 & 0 & 0 \\ 30/L^2 & -180/L^3 & 180/L^4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 9 & 36/L & 30/L^2 \\ 0 & 0 & 0 & 36/L & 192/L^2 & 180/L^3 \\ 0 & 0 & 0 & 30/L^2 & 180/L^3 & 180/L^4 \end{pmatrix}.$$

Some of the off-diagonal terms for the second track switch sign because the matrix value is summed over odd powers of x for these values.

The constraint equations are written $\mathbf{D}\eta + \mathbf{d} = \mathbf{0}$. By inspection we see that $\mathbf{d} = \mathbf{0}$ and

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

We can compute $\mathbf{V}_D$ as

$$\begin{aligned} \mathbf{V}_\lambda = \mathbf{V}_D &= (D\mathbf{V}_{\eta_0}\mathbf{D}^t)^{-1} \\ &= \frac{n}{\sigma^2}\begin{pmatrix} 18 & 0 & 60/L^2 \\ 0 & 384/L^2 & 0 \\ 60/L^2 & 0 & 360/L^4 \end{pmatrix}^{-1} = \frac{n}{\sigma^2}\begin{pmatrix} 1/8 & 0 & -L^2/48 \\ 0 & L^2/384 & 0 \\ -L^2/48 & 0 & L^4/160 \end{pmatrix} \end{aligned}$$

The new track variables are computed, using the matrices we just determined, from

$$\lambda = \mathbf{V}_D(\mathbf{D}\eta_0 + \mathbf{d}) = \mathbf{V}_D \begin{pmatrix} a_{1_0} - a_{2_0} \\ b_{1_0} - b_{2_0} \\ c_{1_0} - c_{2_0} \end{pmatrix} = \frac{n}{\sigma^2}\begin{pmatrix} (a_{1_0} - a_{2_0})/8 - (c_{1_0} - c_{2_0})L^2/48 \\ (b_{1_0} - b_{2_0})L^2/384 \\ -(a_{1_0} - a_{2_0})L^2/48 + (c_{1_0} - c_{2_0})L^4/160 \end{pmatrix}$$

$$\eta = \eta_0 - \mathbf{V}_{\eta_0}\mathbf{D}^t\lambda.$$

Finally, we can compute the full covariance matrix of the new variables as

$$\begin{aligned} \mathbf{V}_\eta &= \mathbf{V}_{\eta_0} - \mathbf{V}_{\eta_0}\mathbf{D}^t\mathbf{V}_\lambda\mathbf{D}\mathbf{V}_{\eta_0} \\ &= \frac{\sigma^2}{n}\begin{pmatrix} 9/8 & 0 & -15/8L^2 & 9/8 & 0 & -15/8L^2 \\ 0 & 3/2L^2 & 0 & 0 & 3/2L^2 & 0 \\ -15/8L^2 & 0 & 45/8L^4 & -15/8L^2 & 0 & 45/8L^4 \\ 9/8 & 0 & -15/8L^2 & 9/8 & 0 & -15/8L^2 \\ 0 & 3/2L^2 & 0 & 0 & 3/2L^2 & 0 \\ -15/8L^2 & 0 & 45/8L^4 & -15/8L^2 & 0 & 45/8L^4 \end{pmatrix} \end{aligned}$$

This covariance matrix (which took me 2 hours to compute, so I hope someone is reading this) exhibits all the properties we would expect from a constraint in which track parameters are forced to be equal: (1) The correlation between any two parameters in the same track is the same as that between the same two parameters in the other track or between both

tracks; (2) The curvature variance (parameter 3) is 32 times smaller than it was for the unconstrained fit. This makes sense because we can think of the 2 tracks forming a single long track with twice as many measurements and twice the length of a single track. The factor of 1/32 comes from the fact that the variance for the curvature scales like $1/nL^4$. This same argument cannot be applied to the other 2 parameters because their accuracy depends on where the measurements take place, i.e., in the unconstrained case they are determined externally to the measurement region whereas when constraints are applied they are determined within the measurement region of the "single" track.

Appendix: Tricks for Inverting Large Matrices

1. Factorization method

Many square matrices are "almost" invertible in the sense that they can be written as the sum of a large invertible piece plus a piece that spoils the inversion. An example which comes up frequently is the band matrix

$$\begin{pmatrix} xxx & 0 & 0 & \cdots & yyy \\ 0 & xxx & 0 & \cdots & yyy \\ 0 & 0 & xxx & \cdots & yyy \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ yyy & yyy & yyy & \cdots & zzz \end{pmatrix}$$

where "xxx", "yyy" and "zzz" are matrices (xxx and zzz are square). Although there are no doubt published ways of inverting such matrices, I came up with an exact algorithm which solves the problem neatly.

Let's consider the general case first, i.e. not assume any symmetry. We write the matrix we want to invert and its inverse as

$$V^{-1} = \begin{pmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{pmatrix}^{-1} = \begin{pmatrix} A & C \\ B & D \end{pmatrix}$$

where V_{11} and V_{22} are square matrices and assume that V_{11} is invertible. Then the equation

$$\begin{pmatrix} A & C \\ B & D \end{pmatrix} \begin{pmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

must be satisfied. Solving the implied four equations for A, B, C and D yields

$$V^{-1} = \begin{pmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{pmatrix}^{-1} = \begin{pmatrix} V_{11}^{-1} + V_{11}^{-1}V_{12}S^{-1}V_{21}V_{11}^{-1} & -V_{11}^{-1}V_{12}S^{-1} \\ -S^{-1}V_{21}V_{11}^{-1} & S^{-1} \end{pmatrix}$$

where $S = V_{22} - V_{21}V_{11}^{-1}V_{12}$. Since the only matrix that must be inverted is S (V_{11}^{-1} is already known), the dimensions of V_{22}, and hence S, should be chosen to be as small possible for maximum effectiveness. If V is symmetric then $V_{21} = V_{12}^t$.

If the matrix is such that only the element V_{22} can be inverted, the sequence of steps that led top the previous solution can be modified to give

$$V^{-1} = \begin{pmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{pmatrix}^{-1} = \begin{pmatrix} S'^{-1} & -S'^{-1}V_{12}V_{22}^{-1} \\ -V_{22}^{-1}V_{21}S'^{-1} & V_{22}^{-1} + V_{22}^{-1}V_{21}S'^{-1}V_{12}V_{22}^{-1} \end{pmatrix}$$

where $S' = V_{11} - V_{12}V_{22}^{-1}V_{21}$. The same statements about maximum effectiveness apply here: the dimensions of V_{11} should be chosen to be as small as possible.

2. Woodbury formula

Instead of factoring the matrix into 4 pieces, it is sometimes possible to write it as $A + UV^t$, where A is invertible. A is a $n \times n$ matrix and U, V are $n \times p$ matrices, where $p < n$. Our strategy, as in the previous section, is to find a way to invert the total matrix by inverting another smaller matrix of size $p \times p$.

The general expansion of the inverse of a sum of matrices $A + \delta A$ is

$$(A + \delta A)^{-1} = A^{-1} - A^{-1}\delta A A^{-1} + A^{-1}\delta A A^{-1}\delta A A^{-1} + \ldots$$

In our case $\delta A = UV^t$ so the expansion becomes

$$(A + UV^t)^{-1} = A^{-1} - A^{-1}UV^tA^{-1} + A^{-1}UV^tA^{-1}UV^tA^{-1} - \ldots$$

Looking at the expansion we can pick out a repeating unit $X = V^tA^{-1}U$ which is of size

$p \times p$. The equation now becomes

$$\begin{aligned}(A + UV^t)^{-1} &= A^{-1} - A^{-1}UV^tA^{-1} + A^{-1}UXV^tA^{-1} - A^{-1}UX^2V^tA^{-1} + \ldots \\ &= A^{-1} - A^{-1}U(1 - X + X^2 - X^3 + \ldots)V^tA^{-1} \\ &= A^{-1} - A^{-1}U(1 + X)^{-1}V^tA^{-1} \\ &= A^{-1} - A^{-1}U(1 + V^tA^{-1}U)^{-1}V^tA^{-1}\end{aligned}$$

where $(1 + V^tA^{-1}U)$ is a $p \times p$ matrix. This expression is called the Woodbury formula. It is stated without proof in Numerical Recipes, p.68.