A Description of a Beam Chamber Reconstruction Algorithm

E.P. Hartouni*

Lawrence Livermore National Laboratory, Livermore, California, USA

(Dated: June 11, 2004)

In this document I describe a beam chamber reconstruction algorithm. The algorithm has its origins in FNAL E690 experiment and provides a simple and robust procedure for finding and fitting beam tracks. There are four stages of the algorithm: chamber hit association, track finding, view matching and track fitting. The major virtue of the algorithm is its simplicity, and the application of constraints in a graduated manner from "least" to "most" which, in the past, has led to high track reconstruction efficiency.

INTRODUCTION

The algorithm I will describe here assumes three chambers with four "views". The track trajectories are straight lines. The "first chamber" has the smallest z-position (most "upstream"), the "last chamber" has the largest z-position (most "downstream"). The approach is simple to outline: for each chamber plane j there is a list of hits $\{H_i\}_j$. The planes are grouped into 4 views (referred to as u, v, s, t). The first step, *hit association*, the adjacent hits within each plane are found and grouped together.

The second step loops through all pairs of associated hits in the first and last chambers. The two wire indices are related by the geometry to the middle chamber, and when combined in a bi-linear relation predict the wire index in the middle chamber. The existence of an associated hit in the middle chamber (within some "window" or "road") is used to test if the first-last wire pair form a candidate track. For candidates which pass the test the wire indices are "tagged" and associated with the candidate in a list of candidates for that particular view. This is done independently in all four views.

The third step takes track candidates in pairs of views, usually $u - s$ and $v - t$ are sufficient. The track candidate list is looped over, now the wire indices are used to predict wire indices in the complementary views. That is, the *straight line 3D trajectory* formed from the candidate $u - s$ tracks are used to predict the wire indices in the v and t views. A search is performed in the view hit lists for the predicted hits. A test on the number of found hits is used to judge the validity of the candidate trajectories. Associated hits are tagged to the candidate trajectories. This is done in both view pairs.

The fourth step takes the candidate trajectories and their associated hits and performs a least-squares fit. This takes place in two stages, first with hit index alone, second using the TDC time information. Generally, the hit index fitting step allows hits to be "reassigned" as a function of trajectory. No more then three iterations are necessary to converge on a set of hits. (Note that the fit is still a "non-linear" fit because the hit index can change depending on the parameter values found at each iteration). Once this is done, the TDC information is used in three more iterations to refine the parameter values. A "clean up" step is performed at the end of the fitting. This step loops over all trajectory candidates and compares the parameters, if the parameters fall within some window the two candidates are declared to be "the same" and the candidate with the best "chi-squared" is chosen. This is done for all candidates.

In the end there is a list of all 3D trajectory candidates, or "tracks" supported by the hits in the three beam chambers. The remainder of the document describes the details.

HIT ASSOCIATION

Charged particles passing through the chambers ionize the gas in the vicinity of the trajectory. The most common cause of multiple wires detecting a single charged particle is the case when the particles is not perpendicular to the wire plane, but pass between two wires, ionizing the gas near both wires, see Fig. 1. In the ionization process, the kinetic energy, T, of the ionized electrons follows a $1/T^2$ distribution. The implication being that electrons can have energies high enough to cause further ionization. Most of the electrons have trajectories perpendicular to the charged particle trajectory. The very high energy electrons,(*e.g.* ones which have a discernable track length in a bubble chamber) are referred to as delta-rays. In a chamber these delta-rays can cause ionization in adjacent wires, thus a track may have multiple hits in a single chamber plane. This case is illustrated in Fig. 2.

The goal of the hit association step is to reduce the number of first and last chamber combinations. The list of hits for each view is looped over. If the there are two adjacent wires, then only one wire index is entered into the hit list.

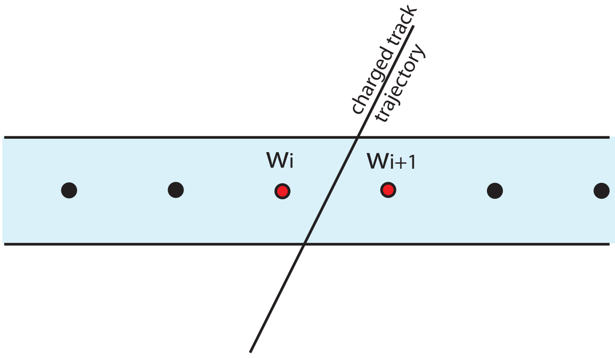

FIG. 1: **Charged track trajectory crossing two wire cells. Both wires w_i and w_{i+1} are "hits".**

If $w_{i+1} = w_i + 1$ then only the hit $w_i + 0.5$ is entered into the list. For the case shown in Fig. 1 only the one hit is entered. In the case of Fig. 2 two hits would be entered: $w_i + 0.5$ and $w_{i+2} + 0.5$. One "false" hit is generated in this last case, but the total number of hits is halved.

While this hit association algorithm appears to be very simple, the track finding algorithm must loop over all *pairs* of the first and last chamber hits. The reduction in the number of hits to pair greatly increases the speed of that part of the algorithm.

TRACK FINDING

Every pair of hits in a the first and last chamber, $\{w_{1i}\}$ and $\{w_{3j}\}$, can be used to define a line which intersects the second chamber in that view:

$$w^*_{2ij} = c_1 w_{1i} + c_2 w_{3j} \tag{1}$$

which is derived from the better known equation:

$$w^*_{2ij} = w_{3j} + \frac{w_{3j} - w_{1i}}{z_3 - z_1}(z_2 - z_3). \tag{2}$$

The definition of the constants in (1) is:

$$c_1 = \frac{z_2 - z_3}{z_3 - z_1} \tag{3}$$

$$c_2 = \frac{z_2 - z_1}{z_3 - z_1} \tag{4}$$

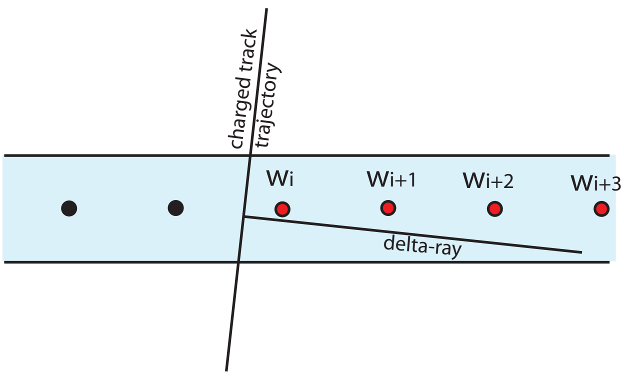

FIG. 2: **Charged track trajectory with delta-ray shown. The set of wires w_i through w_{i+3} are "hits".**

which are "pure" geometric constants. The test for the candidate pair is essentially to search a road $\pm n$ wires wide around the prediction w^*_{2ij} in the list of the middle chamber hits $\{w_{2k}\}$.

Pairs passing this test are kept in a list (w_{1i}, w_{3j}) in the track candidate list. Each of the four views has a list of such pairs of candidates which are then passed to the View Matching stage of the algorithm.

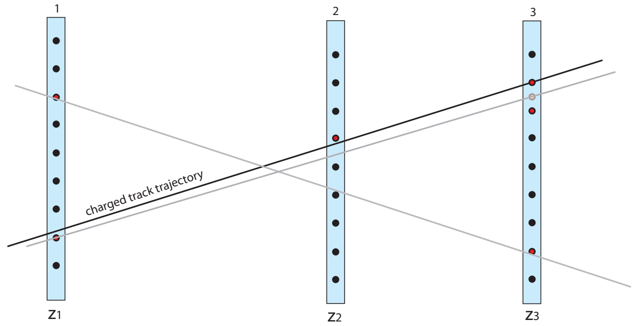

FIG. 3: **Track finding in a single view. The red centered dots indicate hit wires. The light colored "wire" is the "hit" from the hit association algorithm. The black line indicates the "true" trajectory. The gray lines indicate candidate tracks. Whether or not the candidates are kept depends on the width of the "road" used in the test.**

VIEW MATCHING

The geometry of the chambers is such that any two views gives a space point in the plane of the chamber. The wire forms a line in space at a given z_{plane} which depends on the angle of the view. The angles are defined with respect to vertical, and the assumption of "perfect" geometry is made (the chambers planes are vertical, and the chamber vertical is aligned with the coordinate system). The wire position is then given by:

$$y = x \cot \theta_u + x_{ui}, \tag{5}$$

with x_{ui} is a geometric constant associated with the definition of the center of the chamber, the wire pitch of the plane, δ, and the angle of the wires, θ_u:

$$\Delta x_u = \delta \sec \theta_u, \tag{6}$$

The wire spacing projected onto the x-axis. The two equations from the candidate wires in the two matched views are solved to give the two unknowns: the (x, y) coordinate at the chamber.

$$y = x \cot \theta_u + x_{ui} \tag{7}$$
$$y = x \cot \theta_s + x_{si}. \tag{8}$$

This is easily solved to give:

$$x = \frac{x_{si} - x_{ui}}{\cot \theta_u - \cot \theta_s} \tag{9}$$
$$y = \frac{x_{si} \cot \theta_u - x_{ui} \cot \theta_s}{\cot \theta_u - \cot \theta_s}. \tag{10}$$

This is done for all combinations of candidate tracks, here in the $u - s$ views, and also in the $v - t$ views. The two points, $(x, y)_1$ and $(x, y)_3$ define a 3D trajectory. To test the match the candidate trajectory is projected into the "other" two views, in the case above, into the v and t views. Using (6) above the wire number is given by:

$$x_{vi} = \Delta x_v \times i - x_{v0}. \tag{11}$$

The trajectory is projected as a straight line through the three chamber system:

$$x(z) = x_1 + \frac{x_3 - x_1}{z_3 - z_1}(z - z_1) \tag{12}$$

$$y(z) = y_1 + \frac{y_3 - y_1}{z_3 - z_1}(z - z_1). \tag{13}$$

In the i^{th} chamber for v, s views the predicted wire number is:

$$w^*_{i,(v,s)} = (x_{(v,s)0} + y(z_i) + x(z_i)\cot\theta_{(v,s)})/\Delta x_{(v,s)}. \tag{14}$$

A road of $\pm n$ wires (different, in general, from the track finding step above) is applied around the prediction to the lists of hit wires in the appropriate wire planes. There are 6 predictions which are made. The final test is to require that some number of predictions are found. This is usually set to at least 5 out of 6 predictions find a hit within the road around the predicted wire.

TRACK FITTING

The track fitting section takes the candidate wire pairs from both the $u - s$ and $v - t$ view pairings and performs a linearized-least-squares fit with the 12 planes of wires to four parameters. Four parameters are required to describe a straight line in 3D space. The choice of parameters is arbitrary, to some extent. In this case the choice of $(x, y)_1$ and $(x, y)_3$ is made, the two space points of the trajectory which intersects the first and last chambers. As mentioned in the introduction, the fit is non-linear because the wires used in the fit (in the first part of the fit) and the "side" to which the track is assigned (in the last part of the fit) can change as function of the parameter values.

The formal least squares expression is given by:

$$LSQ = \sum_{i=1}^{12} W_i \left(w_{0i} - w_i + \sum_{j=1}^{4} p_j c_{ij} \right)^2 \tag{15}$$

where p_j are the four parameters, W_i is a weight, w_{0i} is the origin, in wire number, for the i^{th} plane, w_i is the wire assigned to the track candidate in the i^{th} plane and c_{ij} is a set of geometric constants which transforms the track parameters into wire number. The best estimate for the parameters occurs where the LSQ is minimum with respect to variation of the parameters. Thus the set of equations in the partial derivatives of LSQ with respect to the parameters is set to zero and solved.

$$\frac{\partial LSQ}{\partial p_k} = 0 = 2\sum_{i=1}^{12} W_i c_{ik} \left(w_{0i} - w_i + \sum_{j=1}^{4} p_j c_{ij} \right) \tag{16}$$

leading to the matrix equation:

$$\begin{pmatrix} \sum W_i c_{i1}(w_{0i} - w_i) \\ \sum W_i c_{i2}(w_{0i} - w_i) \\ \sum W_i c_{i3}(w_{0i} - w_i) \\ \sum W_i c_{i4}(w_{0i} - w_i) \end{pmatrix} = \begin{pmatrix} \sum W_i c_{i1}^2 & \sum W_i c_{i1} c_{i2} & \sum W_i c_{i1} c_{i3} & \sum W_i c_{i1} c_{i4} \\ \sum W_i c2_{i2} c_{i1} & \sum W_i c_{i2}^2 & \sum W_i c_{i2} c_{i3} & \sum W_i c_{i2} c_{i4} \\ \sum W_i c2_{i3} c_{i1} & \sum W_i c_{i3} c_{i2} & \sum W_i c_{i3}^2 & \sum W_i c_{i3} c_{i4} \\ \sum W_i c2_{i4} c_{i1} & \sum W_i c_{i4} c_{i2} & \sum W_i c_{i4} c_{i3} & \sum W_i c_{i4}^2 \end{pmatrix} \times \begin{pmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{pmatrix} \tag{17}$$

where the sums here are for $i = 1, 12$. Rewritten as the matrix equation:

$$\overrightarrow{\mathbf{\Delta w}} = \underline{\mathbf{C}}\vec{\mathbf{p}} \tag{18}$$

inverting the matrix $\underline{\mathbf{C}}$ and solve for $\vec{\mathbf{p}}$ gives the solution for the parameters satisfying the conditions in (16):

$$\vec{\mathbf{p}} = \underline{\mathbf{C}}^{-1}\ \overrightarrow{\mathbf{\Delta w}}\ . \tag{19}$$

The $\overrightarrow{\mathbf{\Delta w}}$ can be found in a simple table lookup where the table is addressed by a 5-wire road bit map and the calculation of Δw the calculated - central wire prediction. The table output is given by:

$$T = \begin{cases} \min(7/8, \max(-7/8, \delta)) & |\delta| < \lim \\ 0 & |\delta| \geq \lim \\ 0 & \delta = 0, \delta \text{ changes sign, } 2 \text{ solutions} \end{cases} \tag{20}$$

It is not necessary to use a table lookup in the offline reconstruction code, but this is an illustration of applying the criteria for deciding which wire, in a multi-hit-wire road. These table outputs, which give the effective Δw which is put into (19), gives the change to the parameters, the new parameters are used to calculate new Δw, etc. There is no convergence condition given for this algorithm, three iterations are sufficient to arrive at the final value for the parameters.

It is possible to do additional iterations with the drift-times. The drift-times are used in the Δw calculation, and the iterations are performed fixing the wire, but allowing the left-right change in the assignment. Three iterations are also performed.

GEOMETRIC CONSTANTS

My experience is that the geometric constants take the longest time to get correct. Positions of the wire planes, the "roll" of the planes, and the definition of the $(x = 0, y = 0)$ point in each plane (the "origin wire number") also have to be described. Further, the calibration of the drift times is somewhat iterative, using the trajectories without the drift times to define a position in the plane, and a plot of that distance vs. the drift-time to define the position from the drift-time table.

ACKNOWLEDGMENTS

This reconstruction algorithm takes advantage of the redundant, 4-view mini-drift chambers used in the FNAL E-690 experiment. The entire experiment, and its predecessor BNL E766m where designed to be efficient, multi-particle spectrometers with open geometry. The algorithms which were developed were designed to be simple enough to be implemented in an online track reconstruction processor. The preceding discussion is based on the work of the BNL E766/FNAL E690 collaboration, much of which is contained in thesis and publications, but the majority is represented in the undocumented code associated with that experiment. See the E690 web page (`http://www-e690.fnal.gov/`) for what details are available.

---

* `hartouni1@llnl.gov`