

CHAPTER 3

IMPLEMENTATION OF THE ARBITRARY- VERTEX HOUGH TRANSFORM ALGORITHM

3.1 General Comments

The algorithm outlined in Chapter 2 for the AVHT has been implemented using the ANSI C programming language. The resulting computer code is displayed in Appendix B. It is compilable with any UNIX or IBM PC operating system and can also be used on VAX with little modification. The details of implementation are beyond the scope of this paper, but they can be gleaned from the source code itself. This chapter will elucidate the cuts implicitly employed in the AVHT, as well as some algorithms of general interest, to the purpose of providing the user with an understanding of the salient tracking parameters and an overview of the interface.

Where necessary, references to C functions and variables from the AVHT code of Appendix B will be made. When referring to these functions or variables for the first time, the file in which they are defined will be denoted in the following fashion: the function '*LeastFit*' found in file '*bjhough.c*' is equivalent to '*bjhough.c:LeastFit*'. Thereafter, they will be simply denoted by their names in italics, e.g. *LeastFit*. Usually, a description of the usage of the variable or function can be obtained from the relevant header file, *bjhough.h* in this case.

The coordinate system employed by the AVHT tracker is not the same as the standard one detailed in Chapter 1. The AVHT coordinate system was chosen to help diagonalize the covariance matrix of the LSF.¹ Because approximately half of the MWPC's have their wires in nearly parallel directions, it is natural to choose the y-axis parallel to the average orientation of these wires, yielding an approximate 45 degree counterclockwise rotation of the coordinate system away from the standard. The x-axis is taken perpendicular to the y- and z-axes to form a right-handed coordinate system² in contrast to the standard system.

¹ See Section 3.3

² The function *linalg.c:MMult* can be used in combination with the matrices *coords.c:MCBtoNew* or *coords.c:MCBtoOld* to rotate a coordinate vector to the AVHT or standard coordinate systems, respectively.

Much of the analysis in this chapter will be based upon Monte Carlo simulation. The Monte Carlo effort in Minimax has a history longer than the experiment itself; however, only the current methods are of relevance. A typical simulation of a collider run consists of a set of about 10,000 minimum-bias $p\bar{p}$ events generated by PYTHIA³. For each event, PYTHIA produces a list of collision products consisting of each particle's species and momentum-vector. These particles are then passed to GEANT⁴, which follows the particles through a complete simulation of the MiniMax geometry and emulates particle-material interactions and particle decays. GEANT also emulates the standard collider trigger so that only events where enough energy has been deposited in the C and D scintillators are kept (this usually eliminates about half of PYTHIA's minimum-bias events).

Currently, GEANT creates data summary files (DST's⁵) each consisting of a list of all particles (primary and secondary) in each event which deposit energy in at least eight MWPC's. The particle's species, point of origin and momentum-vector are recorded. GEANT also supplies data files containing the same type of information present in normal data—MWPC wires hit, energy deposition, etc. The PYTHIA-GEANT simulation team does a very good job of emulating the qualitative properties of actual data including such things as wire-hit distributions, MWPC noise, and the amount of energy deposition in the scintillators and calorimeters. They thus provide a valuable tool for fine-tuning tracking algorithms. The output of GEANT can be analyzed (tracked) and the results compared to the GEANT DST files to measure tracker efficiency and to detect systematic errors.

The primary purpose of the AVHT algorithm is to produce DST files which contain the *heads* and *tails*⁶ present in the events from a data run or a GEANT simulation. A DST file can then be analyzed without reference to the original MWPC data or the requirements for tracking. These DST files are employed throughout this and the next chapter for the purpose of analyzing efficiencies, errors and cuts; their detailed format is given in Section 3.8. The discussions in this chapter apply to both heads and tails, separately and the term *tracks* will refer to either heads or tails. Analysis schemes which relate heads and tails to each other are not discussed until Section 4.2.

³ PYTHIA version 5.702; last date of change: Feb. 13, 1994.

See also: H.-U. Bengtsson and T. Sjostrand, Computer Physics Commun. **46** (1987) 43. and CERN-TH.7112/93 for a long description of PYTHIA.

⁴ GEANT version 3.21.01; last update: June 1993.

See also: Application Software Group, GEANT Detector Description and Simulation Tool, Computing Networks Division, CERN, 1993. (CERNLIB@CERNVM.CERN.CH)

⁵ DST really stands for Data Summary Tape, but most of these are kept as files on a hard drive.

⁶ Heads are track segments formed by wires from the front eight MWPC's. Tails are the corresponding segments formed by wires from the sixteen MWPC's after the PbC. The presence of the PbC necessitates treating the track segments on either side separately.

3.2 The Minimum Planes Cut

Section 2.2 briefly mentioned the necessity for having a lower bound on the number of planes that can contribute to an AVHT tube. This MinPlanes Cut is perhaps the most conceptually simple cut on acceptable AVHT tubes. It requires that the number of planes with active wires intersecting a tube be greater than some specified value, *bjhough.c:MinPlanes*. The value of this cut depends both on the average efficiency of the MWPC's and on the occupancy (average number of active wires per MWPC) of a given event.

If only the efficiencies of the chambers are considered, the probability that a track activates wires in a given number of chambers can be easily determined. If the average efficiency is denoted ε and N planes are being used for track reconstruction, then the probability that the particle activates at least one wire on M of N planes is

$$P(\varepsilon, N, M) = \binom{N}{M} \varepsilon^M (1 - \varepsilon)^{N-M}. \quad (3.1)$$

M=	Probability	M=	Probability
Planes Hit	N=16	Planes Hit	N=8
16	0.44012667	8	0.66342043
15	0.37063298	7	0.27933492
14	0.14630249	6	0.05145643
13	0.03593395	5	0.00541647
12	0.0061466	4	0.00035635
11	0.00077641	3	1.5004E-05
10	7.4917E-05	2	3.9484E-07
9	5.6329E-06	1	5.9375E-09

Table 3-1: Probability of a Particle Activating Wires on M/N MWPC's

If ε is taken to be 0.95, which is a good approximation to reality, then Table 3-1 shows the probabilities of a particle activating wires in M of N chambers for relevant values of N .

Clearly, this probability falls off very rapidly with the number of chambers hit. This is fortunate because an arbitrarily small *MinPlanes* value cannot be used if excessive spurious tracks are to be avoided. The computation of the expected number of spurious tracks is very difficult in any approximation; however, a qualitative estimate can be made by assuming a uniform distribution of active wires in the chambers⁷. If the mean chamber occupancy is taken to be w wires, then the average number of spurious tracks per event forming through the coincidence of wires is proportional to

⁷ The distribution is actually more like $1/r$, where r is the distance from the TeV beam axis. The probability that a wire is active is then computed by integrating each of the wires in a plane over r (assuming each wire is 0.1" wide in the MWPC) and normalizing the distribution. However, for the above argument, this level of detail does not add much to the qualitative results.

$$S(u = w / 128, N, M) = \binom{N}{4} \binom{N-4}{M-4} u^M (1-u)^{N-M} 128^4. \quad (3.2)$$

Equation (3.2) is only a proportionality because the spurious tracks must be constrained to actually pass through the physical MWPC planes. However in reality, most spurious tracks are caused by coincidence of wires with sets of wires from an actual track. In the current MiniMax configuration, approximately half the MWPC's have parallel wires, so each real track passing through leaves a geometric plane containing about $N/2$ active wires. The average number of spurious tracks per event that will be formed in this plane is proportional to

$$P(u, N, M) = \binom{N/2}{M - N/2} u^{M-N/2} (1-u)^{N-M}, \quad (3.3)$$

which, taking the constants of proportionality to be similar, is much larger than (3.2). Also, as incident particle multiplicity increases, this will clearly grow. It should be noted in relation to (3.3) that in tracking heads, any real track passing through the detector will leave four active wires in the same geometric plane. *Any* two additional wires on two other planes are then sufficient to define a line intersecting all six wires with a near perfect χ_v^2 . Therefore, a *MinPlanes* cut of at least seven is required for tracking heads. In any case, it is clear that M should be as large as possible in order that (3.2) and (3.3) remain as small as possible.

A reasonable minimum value for *MinPlanes* is determined by taking the value from Table 3-1 for which there is approximately a 95% chance that a charged track will activate wires on M or more planes. This criteria yields theoretical cuts of seven and fourteen for tracking heads and tails, respectively. These cuts will be implicitly assumed in all further discussions. One should note that GEANT currently assumes perfectly efficient wires, so that inefficiencies resulting from a *MinPlanes Cut* will not be evident in any GEANT-based analysis.

3.3 The Least-Squares Fit

An examination of Fig. 2-5 shows that the linear LSF of a set of wires to a straight line is central to the AVHT. The definition of a track will be taken as (2.1), in the AVHT coordinate system, where the track parameters

$$(x_0, y_0, x_1, y_1) \quad (3.4)$$

and their respective variances

$$(\sigma_{x_0}^2, \sigma_{y_0}^2, \sigma_{x_1}^2, \sigma_{y_1}^2) \quad (3.5)$$

are to be determined from the fit. The parameters (z_0, z_1) , representing the z-coordinates of the AVHT projective planes, are chosen to be (z-position of the first MWPC, z-position of the PbC), for tracking before the PbC, and (z-position of the PbC, z-position of the last MWPC), for tracking after the PbC. Such a choice allows easy matching of “heads” (tracks before the lead) and “tails” (tracks after the lead) to form tracks and showers that span the entire MWPC array. It also represents the smallest distance between projective planes that incorporates all MWPC’s of interest. This is important since the minimum area of the projective planes, for a given set of MWPC’s, scales as the distance between them.

If the (x, y, z) position of one end of the w th wire of a given set of W wires is represented by $(Px[w], Py[w], Pz[w])$ and the unit vector perpendicular to that wire in the direction of increasing wire number on its MWPC plane is $(Ux[w], Uy[w], 0)$, then the “merit function” relating the track parameters to the set W is given by

$$\chi^2 = \frac{1}{\sigma^2} \sum_{w \in W} \left[\{x_0 + (x_1 - x_0)Z - Px[w]\}Ux[w] + \{y_0 + (y_1 - y_0)Z - Py[w]\}Uy[w] \right]^2, \quad (3.6)$$

$$Z = \frac{Pz[w] - z_0}{z_1 - z_0}. \quad (3.7)$$

The expected standard deviation in the distance between a wire and the particle’s path, σ , is 0.029,” as can be verified by a simple Monte Carlo calculation. Because the distance between MWPC’s is large compared to the inter-wire spacing and since successive MWPC’s are not oriented in exactly the same direction, (3.6) assumes that the standard errors in each wire are uncorrelated and thus can each be taken as σ . By following the method of maximum likelihood minimization⁸, the derivatives of (3.6) with respect to each of the parameters (3.4) are set equal to zero. These four conditions are equivalent to a four-dimensional matrix equation

$$\left(\sum_{w \in W} M[w] \right) X = \sum_{w \in W} B[w] \quad (3.8)$$

for the optimal parameters

⁸ Bevington, Philip R. Data Reduction and Error Analysis for the Physical Sciences. (New York: McGraw-Hill Book Company, 1969) 100.

$$X = \begin{bmatrix} x_0 \\ y_0 \\ x_1 \\ y_1 \end{bmatrix}, \quad (3.9)$$

where

$$M[w] = \frac{1}{\sigma^2} \begin{bmatrix} Ux[w]^2(1-Z)^2 & Ux[w]Uy[w](1-Z)^2 & Ux[w]^2(1-Z)Z & Ux[w]Uy[w](1-Z)Z \\ Ux[w]Uy[w](1-Z)^2 & Uy[w]^2(1-Z)^2 & Ux[w]Uy[w](1-Z)Z & Uy[w]^2(1-Z)Z \\ Ux[w]^2(1-Z)Z & Ux[w]Uy[w](1-Z)Z & Ux[w]^2Z^2 & Ux[w]Uy[w]Z^2 \\ Ux[w]Uy[w](1-Z)Z & Uy[w]^2(1-Z)Z & Ux[w]Uy[w]Z^2 & Uy[w]^2Z^2 \end{bmatrix} \quad (3.10)$$

$$B[w] = \frac{1}{\sigma^2} \begin{bmatrix} (1-Z)(Ux[w]^2 Px[w] + Ux[w]Uy[w]Py[w]) \\ (1-Z)(Uy[w]^2 Py[w] + Ux[w]Uy[w]Px[w]) \\ Z(Ux[w]^2 Px[w] + Ux[w]Uy[w]Py[w]) \\ Z(Uy[w]^2 Py[w] + Ux[w]Uy[w]Px[w]) \end{bmatrix}. \quad (3.11)$$

Equation (3.8) has the solution

$$X = C \left(\sum_{w \in W} B[w] \right) \quad (3.12)$$

where C is the inverse of $\left(\sum_{w \in W} M[w] \right)$. The so-called covariance matrix, C , contains the variance (3.5) and covariance terms between the parameters⁹. The goodness of fit can be determined from the square-residual per degree of freedom,

$$\chi_v^2 = \frac{\chi^2}{n-4}, \quad (3.13)$$

where ‘ n ’ is the number of wires used in the fit. The implications of using χ_v^2 as a goodness of fit criterion are examined in Section 3.6. The magnitudes of the variance

⁹ Press, Teukolsky, Vetterling and Flannery. Numerical Recipes in C, Second Edition. New York: Cambridge University Press, 1992) 673.

terms can also be analyzed to determine the confidence levels of the fitted track parameters.

A Monte Carlo simulation was used to determine the expected distribution of the elements of C for tracking heads and tails. Since C does not depend upon which wires were actually activated in a given plane, the distribution of the elements in C obtained from a LSF¹⁰ of GEANT tracks should reproduce the actual distributions obtained in practice. Matrices (3.14) display the mean values of C (in units of inches²) for GEANT head and tail tracks, respectively.

$$\begin{bmatrix} 0.000631 \pm 0.000030 & -0.000040 \pm 0.000050 & -0.000305 \pm 0.000021 & 0.000045 \pm 0.000033 \\ -0.000040 \pm 0.000049 & 0.000564 \pm 0.000178 & 0.000045 \pm 0.000021 & -0.000427 \pm 0.000132 \\ -0.000305 \pm 0.000021 & 0.000045 \pm 0.000021 & 0.000685 \pm 0.000058 & -0.000045 \pm 0.000058 \\ 0.000045 \pm 0.000033 & -0.000427 \pm 0.000132 & -0.000045 \pm 0.000058 & 0.002060 \pm 0.000459 \end{bmatrix} \quad (3.14a)$$

$$\begin{bmatrix} 0.000371 \pm 0.000532 & 0.000111 \pm 0.001197 & -0.000124 \pm 0.000092 & -0.000096 \pm 0.000531 \\ 0.000111 \pm 0.001197 & 0.016932 \pm 0.005762 & -0.000125 \pm 0.000456 & -0.013610 \pm 0.002155 \\ -0.000124 \pm 0.000092 & -0.000125 \pm 0.000456 & 0.000188 \pm 0.000049 & 0.000184 \pm 0.000340 \\ -0.000096 \pm 0.000531 & -0.013610 \pm 0.002155 & 0.000184 \pm 0.000340 & 0.030090 \pm 0.004000 \end{bmatrix} \quad (3.14b)$$

The first thing to notice about (3.14) is the relatively small size of the mean variances in the parameters for heads and tails compared to the 0.1” spacing between MWPC wires. The differences in the values for heads and tails do not stem so much from the number of contributing chambers as from the actual orientations of the chambers. Referring to Table A-2, Appendix A, it is seen that in front of the PbC, the wires in “non-parallel” chambers are oriented at small angles with the x-axis. However, after the PbC, the non-parallel chambers are all oriented at small angles to the y-axis, effectively increasing the resolution along the x-axis and decreasing it along the y-axis.

The values given in (3.14) can be checked numerically. By applying the AVHT algorithm to the events from the GEANT simulation¹¹ and comparing the parameters from

¹⁰ The function employed to perform a LSF is *bjhough.c:LeastFit*. It employs the Numerical Recipes in C function *nrc.c:gaussj* to actually solve (3.8). It also relies upon the precomputed values of $M[w]$ and the coefficients in $B[w]$ that are determined by *bjhough.c:MakeLSFData*.

¹¹ In order to achieve greatest accuracy, the GEANT events examined had only a single head or tail. The AVHT DST used was based on the standard *MinPlanes Cut*, a χ_v^2 cut of 6.0, no *Same-Track Cut*, no *Source Cut*, but a total occupancy cut of 200 wires or less—a very loose scheme. There were 1395 1-head events for which 5073 different tracks were found; there were 1218 1-tail events for which 13,948 tracks were found. In order to reduce the contribution from spurious tracks, tracks in the AVHT any parameter more than 1.5” off from the respective parameter in the GEANT track were ignored.

a LSF to the GEANT tracks to the parameters from the tracks found from the AVHT, the sample covariance matrices for heads and tails can be obtained:

$$\begin{bmatrix} 0.007168 & 0.001208 & -0.002905 & 0.001163 \\ 0.001208 & 0.022434 & -0.000033 & -0.013472 \\ -0.002905 & -0.000033 & 0.005105 & -0.003981 \\ 0.001163 & -0.013472 & -0.003981 & 0.038234 \end{bmatrix} \quad (3.15a)$$

for heads and

$$\begin{bmatrix} 0.001439 & 0.003864 & -0.000528 & -0.002698 \\ 0.003864 & 0.092374 & -0.001170 & -0.062805 \\ -0.000528 & -0.001170 & 0.000733 & 0.002100 \\ -0.002698 & -0.062805 & 0.002100 & 0.106902 \end{bmatrix} \quad (3.15b)$$

for tails. The elements in these matrices (in units of inches²) represent the sample variances and covariances obtained in practice and are larger than the expected values given in (3.14). The difference can be explained by noting that (3.14) represents the covariances resulting from actually *measuring* a track with a set of wires. However in reality, the AVHT is *guessing* which wires might represent a track. The errors involved in this guessing process, as shown in (3.15), are larger than the measurement errors. So, when examining tracks from a GEANT DST, the covariance matrix is just C ; however, for tracks obtained via the AVHT, the covariance matrix cannot be determined exactly and (3.15) is the best approximation available—even though fluctuations similar to those in (3.14) are expected. However, when employing tracks in the GEANT DST to compare with AVHT tracks or to simulate AVHT tracks, the errors (3.15) should be employed. The expected standard errors in the track parameters are determined from (3.15):

$$(\bar{\sigma}_{x_0} = 0.085", \bar{\sigma}_{y_0} = 0.150", \bar{\sigma}_{x_1} = 0.071", \bar{\sigma}_{y_1} = 0.196") \quad (3.16)$$

for heads and

$$(\bar{\sigma}_{x_0} = 0.038", \bar{\sigma}_{y_0} = 0.304", \bar{\sigma}_{x_1} = 0.027", \bar{\sigma}_{y_1} = 0.327") \quad (3.17)$$

for tails. Note that these errors are larger in the y-direction than in the x-direction. This is, as previously mentioned, purely a geometric effect resulting from MWPC orientations.

In practice, the LSF is modified somewhat from the ideal situation just discussed. Often, several, n , adjacent wires on the same plane can contribute to the LSF.

Technically, these wires do not represent separate measurements of the track on the plane and thus should not be treated with the same weight in the fit as wires from other planes. Taken together, the n wires should have the same weight in the LSF as any measurement on any other plane; therefore, each wire should be included with a weight of $1/n$, or, equivalently, only the average wire number of these n wires should be used. The latter is actually done internally by the AVHT LSF functions and has been seen to yield a nicer χ_v^2 distributions.

3.4 Probability Distributions

Many parameterized cutoffs employed by the AVHT that will be discussed in the following sections are essentially probabilistic determinations. The formal definitions and approximations used must be developed before continuing.

Calculation of the n -dimensional Gaussian probability distribution is of specific interest. Given a measurement, \vec{x} , the probability that it came from the Gaussian distribution centered around $\vec{\mu}$ with a covariance matrix, C , representing the standard variances and covariances in the distribution, is given by

$$P_{nD}(y, C) = \frac{\sqrt{\det(C^{-1})}}{\sqrt{2^n \pi^n}} \int_{\vec{z} \cdot C^{-1} \cdot \vec{z}^t > y^2} \exp\left(-\frac{\vec{z} \cdot C^{-1} \cdot \vec{z}^t}{2}\right) d\vec{z} \quad (3.18)$$

$$y = \sqrt{(\vec{x} - \vec{\mu}) \cdot C^{-1} \cdot (\vec{x} - \vec{\mu})^t} . \quad (3.19)$$

Eq. (3.19) gives the number of standard deviations \vec{x} is from $\vec{\mu}$. One can always change coordinates in (3.18) to $\vec{u}^t = M \cdot \vec{z}^t$ such that $(M^{-1})^t \cdot C^{-1} \cdot (M^{-1}) = I$. This transformation simplifies (3.18) to

$$P_{nD}(y) = \frac{1}{\sqrt{2^n \pi^n}} \int_{|\vec{u}|^2 > y^2} \exp\left(\frac{-\vec{u} \cdot \vec{u}^t}{2}\right) d\vec{u} , \quad (3.20)$$

a function independent of the covariance matrix. Angular integration of (3.20) yields

$$P_{nD}(y) = \frac{1}{\sqrt{2^{n-2}}} \frac{1}{(n/2 - 1)!} \int_y^\infty \exp\left(\frac{-r^2}{2}\right) r^{n-1} dr . \quad (3.21)$$

The integral (3.21) is easy to evaluate for n-even:

$$P_{nD, n\text{-even}}(y) = e^{-y^2/2} \sum_{k=0}^{\frac{n-2}{2}} \frac{y^{2k}}{2^k k!} \quad (3.22)$$

$$P_{2D}(y) = e^{-y^2/2} \quad (3.23)$$

$$P_{4D}(y) = e^{-y^2/2} \left(1 + \frac{y^2}{2}\right) \quad (3.24)$$

Eq. (3.22) gives the *exact* probability corresponding to (3.18).

The odd-dimensional cases are somewhat more difficult, yielding equations related to $P_{1D}(y)$. The integration of (3.18) for the one-dimensional distribution yields error-functions—not trivially evaluated numerically. However, the integral can be approximated with a fair degree of accuracy. The derivation begins with choosing a mean, μ , and standard deviation, σ , of the one-dimensional distribution. Then, analogous to (3.18), the probability that a single measurement, x , comes from this distribution is given by

$$P_{1D}(y = \frac{|x - \mu|}{\sigma}) = \sqrt{\frac{2}{\pi}} \int_y^{\infty} \exp(-\frac{z^2}{2}) dz. \quad (3.25)$$

For small y , $P_{1D}(y)$ can be approximated by

$$P_{1D}(y) \approx 1 - \sqrt{\frac{2}{\pi}} \int_0^y (1 - \frac{z^2}{2}) dz \quad (3.26)$$

$$P_{1D}(y) \approx 1 - \sqrt{\frac{2}{\pi}} y + \frac{1}{6} \sqrt{\frac{2}{\pi}} y^3. \quad (3.27)$$

As shown in Fig. 3-2, (3.27) is only accurate to about $y = 1$; however, beyond this region another approximation can be made. By changing variables to $u = z - y$, (3.25) becomes

$$P_{1D}(y) = \sqrt{\frac{2}{\pi}} \int_0^{\infty} \exp(-\frac{u^2 - 2uy - y^2}{2}) du. \quad (3.28)$$

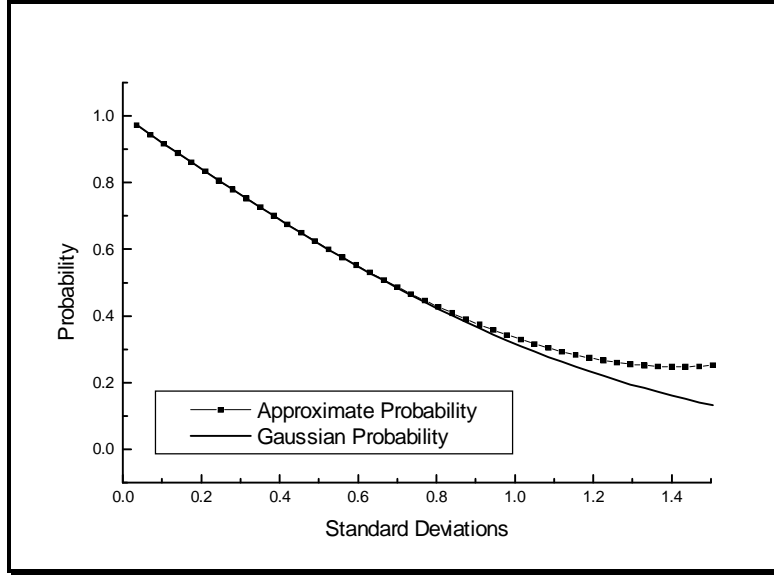


Fig. 3-2: Small y Approximation to the Gaussian Probability Distribution

For large y, (3.28) can be approximated by replacing u^2 with uy , evaluating the integral and getting

$$P_{1D}(y) \approx \sqrt{\frac{8}{9\pi}} \frac{\exp(-y^2/2)}{y}. \quad (3.29)$$

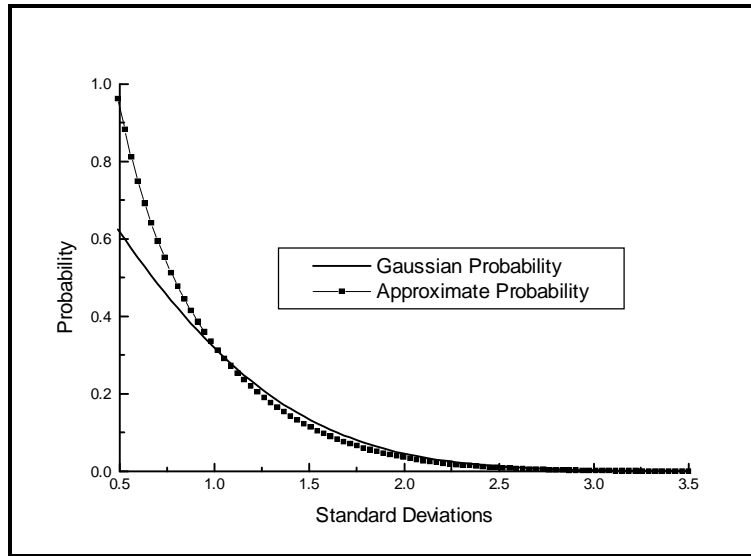


Fig. 3-3: Large y Approximation to the Gaussian Probability Distribution

Fig. 3-3 indicates that (3.29) is a good approximation for $y \geq 1$, though not quite as accurate as its small- y counterpart in its domain. The combination of Eqs. (3.27) and (3.29), as compared to the exact solution of (3.25) in Fig. 3-4, shows that they

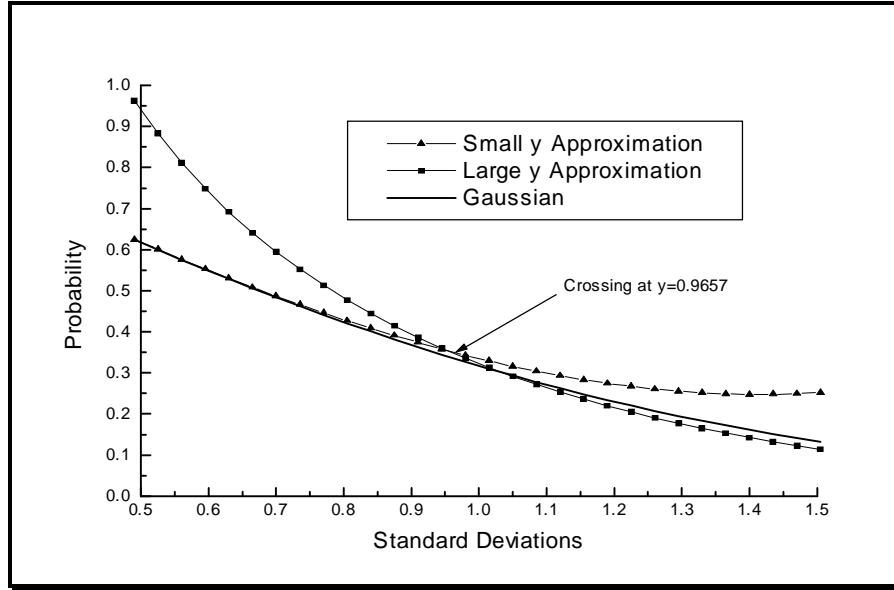


Fig. 3-4: Joining the Approximations

approximate the one-dimensional Gaussian probability distribution fairly well throughout the entire range of y with a transition point at $y=0.9657$. The approximate solution to (3.25) is, then,

$$P_{1D}(y) \approx \left\{ \begin{array}{l} 1 - \sqrt{\frac{2}{\pi}}y + \frac{1}{6}\sqrt{\frac{2}{\pi}}y^3 \text{ for } y \leq 0.9657 \\ \frac{\sqrt{8}}{\sqrt{9\pi}} \frac{\exp(-y^2/2)}{y} \text{ for } y > 0.9657 \end{array} \right\}. \quad (3.30)$$

A situation of great practical interest arises when one wishes to determine whether two measurements, \vec{x} and \vec{w} , with their respective covariance matrices, C_x and C_w , are actually measurements from the same underlying distribution (the parameters of the actual distribution being unknown). The probability that the measurements are “the same” is defined in terms of the difference $\vec{y} = \vec{x} - \vec{w}$ and the respective covariance matrix

$$(C_y)_{ij} = \frac{\partial \vec{y}}{\partial \vec{x}_i} \frac{\partial \vec{y}}{\partial \vec{x}_j} (C_x)_{ij} + \frac{\partial \vec{y}}{\partial \vec{w}_i} \frac{\partial \vec{y}}{\partial \vec{w}_j} (C_w)_{ij} \quad (3.31)$$

or,

$$C_y = C_x + C_w. \quad (3.32)$$

obtained through propagation of errors. Then, this probability is given by

$$P_{same}(\vec{y}, C_y) = P_{nD}(\sqrt{\vec{y} \cdot C_y^{-1} \cdot \vec{y}^t}) \quad (3.33)$$

and shall be referred to as the *probability of coincidence*.

3.5 Identifying Equivalent Tracks

As mentioned in Chapter 2, the AVHT is likely to detect the same track in different tubes. It is not difficult to see that often the AVHT will detect tracks that are nearly the same—differing by only a couple of contributing wires. Until removed, these superfluous tracks represent a serious class of spurious tracks. To remove them, a method must be developed to determine whether two tracks are really just different approximations to the trajectory of the same particle.

The methods previously employed have unacceptable shortcomings. Comparison of the number of wires common to the two tracks, while the simplest method, does not account for the fact that two tracks with many different wires can each have very similar trajectories as determined by a LSF. A more sophisticated method looks at the track parameters (3.4) for two tracks and defines the metric distance

$$d = \sqrt{(x_0 - x'_0)^2 + (y_0 - y'_0)^2 + (x_1 - x'_1)^2 + (y_1 - y'_1)^2}, \quad (3.34)$$

where the primed and unprimed parameters represent two separate tracks. Equation (3.34) determines how close the two tracks are in physical space as they pierce the (z_0, z_1) planes. The metric distance criterion can be improved upon by taking the errors in the tracks parameters into account—effectively using all available track information.

The actual probability that two sets of track parameters, denoted \vec{p} and \vec{p}' , with individual covariance matrices, C_p and $C_{p'}$, are really two measurements of the same underlying track can be calculated by (3.33). In the present case, the covariance matrices

for each set of tracks are equivalent and taken to be (3.15), so the probability of coincidence is simply

$$P_{same}(\vec{p}, \vec{p}', C_p) = P_{4D} \left(\frac{\sqrt{(\vec{p} - \vec{p}') \cdot C_p^{-1} \cdot (\vec{p} - \vec{p}')^t}}{\sqrt{2}} \right). \quad (3.35)$$

In order to use (3.35) to determine whether two tracks are equivalent, some understanding of the distributions of $P_{same}(\vec{p}, \vec{p}', C_p)$ for tracks that either are or are not actually redundant must be obtained. The GEANT DST can be used to determine the expected probability distribution for real tracks—ones that generally do not possess the same trajectory. For each event in the GEANT DST, $P_{same}(\vec{p}, \vec{p}', C_p)$ for each pair of tracks is determined¹². The distributions are then compiled over all events. To obtain the best approximation of what the AVHT tracker will encounter, the covariance matrices (3.15) are used in the computation of (3.35); the results are displayed in Figs. 3-5 and 3-6.

$P_{same}(\vec{p}, \vec{p}', C_p)$ is generally small, therefore, the quantity $S = -\ln(P_{same}(\vec{p}, \vec{p}', C_p))$ is used instead as a measure. The graphs in this section refer to the “integrated normalized distribution;” the vertical-coordinate on the graphs corresponding to a given S -value, s , represents the percentage (100% = 1 normalization) of track-pairs with an S -value less than or equal to s . For example, in Fig. 3-5, approximately 2.5% of all pairs from 5-head events have $S < 5.0$. One should note that events with track-multiplicity n will contribute $n(n-1)/2$ track pairs to their respective distributions. The salient features of these distributions include the initial leveling off (or cresting) of the curves and their subsequent slope. The former characteristic gives a strong indication as to when most tracks that are actually measurements of the same track have been taken into consideration. The latter gives a measure of the effective “interaction” between tracks that are and are not redundant. The flatter the slope, the better—in the sense that the probability of coincidence for tracks that are redundant is more localized on the graph and contains less contamination from pairs of non-redundant if the slope is flatter.

Figs. 3-5 and 3-6 prominently show that the probability of coincidence of real tracks is small except for some cases (~3.5%) where pairs of particles happen to travel very close together. It will turn out that many occurrences of pairs with small S -values are beyond the resolution of the MiniMax detector. The above figures seem to indicate that the distributions are qualitatively independent of multiplicity. This is of primary importance in the justification of any cut (in this case, called the *Same-Track Cut*) on S -values used on low- and high-multiplicity events. However, this conjecture is incorrect. A qualitative estimate of the number of tracks lost per event due to a *Same-Track Cut* of s can be obtained by first assuming that at most one track will be lost in any given event. Then, the expected number of tracks lost per event is $f(s, n)n(n-1)/2$, where $f(s, n)$ is the integrated normalized distribution of the probability of coincidence as a function of

¹² The function `bjhough.c:SameTrackProb` determines the probability of coincidence for two heads or two tails. It returns the S -value.

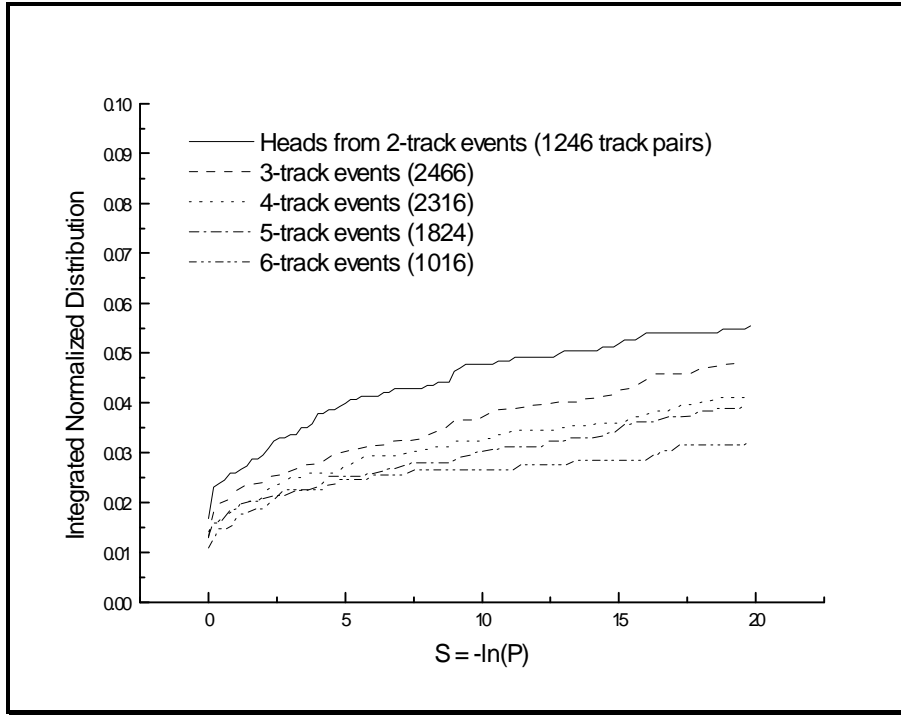


Fig. 3-5: Integrated Probability of Coincidence Distribution for Real Heads

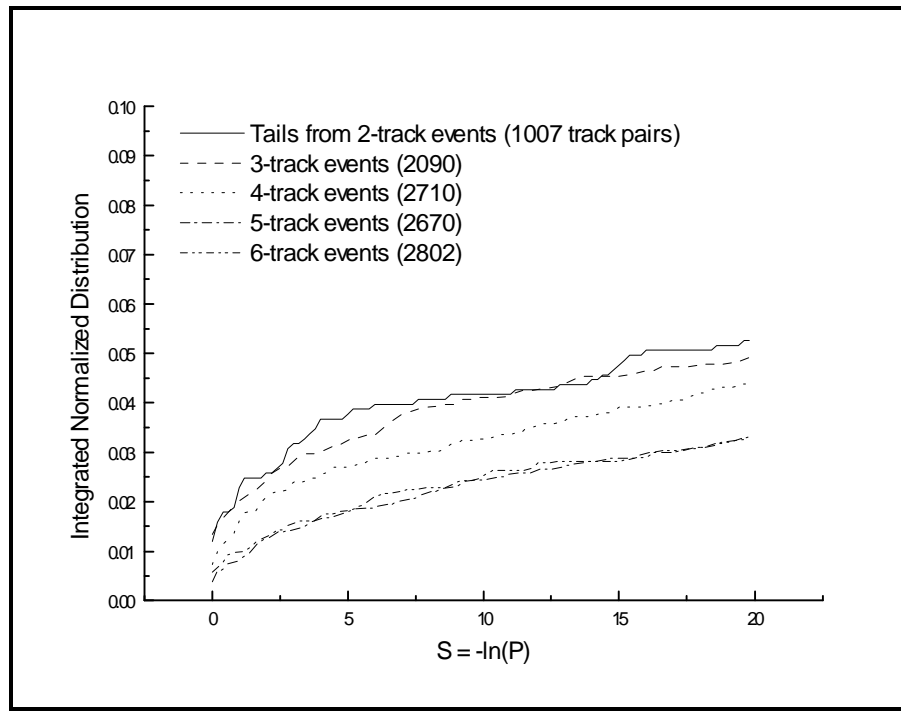


Fig. 3-6: Integrated Probability of Coincidence Distribution for Real Tails

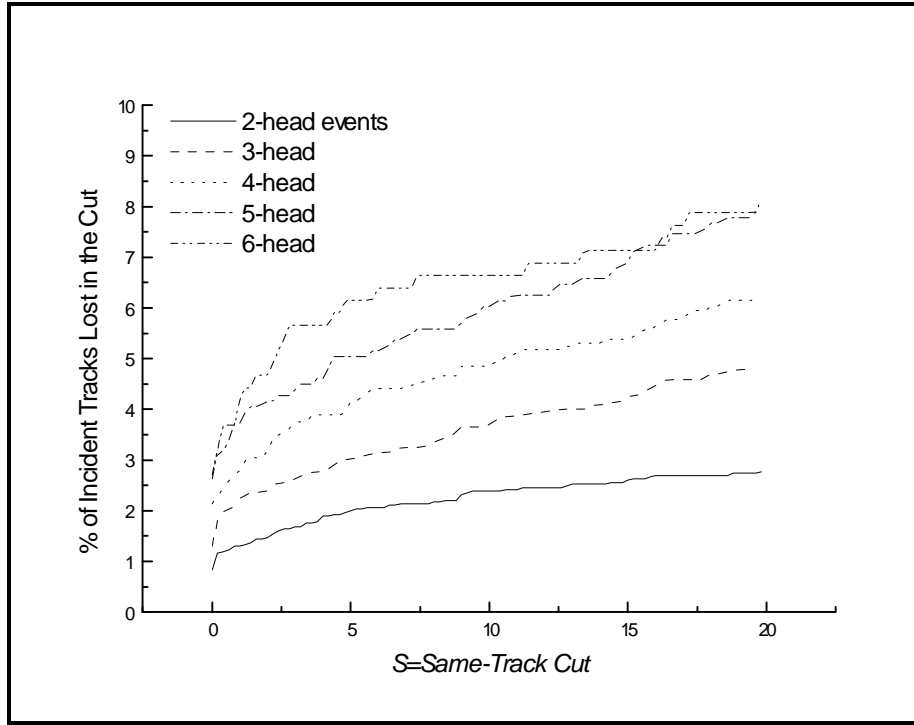


Fig 3-7. Expected Percentage of Heads Lost Due to a *Same-Track Cut*

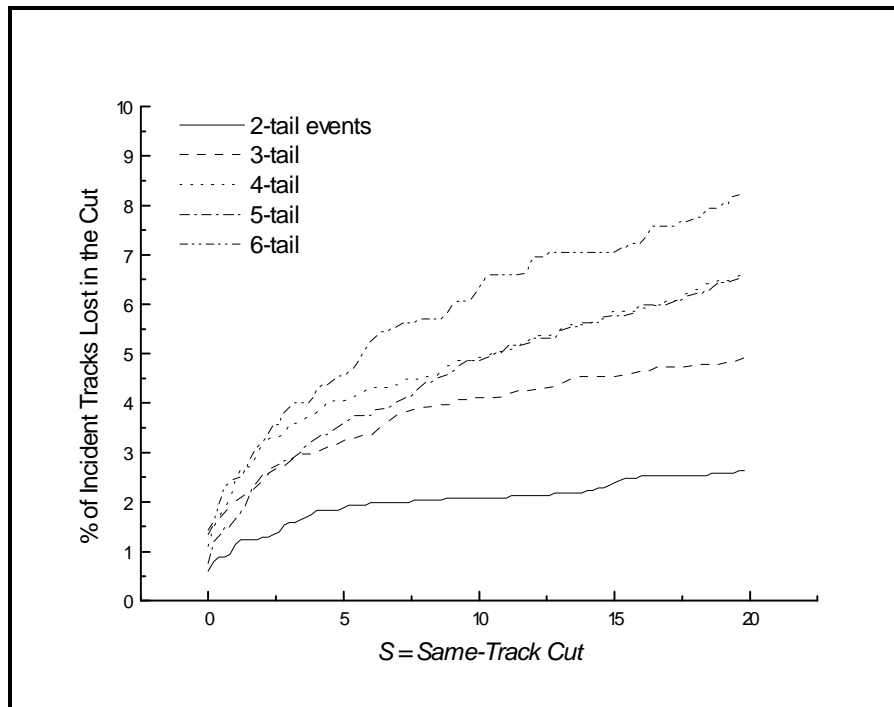


Fig 3-8. Expected Percentage of Tails Lost Due to a *Same-Track Cut*

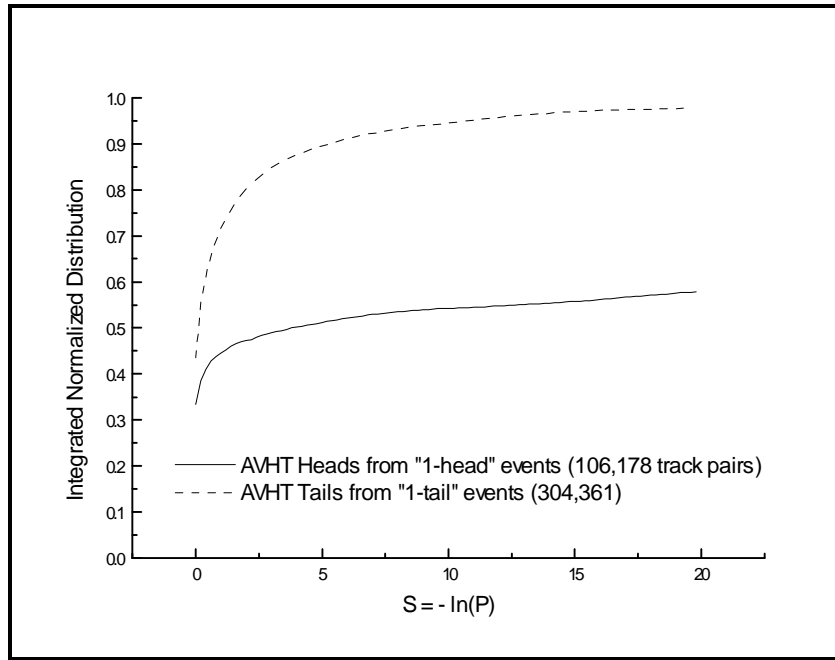


Fig. 3-9: Distribution of the Integrated Probability of Coincidence for Tracks Found in Single-Track Events

event multiplicity and $n(n-1)/2$ is the number of distinct track pairs in a multiplicity- n event. The data in Figs 3-5 and 3-6 can now be employed to estimate the probable percentage of incident tracks lost as a function of the *Same-Track Cut*: Figs. 3-7 and 3-8.

The application of a *Same-Track Cut* of 2.5 for heads, to be justified later, results in the loss of 1.5% to 5.0% of the incident tracks, with the percentage increasing with event multiplicity. For tails, a *Same-Track Cut* of 5.0 is used and approximately the same number of tracks are expected to be lost as a result. Even though these are fairly large percentages, some of the tracks lost are uninteresting (soft shower products). The actual efficiencies due to this cut are evaluated in Chapter 4. One can see from the shape of the distributions in Figs. 3-7 and 3-8 that a smaller *Same-Track Cut* will yield an improved efficiency in terms of the number of tracks lost; however, a lower bound on the cut can be obtained from the analogous distributions for tracks that are redundant, Fig. 3-9.

In contrast to Figs. 3-5 and 3-6, the probability of coincidence distribution for redundant AVHT tracks can be obtained directly by applying the AVHT to events from GEANT that contain only one track. Except for an inevitable admixture of spurious tracks, all tracks found in these events should be redundant. The distributions obtained in this way are shown in Fig. 3-9. Note that no *Same-Track Cut* was employed in the creation of these distributions, however only events with 200 or less total active wires were considered in order to keep the number of tracks found manageable (3743 events were examined).

That the distributions do not rise immediately to one indicates the presence of spurious tracks contaminating the AVHT DST. These spurious tracks also account for the slight slope in the tails of the distributions. While the contamination may seem large, most of these tracks are heads that do not point towards either C0 or the beam pipe and thus can be removed by an origin cut (see Section 4.1). The spurious track contribution for the tails is relatively small, mainly because the *MinPlanes Cut* is so much higher after the lead. Figs. 3-5 through 3-9 indicate that redundant tracks have a probability of coincidence distribution whose overlap with that for real tracks is fairly small. Therefore, the probability of coincidence can provide an effective method for removing redundant tracks. The *Same-Track Cut* that should be employed represents the *S*-values by which the distributions in Fig. 3-9 have crested. These values can be estimated qualitatively by approximating the tails of the distributions by straight lines and looking where they break from the initial rise in the curve. This gives a *Same-Track Cut* of approximately 2.5 for heads and 5.0 for tails. Tracks pairs whose *S*-value is less than this can be grouped together and called the same. In practice, the track in each such group with the best χ_v^2 is kept as the representative.

One application for the *Same-Track Cut*, other than for eliminating redundant tracks, consists of comparing the AVHT output with the GEANT DST file. There is no a priori reason to believe that the *Same-Track Cut* should be exactly the same for comparing GEANT DST tracks and tracks found by the AVHT. Therefore, all pairs of GEANT DST and AVHT DST¹³ tracks for each event were compared to determine their probability of coincidence distributions: Figs 3-10 and 3-11.

Features seen in previous figures are also shown below in Figs. 3-10 and 3-11. Most similar tracks have been accounted for by *S*=1. Since the distributions crest very early and have virtually zero slope, the *Same-Track Cut* on the probability of coincidence will clearly work quite well for determining whether a given track from the AVHT DST corresponds to a given track in the GEANT DST. This cut can also be seen not to depend on the number of tracks actually present in the event since all the curves level off at after similar *S*-values and have comparable subsequent slopes.

¹³ This AVHT DST was created using a *Chi-Squared Cut* of 6.0 and *Same-Track Cuts* of 2.5 and 5.0 for heads and tails, respectively; however, only events with less than 200 total active wires were examined.

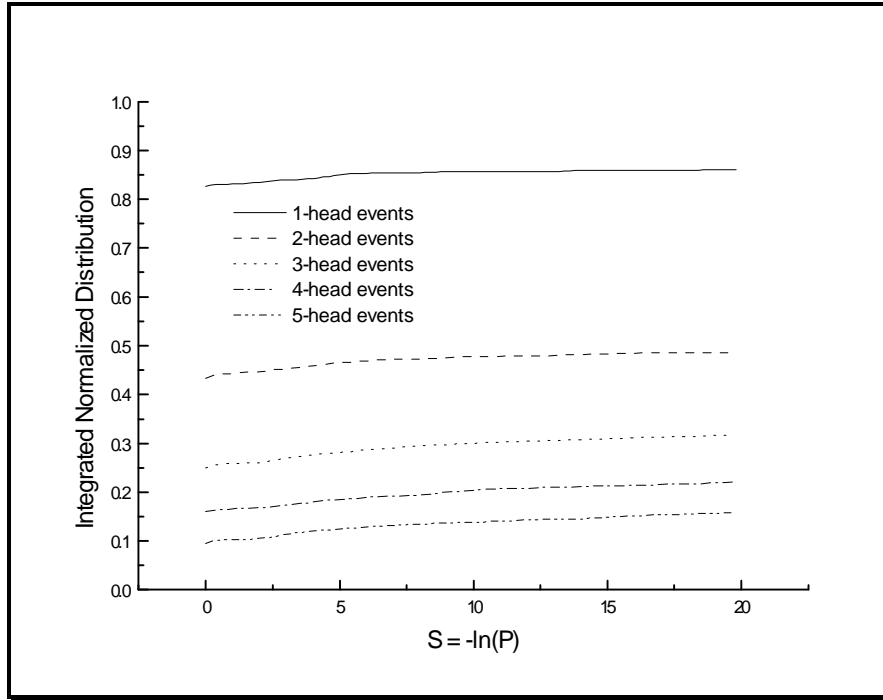


Fig. 3-10: Integrated Probability of Coincidence Distribution for AVHT-GEANT Heads

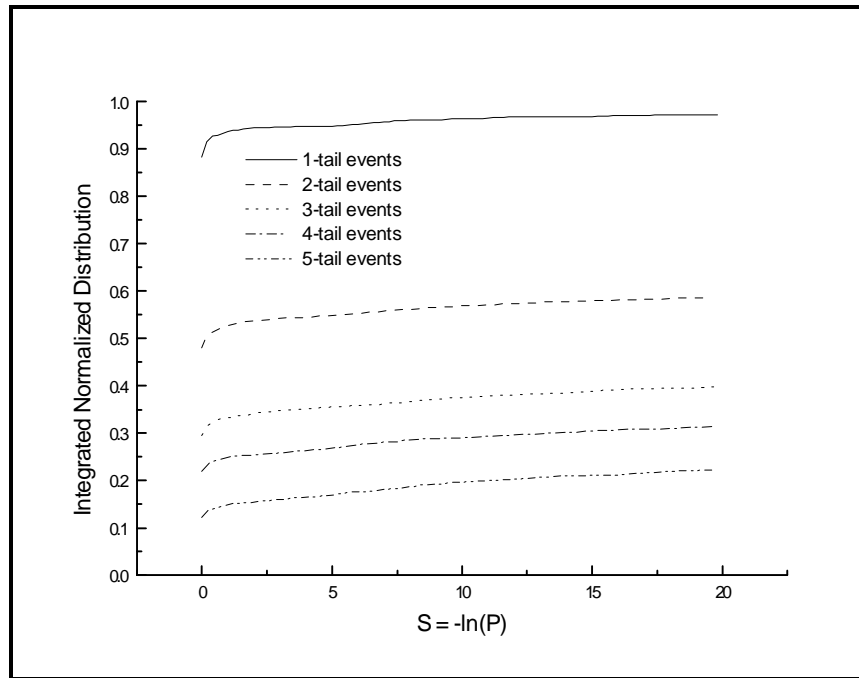


Fig. 3-11: Integrated Probability of Coincidence Distribution for AVHT-GEANT Tails

3.6 Track Quality Distributions

The LSF of Section 3.3 will produce an optimal parameter set, X , for almost any set of wires (assuming there is enough information); however, merely fitting a track to a list of wires does not imply that the track actually represents the trajectory of a real particle. One method for such a determination, referred to as the goodness of fit or the “quality of a track,” examines the reduced χ^2 , χ_v^2 , of the track given by (3.13). For the purposes of this analysis, the smaller the value of χ_v^2 , the higher the probability that the set of wires could represent the trajectory of a real track. To explore this effect, tracks from the AVHT DST (with a *Same-Track Cut*) were compared to the trajectories of actual particles passing through the detector as specified in the GEANT DST. The χ_v^2 distributions for AVHT tracks that approximated GEANT tracks as well as for AVHT tracks that did not are displayed in Fig. 3-12.

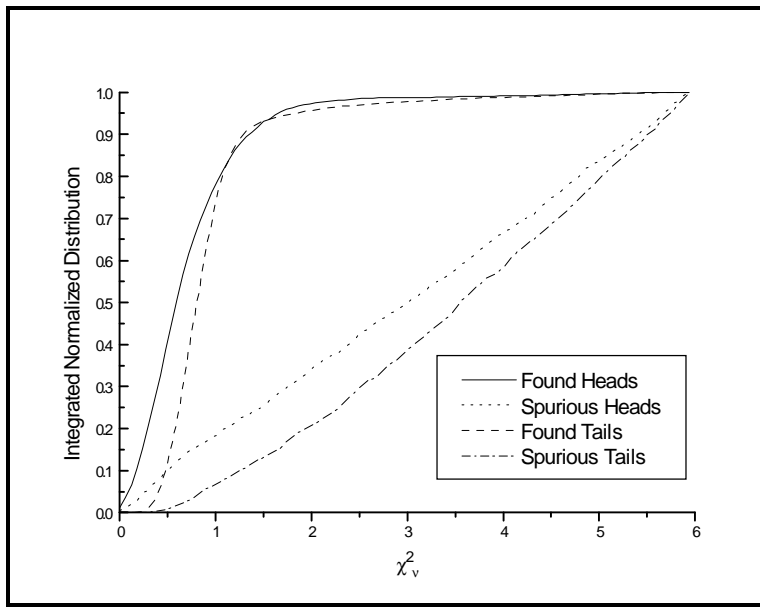


Fig 3-12: χ_v^2 Distribution for AVHT Heads and Tails

The *Same-Track Cut* that was applied to the tracks in the AVHT DST and was also used to determine whether the tracks in the AVHT DST corresponded to tracks in the GEANT DST.

Both distributions for heads and tails clearly show that real tracks found generally have a much smaller χ_v^2 than other (spurious) tracks found. In fact, the χ_v^2 distribution for spurious tracks is uniform. These features are evident from the sharp rise of the found-curves to encompass 98% of real heads and 96.5% of real tails by $\chi_v^2=2.0$.

Making a *Chi-Squared Cut* at the 98% level, while costing the detection of approximately 2% of real tracks, will eliminate over two-thirds of the spurious tracks. This is clearly an advantageous thing to do. Since the distributions for real tracks are fairly flat about the 98% level, the actual cut can be adjusted downward somewhat without significantly affecting the number of tracks found. However, shifting the cut upwards to detect more real tracks is not cost-effective because the slope in the spurious track curve is much larger than that for real tracks. The 98% values for the *Chi-Squared Cut* ($\chi_v^2=2.0$ for heads and $\chi_v^2=2.85$ for tails) will be used through out the rest of this text. Note that, for tails, a cut at the 96.5% level, while costing only 1.5% more real tracks, will eliminate four-fifths of the spurious tails. In practice, the AVHT DST file employs a cut of 3.0 for both heads and tails so that the end-user can employ his own optimal cuts at analysis-time.

3.7 Interfacing the Computer Code

Discussion of the actual implementation of the AVHT algorithm is beyond the scope of this document; however, the actual interfacing of the computer code is not very difficult. In order to examine events from GEANT or real collider runs, one must interface with the set of FORTRAN functions collectively known as *offline*. An example C source file called *track.c* along with its *Makefile* for interfacing with *offline* and creating a DST file are given in Appendix C. The reader is referred to this appendix in connection with the following discussion.

Any interface to *offline* must provide three functions: *user_init_*, *user_anal_* and *user_end_*.¹⁴ The function *user_init_* is called once at the beginning of the program; in it global variables and files can be initialized. Then, for each event, *offline* calls *user_anal_* allowing the user to perform whatever form of data analysis or tracking is necessary on an event by event basis. Finally, after all events have been processed, *user_end_* is evoked allowing one to save data obtained from the analysis of the events, etc. A FORTRAN version of *offline* exists for the analysis of either real or GEANT data files. It should be noted that the data file to be analyzed through *offline* must be in the user's current directory and named¹⁵ *FOR010* for real data files and *fort.12* for GEANT data files. The analysis of GEANT data files can be simplified. Instead of employing the GEANT version of *offline*, one can use the function *trackdef.c:LeadGeantEvent* to directly read the GEANT data file from C, allowing more flexibility and eliminating the need for the three interface functions previously mentioned.

The interface to the AVHT tracking code is equally simple. There are only two functions, *hougher.c:setup_hougher_* and *hougher.c:hougher_* that must be employed,

¹⁴ The trailing underscore is employed for C functions and variables that will be accessed by FORTRAN functions. In FORTRAN, these underscores are unnecessary.

¹⁵ In UNIX, it is actually not necessary for the files to actually be in the current directory. If the files exist elsewhere on the hard drive, a soft link can be created. For example, if the path and file name of the datafile to be analyzed is *path*, a soft link can be created by typing "ln -s *path* *FOR010*". This effectively creates a pointer named *FOR010* in the current directory to the actual file.

although a whole library of other functions is also available for data analysis purposes, see the files *trackdef.h*, *bjhough.h* and *stat.h* in Appendix B. In addition to these functions, three structures, *trackdef.h:Track*, *trackdef.h:Tracks*, and *trackdef.h:Event*, as well as the structure *trackdef.h:EventCommon*, which has the same format as the FORTRAN event common block present in *offline*, should be familiar to the user. Finally, several global tracking variables are available for tweaking the AVHT algorithm. These include: *hougher.c:ChiChiH*, *hougher.c:ChiCutT*, *hougher.c:MetCutH*, *hougher.c:MetCutT*, *hougher.c:MinPlaH* and *hougher.c:MinPlaT*, representing the *Chi-Squared Cuts*, *Same-Track Cuts* and *MinPlanes Cuts*, respectively, to be employed for heads and tails. The default values for these cuts are given in Sections 3-2, 3-5 and 3-6 and also described in the relevant header file, *hougher.h*.

The function *setup_hougher_* must be invoked as the beginning of the program, before any other AVHT functions are accessed, e.g. in *user_init_*. *setup_hougher_* requires one parameter, the name of the detector configuration file to be used. This file describes the orientations of the MWPC planes and must be of the format described in *coords.h*—very similar to Table A-2. Before calling *hougher_* to track an event, the global *Event* structure, *event*, must be filled with the active wires. This can be done with the function *trackdef.c:LoadEventOFF* if the data is already in the FORTRAN event common block. There are other functions in *trackdef.h* that facilitate filling *event* in other situations. When *hougher_* is called, the AVHT algorithm is executed separately before and after the PbC using the parameters given in the previous paragraph. The resulting tracks found are stored in the global *Tracks* structure, *tracks*.

There are a few standard procedures that must be followed in *user_anal_* in order to obtain event information. First, the FORTRAN function *unpacker* must be invoked for load the event data into the FORTRAN event common block *event_*.¹⁶ Then, if the AVHT tracking algorithms are to be used, the function *LoadEventOFF* must be called to copy the event information from the common block into the global AVHT event structure, *event*. Next, to perform the tracking, the function *hougher* must be called. The track information returned in *tracks* can then be analyzed using functions from *stat.c*, user defined functions, or written to a DST file using *trackdef.c:WriteTEvent*.

While the multiplicity of functions and variables may seem formidable, the analysis of Appendix C and the header files in Appendix B, in which the relevant functions and variables are explicitly defined, should clarify the procedure. Once a desired *C* interface to *offline* has been created, one only needs to modify a few lines in the *Makefile* to specify the program name and relevant directories before being able to compile, link and execute the program.

Many functions are provided for analyzing the contents of the DST files that can be created through the above procedures and the program in Appendix C. Many of these are detailed in *stat.h* and *trackdef.h*. An example DST analysis program and its *Makefile* are supplied in Appendix D. For FORTRAN users, the details of the DST file format are given in Section 3.8 so independent analysis procedures can be developed.

¹⁶ See the file *trackdef.h* for a declaration of this common block, *EventCommon*.

defined in Chapter 1. The seventh floating point number on this line is the χ_v^2 of the track. All these numbers are right justified in fields nine spaces wide and have four decimal places of precision. The last two numbers on this line are integers representing the number of wires contributing to the track and the track type¹⁷, respectively. These are right justified in fields nine and two spaces wide; all fields are separated by a single space.

The second line describing a track contains the list of the active wires that contributed to the track. From this list, all information in the previous line could be reconstructed via the LSF, if necessary. Each of the wire numbers are integers right justified in a field five spaces wide and, of course, each field is separated by one space. If p is the plane and w is the wire number on that plane of an active wire, then the wire number, as displayed in the DST, will be $p \cdot 128 + w + 1$, where $p \in [0, 23]$ and $w \in [0, 127]$. The DST file is terminated by the standard end of file (EOF) marker.

¹⁷ Possible track types are 0 and 1, representing heads and tails, respectively.