I. Introduction

The fitting routines are designed to be efficient at fitting tracks to hits that are associated with the particle being tracked. The efficiency can be further increased and errors due to local minima can be decreased by using a set of "FirstGuess" track parameters as the starting point for the fitting routines. Pattern Recognition is designed to address these issues by separating hits according to particles and providing the beginning approximation to the track parameters.

II. Outline

A. Separate hits according to particle
   1. Temporally
   2. Spatially
   3. Group hits according to projections on the u−v plane
      a. Circles (default)
      b. Spots (soft particles)

B. Fit tracks to hits
   1. Calculate circle parameters
      a. Center (u_c, v_c)
      b. Radius
   2. Calculate remaining helix parameters
      a. Azimuthal angle
      b. Phase

C. Auxiliary processes
   1. Histogramming
   2. Logging
   3. Selecting particular classes of events

D. Data files
   1. Passed variables and structures
   2. Ntuple variables

III. Windows

The windows represent the temporal separation of the particles. Each event is organized into time windows. Particles may be present at any particular time during the event. Thus a window may represent a particle or the time gap between particles. Furthermore, window limits are flexible to handle varying length gaps or the possibility of particles overlapping in time.

PC hits are used to define the time windows since they have better time resolution than the DCs. All PC hits that differ in time by less than the namelist variable, FirstGuess WinPCThreshold, are assumed to come from a single particle or coincident particles. The first of these PC hits defines the window t_0. A time window is created about this t_0 using the namelist variables, FirstGuess WinDCStart and FirstGuess WinDCEnd, with the range;

$[t_0 + \text{WinDCStart}, t_0 + \text{WinDCEnd}]$.

WinDCStart is typically a negative number. At the time of this writing,
WinPCThreshold = 100 ns (determined by long PC drift times)
WinDCStart = −50 ns (determined by distribution of PC drift times)
WinDCEnd = 1000 ns (determined by long DC drift times)

The nth window, so defined, is designated window(2n). The windows representing time gaps between particles are designated with the appropriate odd index. In addition, there are windows representing the gap between the beginning of the window and the first particle and between the last particle and the end of the event. (Note that windows representing gaps may have 0 ns width.) Thus, an event with n particles separated in time would have 2n + 1 time windows.

A typical clean event with a muon at t_m and a decay positron at t_p would have these windows.

		t_m		t_p		
<u>Windows</u>						
Particle		[2]		[4]		
Gap	[1]		[3]		[5]	

Particle windows, as defined above, could overlap in time. In such cases, only the non−overlapping portion is used to create each of the particle windows. The remaining overlapping portion is defined as an overlap window. Such an event has one fewer gap window and an additional overlap window, preserving the 2n + 1 count for n particles separated in time. (Note that overlapping windows do not have PC hits. Since PC hits define the windows, overlapping PC hits are indicative of particles coincident in time. In other words, PC time resolution defines coincidence.)

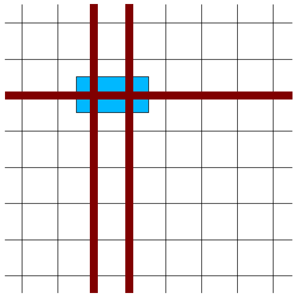
III. Clusters

The clusters represent the first stage of spatial separation of hits. Hits are organized into clusters of adjacent hits within a single time window.

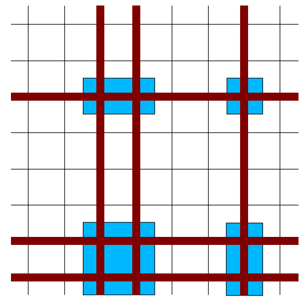
The clustering is accomplished in the context of pairs of planes. Each pair contains a u plane and an adjacent v plane. Pairs can be distiguished as PC or DC, as Dense, Sparse (DCs), or Target (PCs).

A cluster is created from adjacent hit wires for each plane in the pair. A combined UV cluster is created from all possible combinations of one cluster from the u plane and one cluster from the v plane. The minimum and maximum wire is determined for both u and v for each cluster. The position of the cluster center is calculated for each cluster. Coincident particles result in spurious clusters. Occasionally, these clusters have a position that is outside the active region of the detector. Such clusters are removed.

A typical set of clusters for a single particle window and a two particle window:
Red lines represent hit wires. Boxes represent resulting clusters.



Single particle cluster



Two particle window illustrates ambiguity inherant in clustering.

IV. Fitting Tracks

The pattern recognition parameterizes a helix in terms of a projection onto a circle, an azimuthal angle, and a phase. The circle parameters are the center coordinates, u_c and v_c, and the radius, R. The remaining parameters are φ_o, the azimuthal angle at the target, and ω, a phase.

The circle parameters are re–expressed to facilitate a simple χ^2 fit. The equation of a circle is

$$(u - u_c)^2 + (v - v_c)^2 = R^2$$

This is equivalent to

$$S_q + 2Au + 2Bv + C = 0$$

where

$$A = -u_c \qquad B = -v_c \qquad C = A^2 + B^2 - R^2 \qquad S_q = u^2 + v^2$$

Thus, we can solve for the circle parameters by a χ^2 minimization.

$$\chi^2 = \Sigma \frac{1}{\sigma^2} (S_q + 2Au + 2Bv + C)^2$$

First guess beginning at the target, fits circles to all possible combinations of 1 (or 0) cluster per pair until the parameter MaxTmpTracks is reached. Continuing through the pairs, additional clusters are included in the fit if

$$(u - u_c)^2 + (v - v_c)^2 - R^2 < \text{MaxCirErr}$$

The equations for a helix are

$$u = R\cos(\varphi_o + \omega z) + u_c$$
$$v = R\sin(\varphi_o + \omega z) + v_c$$

Inverting these

$$\cos^{-1}\left(\frac{u - u_c}{R}\right) = \varphi = \varphi_o + \omega z$$

$$\sin^{-1}\left(\frac{v - v_c}{R}\right) = \varphi = \varphi_o + \omega z$$

This again leads to a χ^2 fit.

$$\chi^2 = \Sigma \; \frac{1}{\sigma^2} \; (\varphi - \varphi_o - \omega z)$$

First guess does not remove any clusters based on the φ_o, ω fit. First guess does not account for energy loss. The energy loss produces a loss of resolution in the circle fit that is easily accomodated. However, energy loss produces a gradual shift in φ, that is not currently taken into account at the level of first guess.

V. Passing Tracks to Fitting Routines

First guess passes tracks to the fitting routine through the structure, FGResult. FGResult contains a list of all hits that could possibly be in the track, the starting parameters, and the χ^2s from the fits. The best track from all candidates within a time window is passed. The best track is determined by the function, SelectBestTrack. At the time of this writing, the best track is selected on the basis of the χ^2 from the φ_o, ω fit. (Figure of merit chosen from studies that are described later in this document.)

VI. Histogramming

Histograms for pattern recognition are produced both directly and through an ntuple. Histograms that are written directly are given id's from 18001 to 19000. They are defined in hists_mod and filled in the various pattern recognition modules. The resulting histograms are in the standard mofia hbook file that is written to the directory given by the environment variable, MHIST.

The ntuple created by pattern recognition is also written to the same directory. It is a disk resident column−wise ntuple. The ntuple can be quite large. Typically, an ntuple for a single run will be written to several files. The filenames have the format of

RunNumber_FileNumber.nt

The writing of the ntuple is controlled by the namelist variable,

FirstGuess EnableFirstGuessNtuple.

There are two sets of variables that may be written to the ntuple, depending on whether data from the monte carlo banks is included. The choice is controlled by a combination of an internal flag denoting monte carlo input, and the namelist variable, Global UnpackMC.

PAW macros to load the ntuple files, set up cuts and produce histograms are available. It is reported that PHYSICA cannot manipulate column–wise ntuples.

VII. Logging

Log files can be written for some pattern recognition processes and structures. The logging is controlled by namelist variables from the namelist, Log. Current log files and the associated namelist variables:

filename	namelist variable	
cl.log	Log	ClLogging
windc.log	Log	WinLogging
fgresult.log	Log	FGResultLogging

These log files contain data from the cluster, window, and fgresult structures.

VIII. Classifying Events

Pattern recognition contains a set of auxiliary subroutines that can be used to classify and select events. These subroutines are found in classify_mod. Up till now, this code has been used to select events for skimming and to produce histograms of event types. It is anticipated that some routines will be incorporated into pattern recognition as standard features.

The classification routines are controlled by namelist variables:

Classify	SelectCleanEvent	
Classify	SelectBeamPositron	
Classify	SelectDelta	(Currently being developed)
Classify	UserSelect	(User written routine)

Skimming of selected events can be accomplished by setting the parameter, OldSkim = .TRUE., within classify_mod, until the new skimming code is in place. With the completion of the new skimming code, skimming of selected events will be controlled by the following namelist variables:

(In conjunction with SelectCleanEvent)
- Skim SkimClean
- Skim SkimDirty

(In conjunction with SelectBeamPositron)
- Skim SkimBeamPositron
- Skim SkimNoBeamPositron

(In conjunction with SelectDelta)
- Skim SkimDelta
- Skim SkimNoDelta

(In conjunction with UserSelect)
- Skim SkimUserSelect
- Skim SkimUserUnSelect

The user may select and skim based on their own routine inserted according to documentation in the code and using the appropriate namelist variables.

IX. Efficiency of Track Reconstruction of Monte Carlo Data

A study has been conducted to determine the efficiency of pattern recognition on monte carlo data. Efficiency was defined as the ratio of successful reconstructions to total events within a specified acceptance. For the purpose of this study, a successful reconstruction was defined as one where

$$|R_{FG} - R_{MC}| < 0.5 \text{ cm}$$

and

$$\sqrt{(u_c^{FG} - u_c^{MC})^2 + (v_c^{FG} - v_c^{MC})^2} < 0.5 \text{ cm}$$

(FG refers to the reconstructed quantity, MC to the monte carlo bank data.)

Note that the criteria included only the circle parameters and not the full helix parameters.

A set of monte carlo runs were analyzed, with various physics processes. Events were classified according to the variable, FailCode. FailCode = 0 represents a successful reconstruction as defined above. FailCode = 1 represents an event where the pattern recognition failed to find candidate tracks. FailCode = 2 represents an event where the pattern recognition found candidate tracks, but none of them met the criteria of a successful reconstruction.

	% with FailCode			
Monte Carlo	0	1	2	Total
Positron source at target No additional physics	99.9	0.1	0	89907
Positron source at target Multiple scattering	99.8	0.1	0	89757
Positron source at target Multiple scattering and energy loss	99.5	0.3	0.2	89869
¡Positron source at target All available physics in version 2.0 on	99.3	0.4	0.3	89953
Muon beam source All available physics in version 2.0 on	97.9	1.2	0.8	19168

The energy acceptance for each run was

$0.3 < x < 0.9$

The θ acceptance for each run was

$0.55 < |\cos\theta| < 0.955$ $(17.3 < \theta < 56.6)$.

The decay time acceptance for the run with the μ source was

$1000 \text{ ns} < \Delta t < 9000 \text{ ns}$

Figure of merit studies were conducted using the run with the μ source. Candidate tracks were compared according to three criteria:

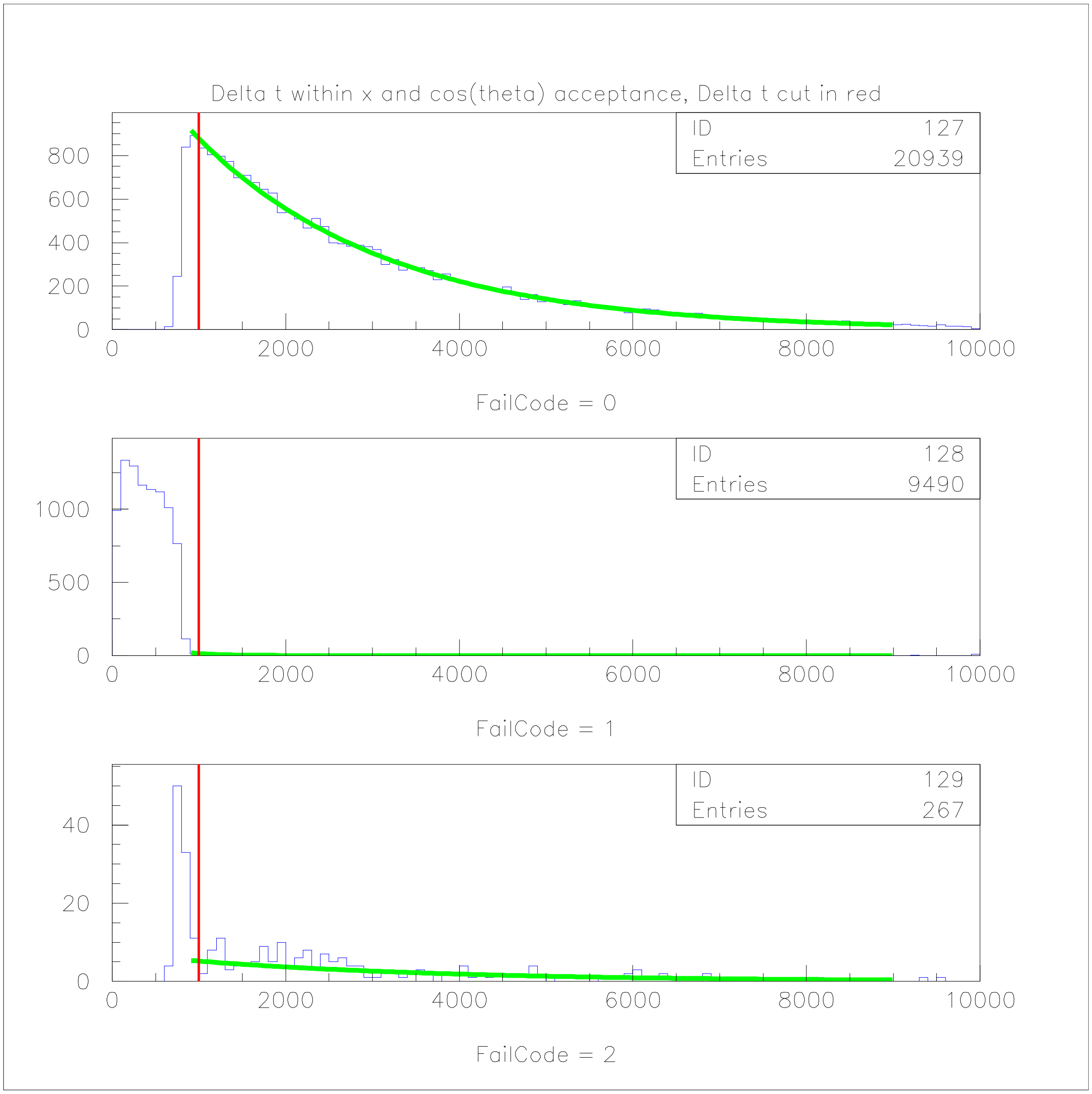
The best fit to the monte carlo track, designated BestTrack

The best χ^2 fit to the circle parameters, designated Best Cir χ^2
The best χ^2 fit to φ_o, ω, designated Best Phi χ^2

For this study, the definition of a successful track required the circle parameters to differ by less than 1 cm, rather than the tighter 0.5 cm requirement used in previous studies.

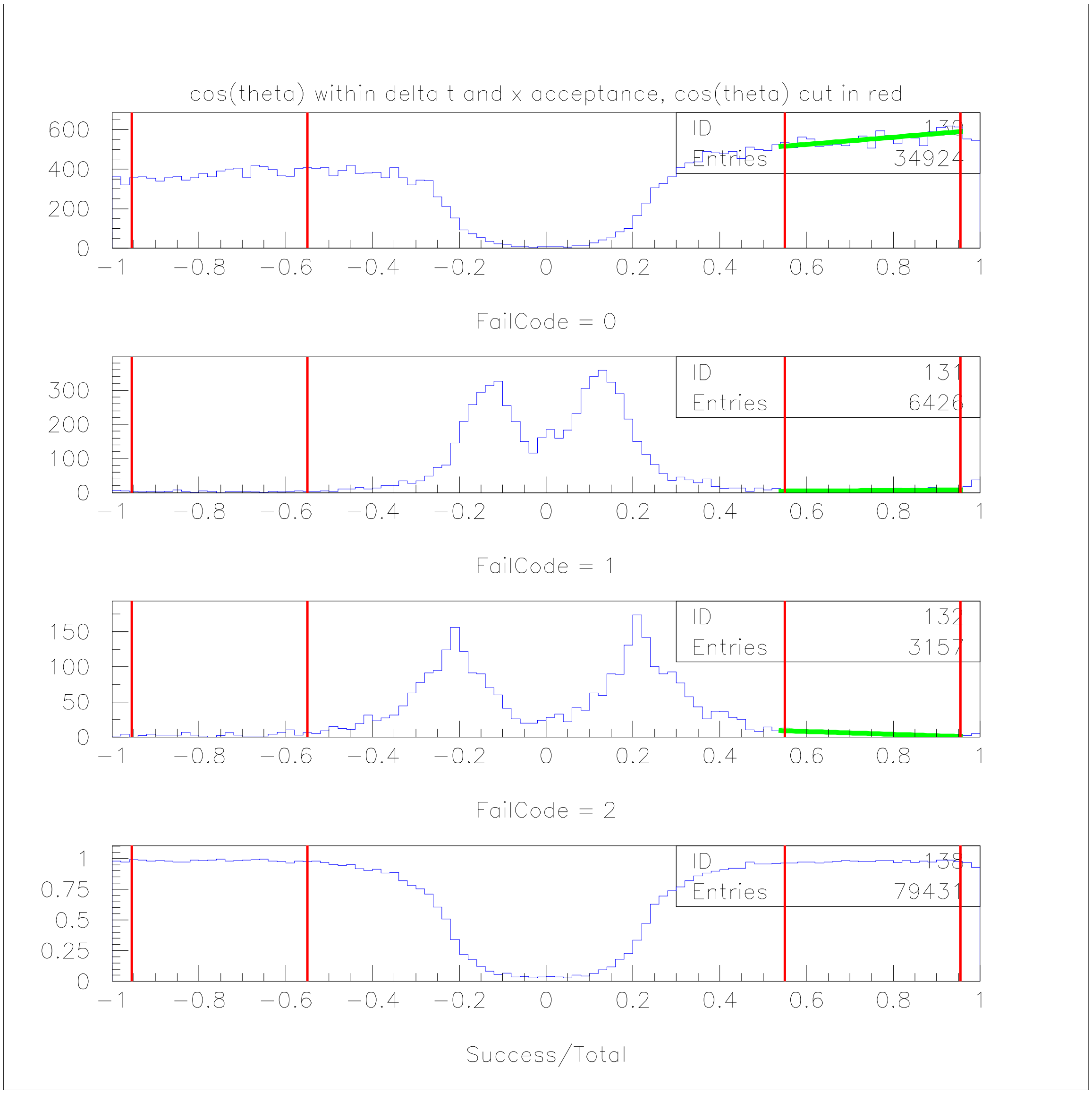
FailCode	***Portion of Total, 19084 (%)***	***BestTrack = Best Cir X^2 (%)***	***Mean Circle X^2***	***BestTrack = Best Phi X^2 (%)***	***Mean Phi X^2***	***BestTrack = Best Cir X^2 = Best Phi X^2 (%)***
0	***18828 (99)***	***18481 (98)***	***1.66***	***18598 (99)***	***0.48***	***18407 (98)***
1	***144 (0.8)***	—	***7.73***	—	***3.73***	—
2	***112 (0.6)***	***78 (70)***	***10.82***	***79 (71)***	***3.44***	***72 (64)***

Acceptance studies were also conducted using the μ source run. The variable associated with the acceptance being studied was unconstrained and plotted for each value of the variable, FailCode. The other acceptances were kept at the nominal values given previously. In the case of cos θ and energy (x) acceptance, an additional plot was made of successful events divided by total events.



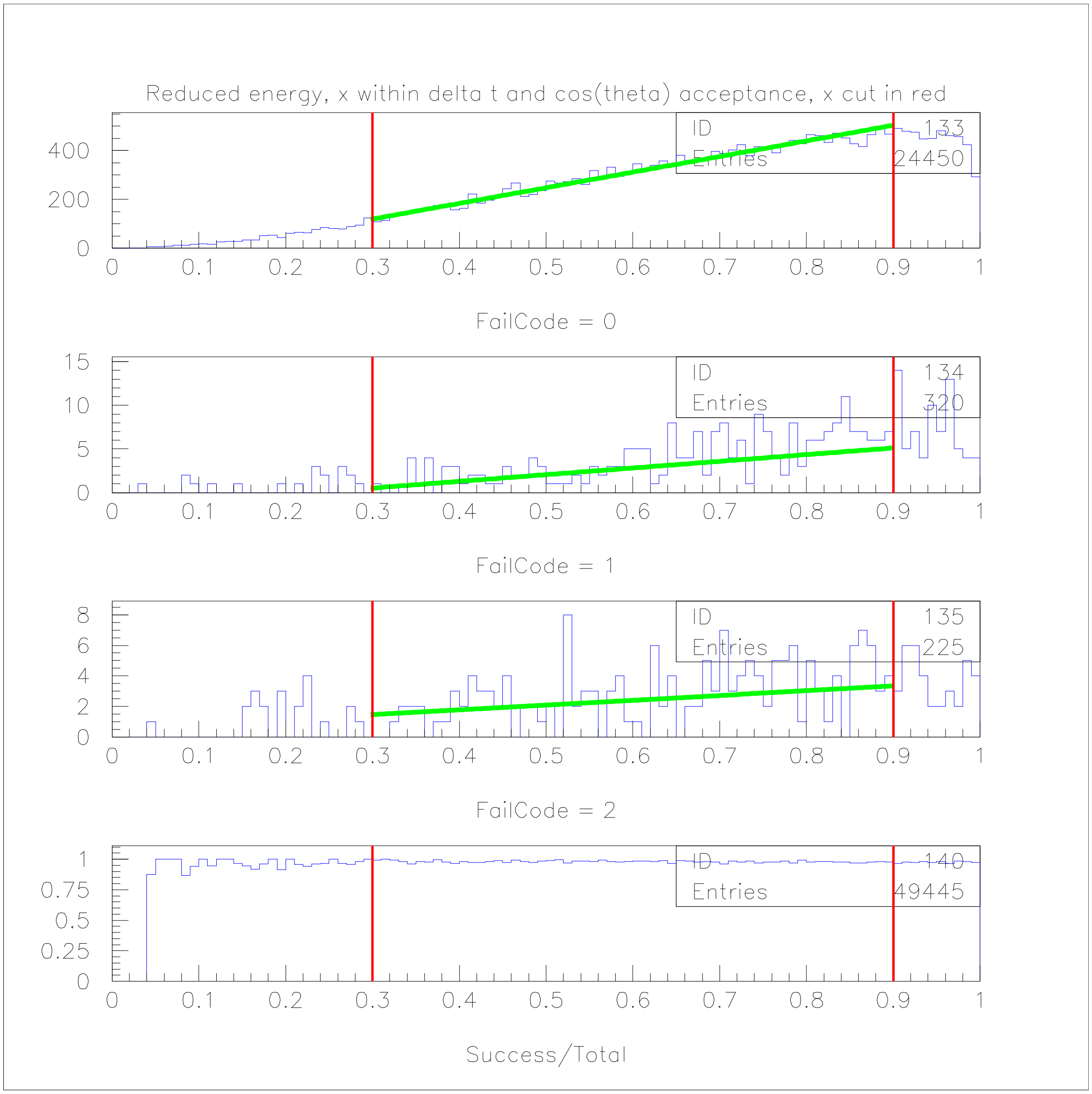
The fit to Δt was an exponential fit of the form $e^{a - \lambda(\Delta t)}$.

FailCode	***a***	**λ *(s^{-1})***	$\frac{\chi^2}{dof}$
0	7.23 ± 0.02	4.57×10^{5} $\pm 0.05 \times 10^{5}$	$\frac{69.89}{79} = 0.88$
1	6. $\pm 1.$	0. $\pm 1.$	$\frac{182.9}{57} = 3.21$
2	2.0 ± 0.3	3.40×10^{5} $\pm 0.10 \times 10^{5}$	$\frac{46.03}{55} = 0.84$



The fit to cos θ was a linear fit of the form, m(cos θ) + b

FailCode	***m***	***b***	$\frac{\chi^2}{dof}$
0	180 ± 40	420 ± 30	$\frac{26.72}{19} = 1.41$
1	8. $\pm 5.$	0. $\pm 4.$	$\frac{46.64}{19} = 2.45$
2	−20. $\pm 4.$	20. $\pm 3.$	$\frac{11.37}{19} = 0.60$



The fit to x was a linear fit of the form, mx + b

FailCode	***m***	***b***	$\frac{\chi^2}{dof}$
0	640 ± 10	$-72. \pm 7.$	$\frac{80.74}{58} = 1.39$
1	$8. \pm 1.$	-1.8 ± 0.8	$\frac{60.98}{55} = 1.11$
2	$3. \pm 1.$	0.5 ± 0.8	$\frac{42.02}{49} = 0.86$

Further acceptance studies were completed by varying the acceptance and noting the percentage of successes and failures. For all but the last of these, the definition of a successful reconstruction was the original definition requiring that the circle paramerters of the reconstruction differ from the monte carlo parameters by less than 0.5 cm. For the last test, the difference was required to be less than 1.0 cm.

Acceptance tests FailCode

Decay Time (ns)	***x***	***cos*** θ	***% = 0***	***% = 1***	***% = 2***
no cut	$0.4 < x$	$0.3 < \|\cos\theta\|$	82	12	5
$1000 < \Delta t < 9000$	no cut	no cut	76	1	23
$1000 < \Delta t < 9000$	$0.4 < x$	$0.3 < \|\cos\theta\|$	95	1	5
$1000 < \Delta t < 9000$	$0.4 < x < 0.9$	$0.3 < \|\cos\theta\|$	95	1	4
$1000 < \Delta t < 9000$	$0.4 < x < 0.9$	$0.4 < \|\cos\theta\|$	97	1	3
$1000 < \Delta t < 9000$	$0.4 < x < 0.9$	$0.55 < \|\cos\theta\|$	98	1	2
$1000 < \Delta t < 9000$	$0.4 < x < 0.9$	$0.55 < \|\cos\theta\| < 0.955$	98	0	2
$1000 < \Delta t < 9000$	$0.5 < x < 0.9$	$0.55 < \|\cos\theta\| < 0.955$	98	0	2
$1000 < \Delta t < 9000$	$0.4 < x < 0.9$	$0.6 < \|\cos\theta\| < 0.955$	98	1	2
$1000 < \Delta t < 9000$	$0.3 < x < 0.9$	$0.55 < \|\cos\theta\| < 0.955$	99	0.5	0.6

X. Classifying the November Data

Studies of the characteristics of the November data are being conducted, primarily using run 3961. The purpose is to determine the rates of various types of events. Two types of events have been successfully identified and quantified for the entire run. In addition, a careful examination by eye of the first hundred events was conducted using the event display.

Muons

	Fast Decay	Later Decay	No Decay	Total
Total Muon	18	63	2	83
No Secondary	14	47	2	63
Secondary	4	16	0	20
In Time with Decay	4	6	0	10
Secondary separated in Time	0	10	0	10
Secondary is Trigger	0	6	0	6
Secondary is not Trigger	0	4	0	4

Beam Positrons

Total Beam Positrons	11
1 Particle	9
2 Particles separated in Time	2

No Particles

6

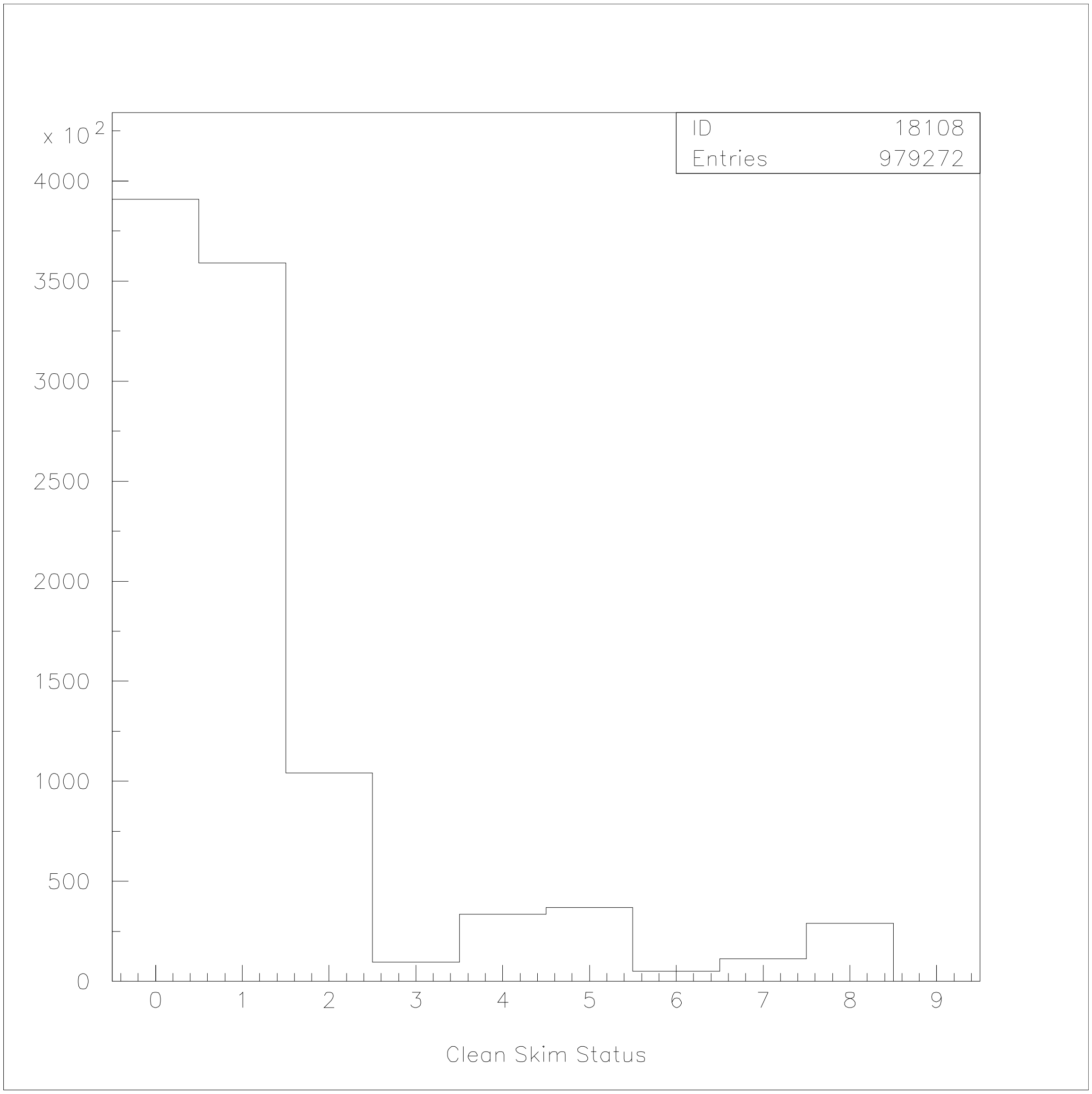
The visual examination of the data characterized events as either μ^+, beam e^+, or no particle. For events that included both a μ^+ and a e^+, the event was counted as a μ^+ event. Within μ^+ events, distinctions were not made between beam e^+s and other secondary particles. μ^+ events were further divided according to decay time, time coincidence of secondaries, and trigger source.

Identification and skimming for the entire run based on event type was accomplished using the routines in classify_mod. Events have been classified as either clean or dirty and as either containing a beam positron or not.

The clean event selection was based on surviving a set of cuts that were applied in the order listed below. If an event failed a cut, it was not checked against the remaining cuts. The skimming code was executed on run 3961. The run contains 979272 events. The number of events removed by each cut is given and illustrated in the following histogram.

Clean event cuts:			Number cut:
1	Number of windows, $n \neq 5$		390820
2	Decay time, $\Delta t < 1$ μs		359040
3	Window 4 $t_0 > 9$ μs		104150
[Cuts 4 and 5 are cuts on window 2.]			
4	Insufficient hits in upstream half of detector		33487
		Number of upstream PC planes hit < 4	
	or	Number of upstream DC planes hit < 14	
	or	Number of upstream target PC planes hit = 0	
5	Particle in downstream half of detector		36905
		Number of downstream outer PC planes hit > 0	
	or	Number of downstream DC planes hit > 0	
[Cuts 6 through 8 are cuts on window 4.]			
6	Insufficient hits in detector		5152
		(Number of upstream PC planes hit < 4	
	or	Number of upstream DC planes hit < 14)	
	and	(Number of downstream PC planes hit < 4	
	or	Number of downstream DC planes hit < 14)	
7 and 8		Particle in both halves of detector	40200

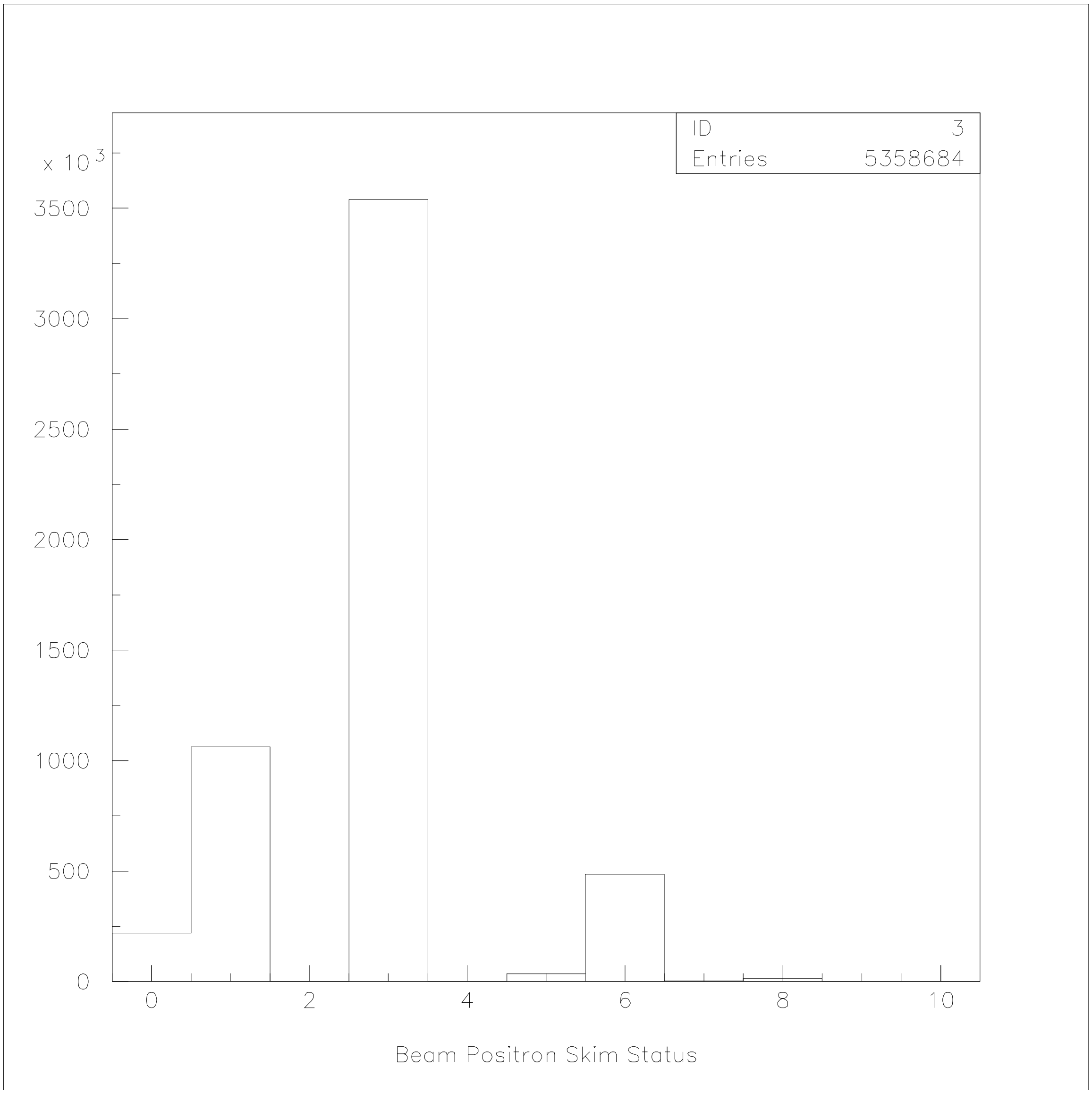
The number of events that passed the skimming code was 390820 (39.9%). These events have a skim status of 0 in the histogram.



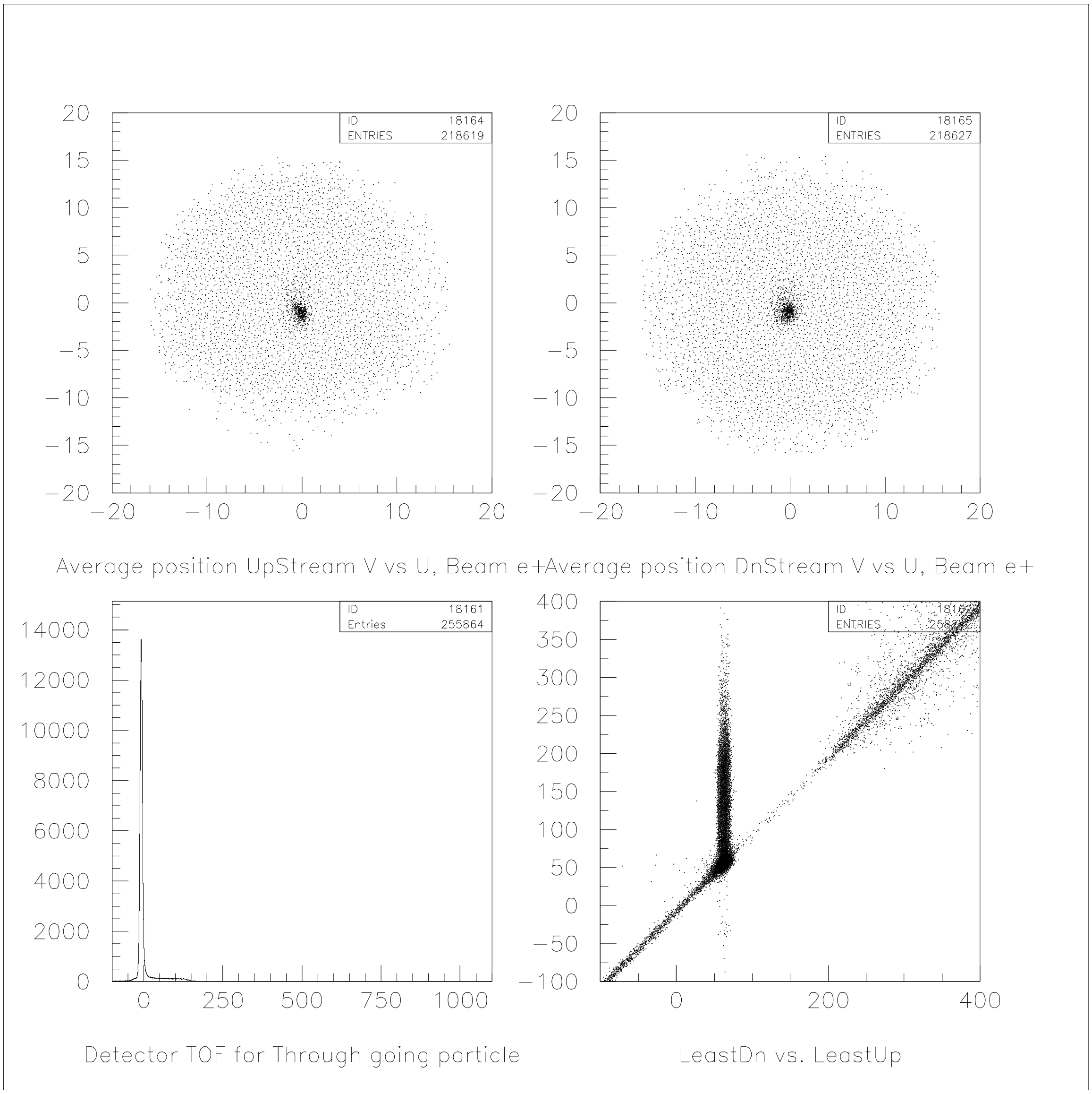
The positron skim was based on a set of cuts that were applied in the order listed below. If an event failed a cut, it was not checked against the remaining cuts. The skimming code was executed on run 3961. The run contains 979272 events, with a total of 5358684 windows. The number of windows failing each cut is given and illustrated in the following histogram. 197958 events, 20.2%, passed the cuts for at least one window and were skimmed.

Positron event cuts:		Number cut:
1	Number of upstream PC planes hit < 4	1061900
2	Number of upstream outer PC planes hit < 2	0
3	Number of downstream PC planes hit < 4	3539500
4	Number of downstream outer PC planes hit < 2	0
5	Time of flight > DetectorTOFcut = 5 ns (Earliest upstream PC time – Earliest downstream PC time)	34947

6	Number of upstream DC planes hit < 15	486700
7	Number of upstream dense DC planes hit < 5	3335
8	Number of downstream DC planes hit < 14	13055
9	Number of downstream dense DC planes hit < 5	595
0	Window passed all cuts. Labeled as beam e^+	218640



The following plots were created from the positron skim file. Upper left shows the average position of hits in the upstream PCs, v vs. u. Upper right shows the same for downstream PCs. Lower left shows time of flight through the detector for through−going particles as calculated by taking the difference of the earliest downstream PC time and the earliest upstream PC time (within a time window). Lower right shows the earliest downstream PC time vs. the earliest upstream PC time (within a time window). It should be noted that the vertical line in the last plot and the tail of the time of flight plot correspond to times from multiple particles.



XI. Analyzing the November Data

The firstguess code has been run on the clean event skim file. The efficiency of reconstruction is 83% without placing cuts on acceptance.

The clean event skimming code is being run on a monte carlo produced file in order to have a comparison for the data efficiency. This document will be updated with the results of that study in the near future.