

Note on Pattern Recognition in the ATLAS Muon Chambers

Werner Krischer
ECP Division

Contents

- I. Introduction
- II. Local pattern recognition
- III. Global track detection
- IV. Conclusion
- V. Acknowledgement
- VI. References

I. Introduction

In [1] a pattern recognition algorithm is described as candidate for use in level-2 triggering. It uses the hits from the RPC's to define roads into the MDT's. In defining tracks in the (8-layer) 'station', it uses only the tube positions, the resulting residuals on points are of the order of several centimeters in space. In [2] a method is described which translated the points for the 3 stations into tracks, using a translation by a look-up table (LUT).

The aim of this note is to indicate a method to get these points and, in addition, the direction at these points with higher precision by using the drift time. These line elements can then be used to define tracks. The added precision will translate into a sharper cut on p_T in the level-2 trigger, and makes the method a candidate even for off-line reconstruction.

In chapter II, we will first describe the basic algorithm used in one station. The LUTs from [2] were not available for our tests, so we developed a simple element "matching" method, which we will describe in chapter III. We compare its performance with the full reconstruction code of the ATLAS reconstruction program version May '95 ("ARMay95").

II. Local pattern recognition.

In this chapter we describe a program developed without detailed ATLAS tracks (see III). It allows to generate (interactively or automatically) tracks at any angle. Fig. 1 shows a display of the output of this program. At the right one sees the tubes of one station and a track generated by two random points. Many problems in reconstruction arise from "noise" hits, so we added random hits to adjacent tubes at a level of 75% in each layer, independent whether hit or not (random drift time, randomly left or right). The tracks have no measurement errors.

From multiple attempts we retained as optimal method of track finding the following :

The distances d of all tube centres from a series of straight lines through a reference point are precisely defined by two parameters, a position (given as index into a table) and an angle: $d = d(\text{index}, \text{angle})$. All d -s are pre-computed for a matrix of values (index, angle), and stored in a LUT. The method consists then of histogramming the hits in two dimensions (angle and position). Every hit gives two entries ($d-r$ and $d+r$); we then find the maximum in this histogram. A binwidth of 0.125 cm was used for $d \pm r$. Fig. 1 shows to the left the histogram and at the bottom a small region magnified around the found maximum. This algorithm gave better results than the standard variable slope histogram of $(x, y+r)$ and $(x, y-r)$. That method ("Hough"-transform) we used for the TRT, where only tracks at very shallow angles are needed. Here, at least for high rapidity, the angles are much larger, and a procedure taking angles into account is essential.

Fig. 2 shows in the middle row the distribution of the difference between found and given position and direction for this method, plotted for 1000 randomly chosen tracks. For many purposes (e.g. level-2 trigger) this precision may be sufficient. If a higher precision is needed, a straight line can be fitted. The corresponding distributions can be seen on the bottom of Fig. 2 to be compatible with rounding errors.

A comment to the least squares solution of overdetermined systems of linear equations : Instead of the numerically unstable method of "normal" equations we used orthogonal triangular (QR) - factorization or singular value decomposition (SVD), developed all algorithms with MATLAB [3] and implemented them in FORTRAN with the LAPACK - library [5] .

To resolve the $\pm r$ ambiguity in each layer, one could, of course, take all hit circles in the maximum bin of the histogram, compute all straight lines with both signs of r and choose the combination with the minimum residual. As this would be rather time consuming ($O(2^{**}n)$), we looked for other methods, from which we will mention two here.

The first one does not make any assumptions about the sign of r and consists of first fitting a line perpendicular [4] to the circle centres :

$$a*x + b*y + c = 0 \text{ with the constraint } a^{**2} + b^{**2} = 1$$

and then fitting

$$a*x + b*y + c = \pm r, \text{ where the sign comes from the residual of the first fit.}$$

It turned out, that this method needed in 18.5 % of all cases at least one additional iteration (sign change) until one got an acceptable fit.

The second method, which we actually used in III., hardly ever needs any iteration, it requires that the signs of the radii are determined first (a compute-intensive task): The four possible tangents of the top and bottom circles are computed and one chooses the tangent, from which the inner circles have the smallest average perpendicular distances of their centres $\pm r$, a procedure that defines all the signs of the inner and outer radii. Then $a*x + b*y + c = \pm r$ was fitted through all $(x,y,\pm r)$.

Without information about the RPC's, we had to use relatively big histograms (large position/angle coverage). Using the information from RPC's the histogram size shrinks [Fig. 3], and one can think of an implementation in a LUT (or a small number of LUTs). Fig. 4 shows a possible data flow: To each pair (index,radius) would correspond 0, 1 or 2 entries in such a LUT. One bit of the LUT corresponds to the decision if this hit belongs to a specific pattern. One would need then as many counters as there are patterns. This would have to be followed by a process that finds the maximum.

Timing of the histogramming: if one chose a data driven signal processing architecture, the processing time would be the number of hits (N) multiplied by the cycle time (e.g. $N * 20$ nanoseconds in a 50 MHz pipelined system).

III. Global track detection

We got from the "big" chambers 3885 tracks (20 Gev) in $0 < \eta < 0.8$ and $80 < \phi < 100$, from the "small" chambers 746 tracks (20 Gev) in $0 < \eta < 0.1$ and $65 < \phi < 70$ and about 100 tracks from the overlap regions.

Because of the very small overlap of the big and small chambers we choose first if we do the pattern recognition in the big or the small chambers dependent on the ratio of the layers hit.

Without any information about the RPC's we first used an angular histogram of the circle centres to reduce the size of the histogram and of the angle range.

Fig. 5 shows a typical example of a track in the small chamber and Fig. 6 a magnified region around the inner station. Solid circles correspond to muons, hollow ones to non-muons and dotted ones are a "mixture" of a muon and some non-muons (with a modified radius).

We first applied the algorithm of II. (with a histogram binwidth of 0.25 cm) and produced 3 line elements. Because we don't have any knowledge about the magnetic field (or because of lack of the LUT from [2]) we tried to find the tracks based on these elements. We fitted a smooth curve (third order polynomial) through the 3 points and directions and the track was accepted if the residual was reasonable. This was the case in 99 % of all tracks.

Figs. 7 and 8 illustrate one of the rare failing cases (from the big chamber) with the global fit residual of 0.204 cm just above our threshold (0.2 cm). Recovery is possible by dropping the station with the highest residual, recomputing the polynomial through the 2 remaining stations, repeating the local analysis in a smaller predicted area and the third order fit. In Fig. 8 a hollow diamond indicates the element we had to drop, and a solid one the refitted element. The accepted global fit residual became 0.028 cm.

Fig. 9 shows the analysis of the pattern recognition program for the big and small chambers. The residuals of the local line fits are around 0.01 cm and of the global cubic fits around 0.05 cm and 0.14 cm for the big and small chambers respectively. On average 7.7 points have been used for the local line fit. In 99% of all events the correct track has been found and 96% of all hits used have the muon particle identification code.

The comparison with ARMay95 shows some disagreement between the 2 methods. About 1 % of the tracks in the big chamber and 26 % (!) of the tracks in the small chambers could not be tracked by ARMay95 from outside into the middle or inner station.

Timing of full algorithm: if with the hypothetical data-driven architecture, one uses the natural parallelism of the stations, the time given at the end of chapter II is the total processing time including the precise momentum measurement according to [2].

IV. Conclusion

Using the drifttime to improve the precision for second level triggering does not seem to pose a major implementation problem. Even a real time (100kHz) architecture could be designed, if the output of the histogram method is considered sufficiently precise. The comparison with the ATLAS reconstruction program seems to indicate that it is difficult to follow tracks through the magnetic field (particularly in the "small" chambers) and that methods that check how well track elements fit together like the method described here (or the LUT - approach from [2]) would not have any particular problem.

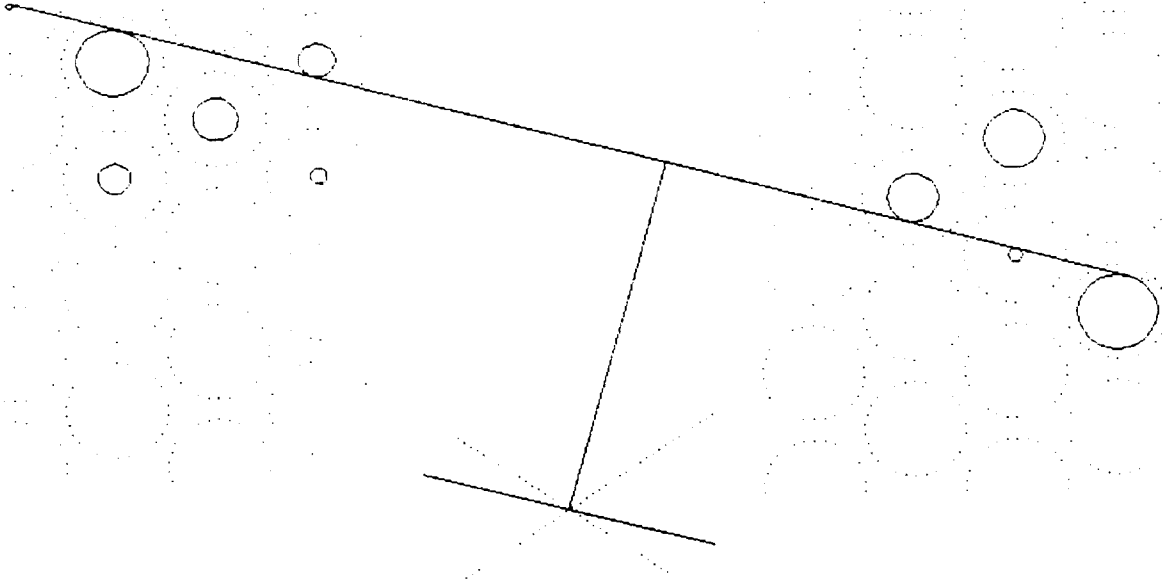
V. Acknowledgement

I would like to thank Traudl Hansl-Kozanecka and Rudy Bock for many useful discussions and in particular Erik Joost Visser for having provided me with the simulated events with the particle identification code and the output of the ATLAS reconstruction program (version May 1995).

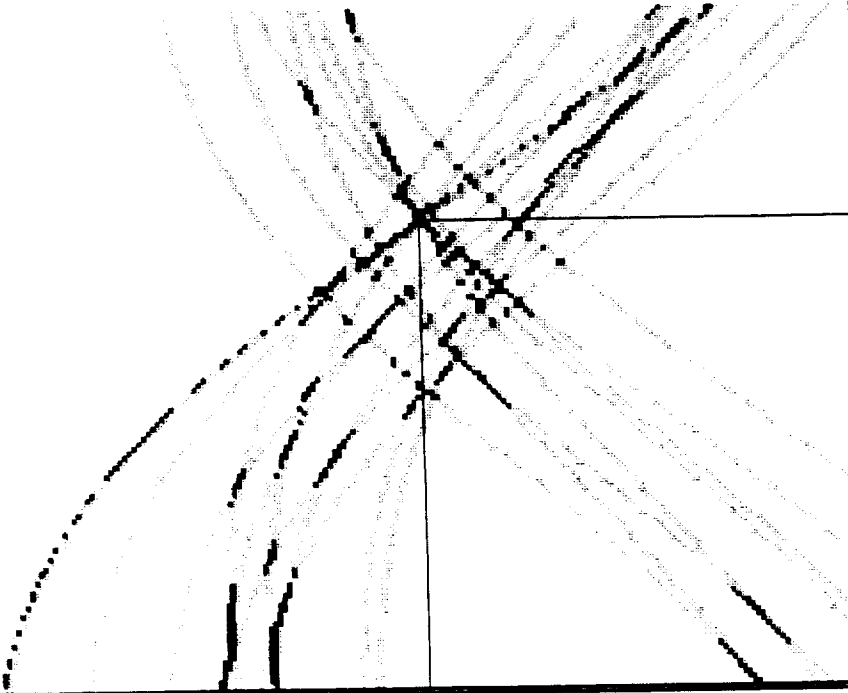
VI. References

1. O.Palamara and S. Petrera, Pattern recognition at the second level muon trigger in the ATLAS barrel region, ATLAS DAQ-NO-017
2. G.Ambrosini and L. Nisati, L2 Muon Trigger, presentation at an ATLAS - trigger meeting
3. MATLAB Reference Guide, 1992, The MathWorks,Inc., Natick, Mass.01760
4. W.Gander and U.von Matt, Some Least Squares Problems, in : W. Gander,Jiri Hrebicek : Solving Problems in Scientific Computing Using MAPLE and MATLAB, Springer Verlag, 1993
5. E.Anderson et al.,LAPACK Users' Guide, Society for Industrial and Applied Mathematics, Philadelphia 1992

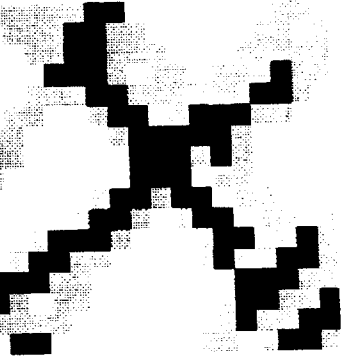
npts : 6 sigma : 0.061349 0.000016



npts : 6 sigma : 0.000019



0 1 6 7 +



PATTERNS

ANGLES

Fig. 1

1000 events WITH random hits added

of wrong peaks in histogram : 9

of events with $\sigma > 0.0001$: 2

remaining events : 989

of ITERATIONS : 0 (method 2 : 185)

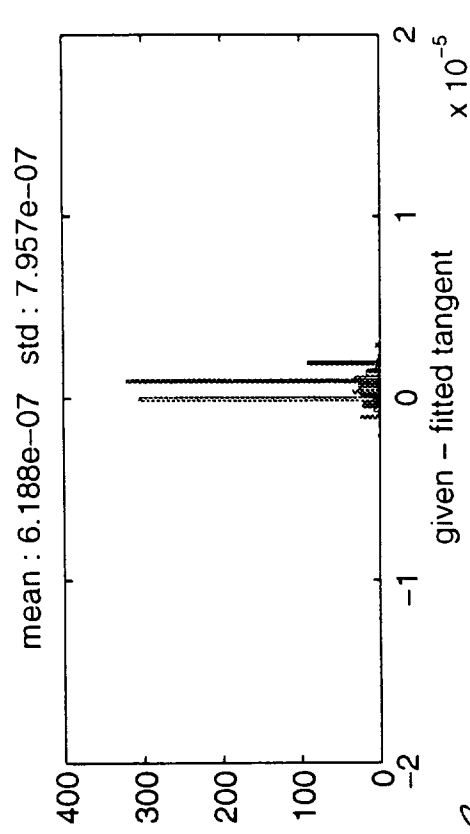
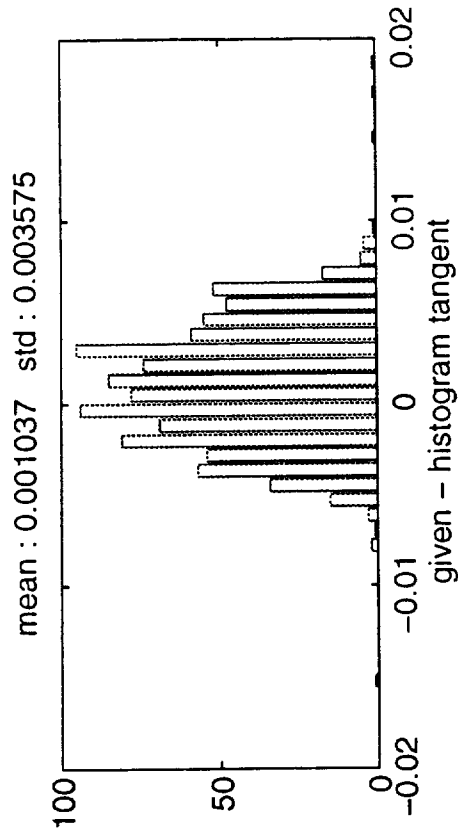
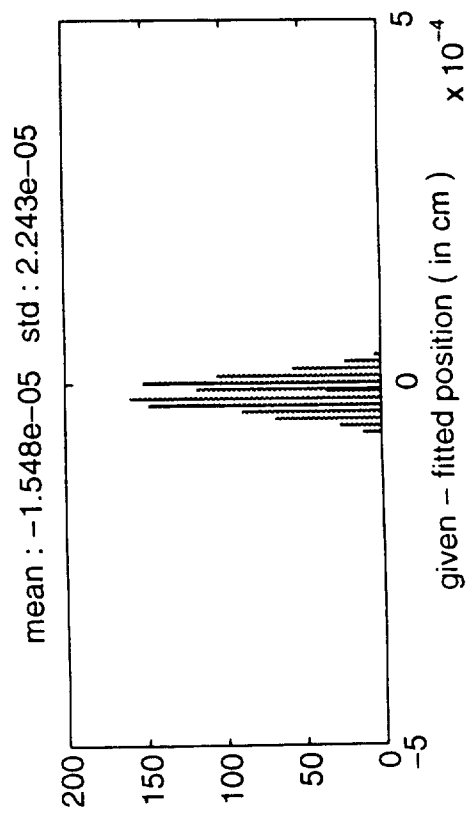
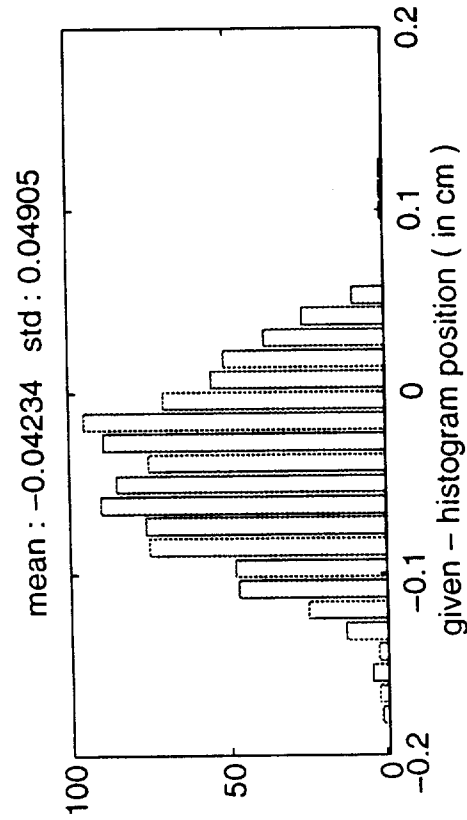
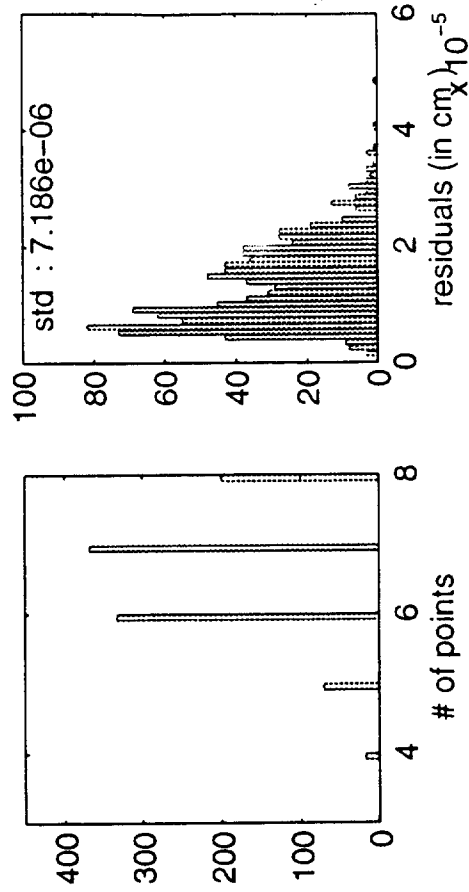
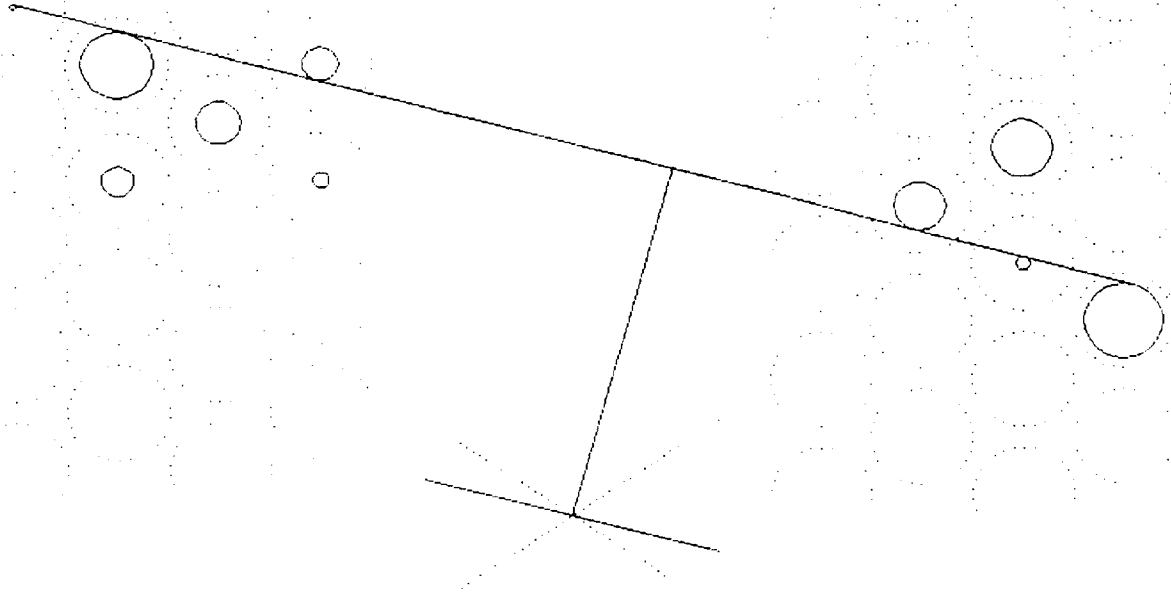
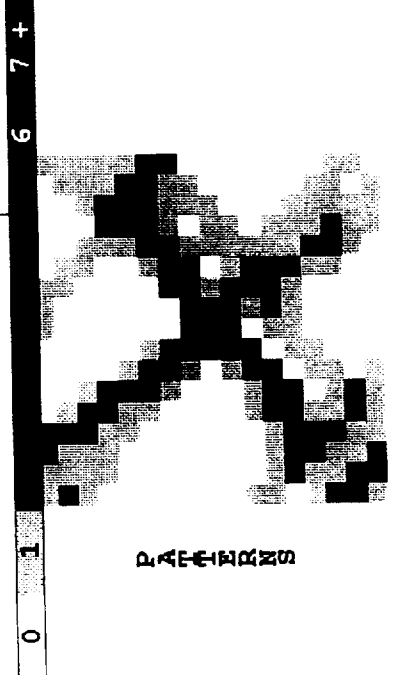


Fig. 2

npts : 6 sigma : 0.061349 0.000016



npts : 6 sigma : 0.000019



ANGLES

Fig. 3

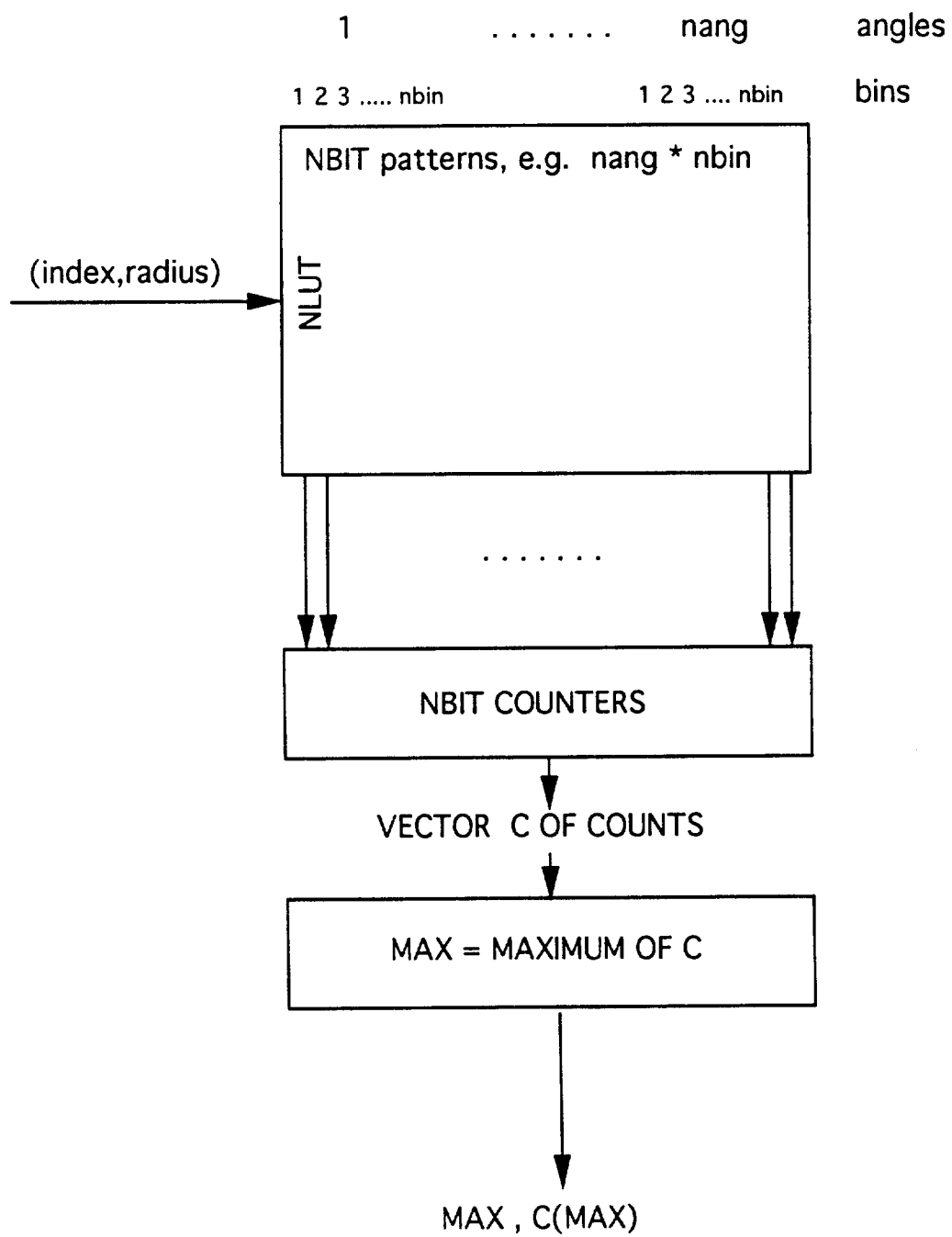


Fig. 4

6 20.00 0.08 65.4

EVENT= 46

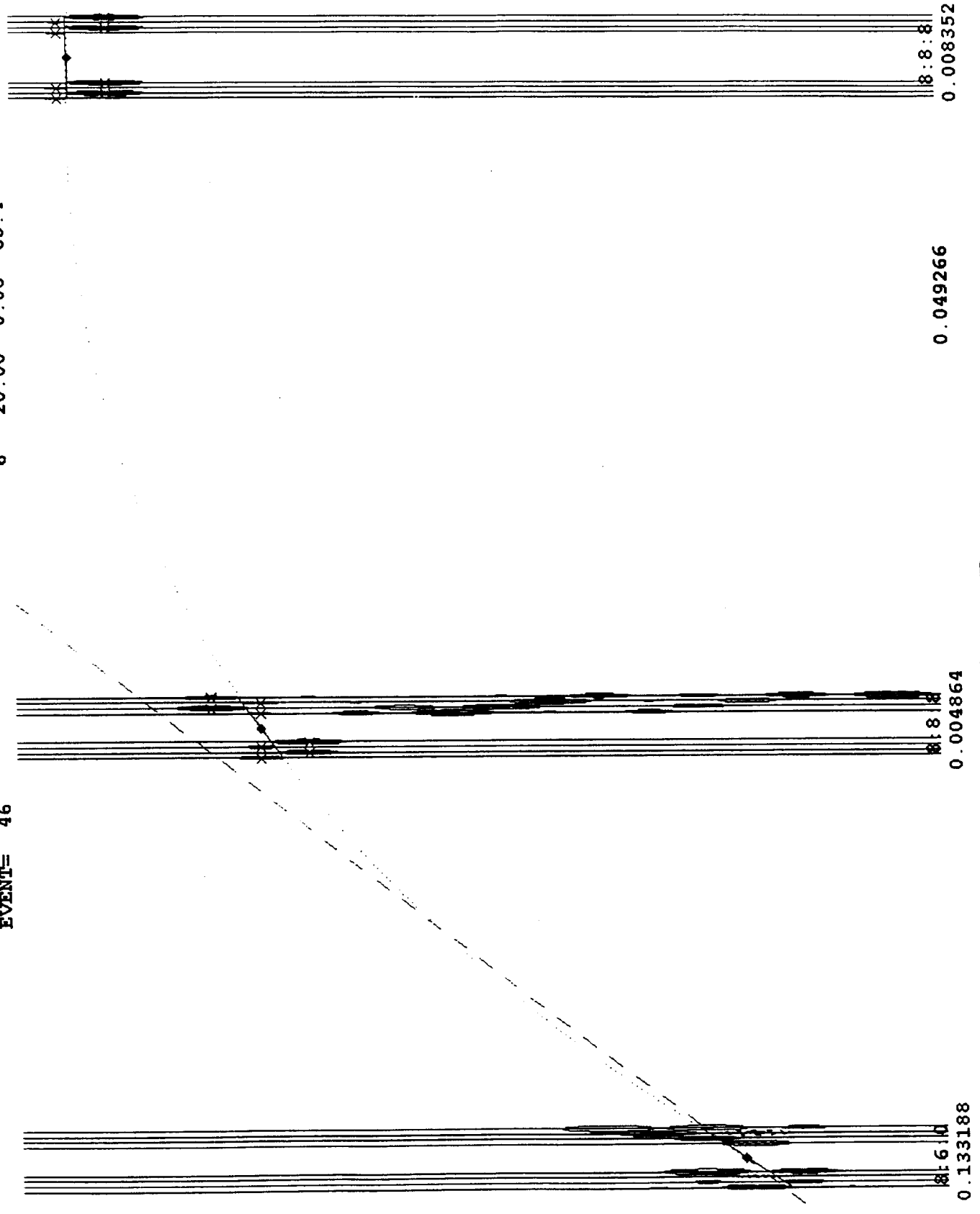
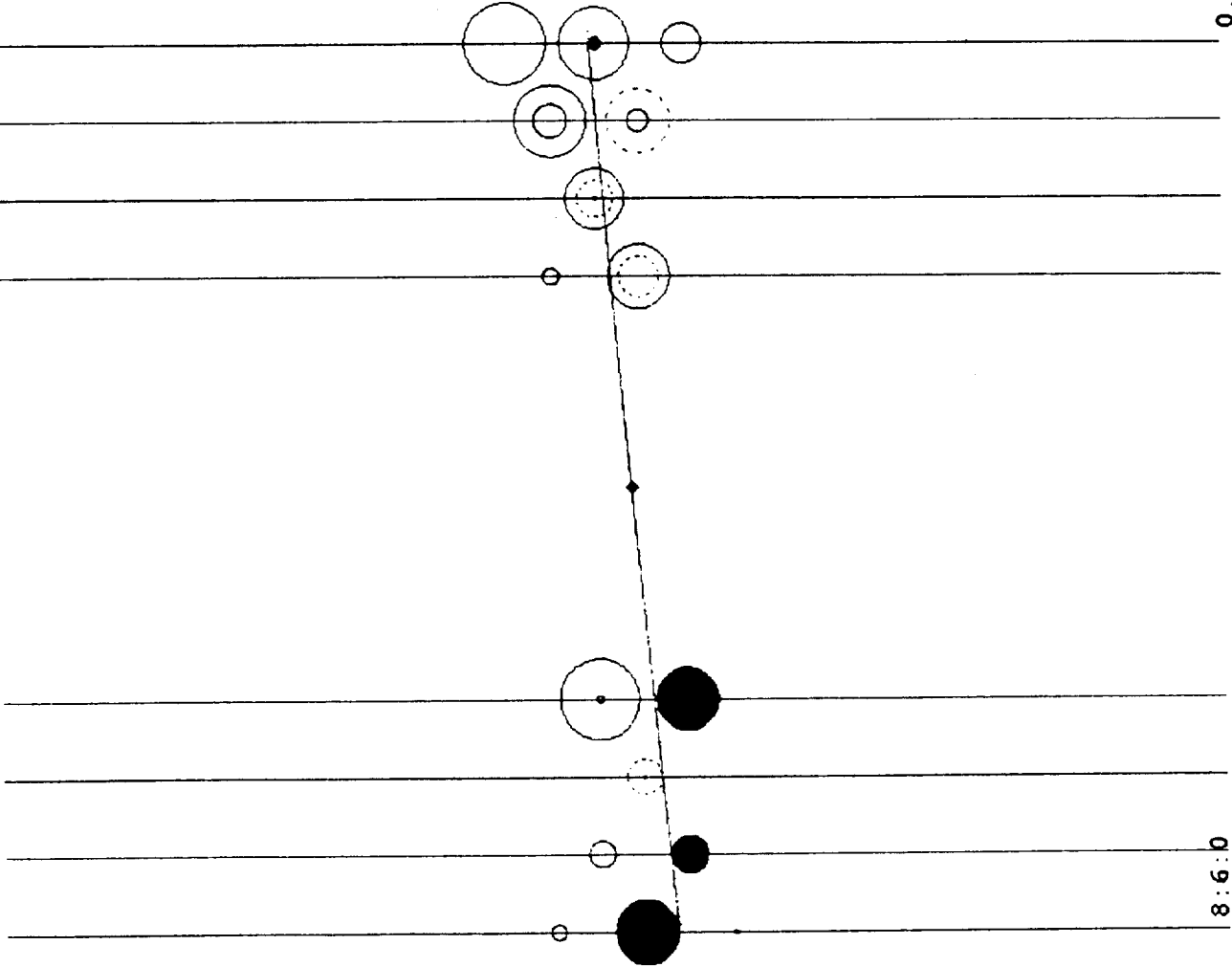


Fig. 5

EVENT= 46

6 20.00 0.08 65.4



0.008352

0.049266

0.004864

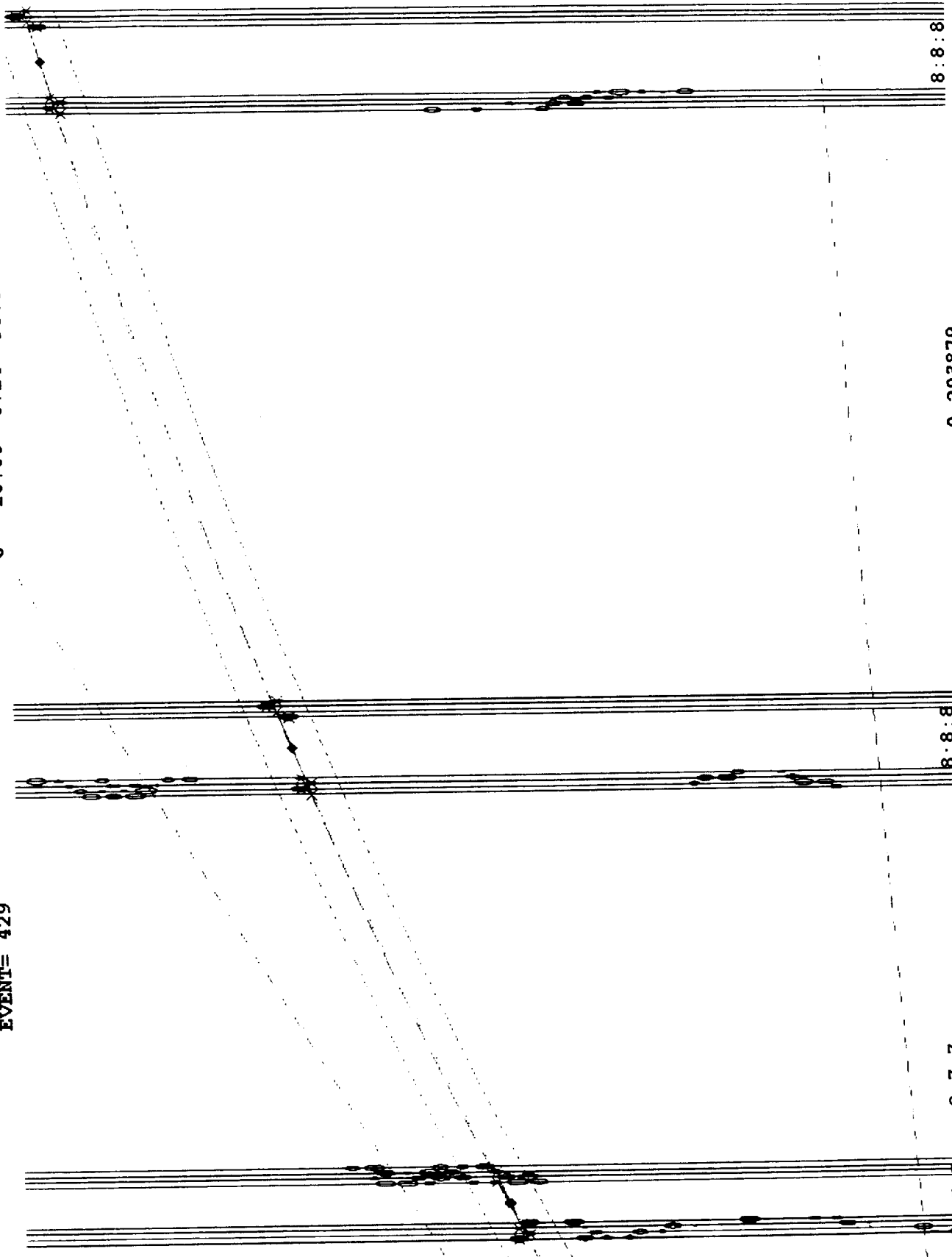
8:6:0

0.133188

Fig. 6

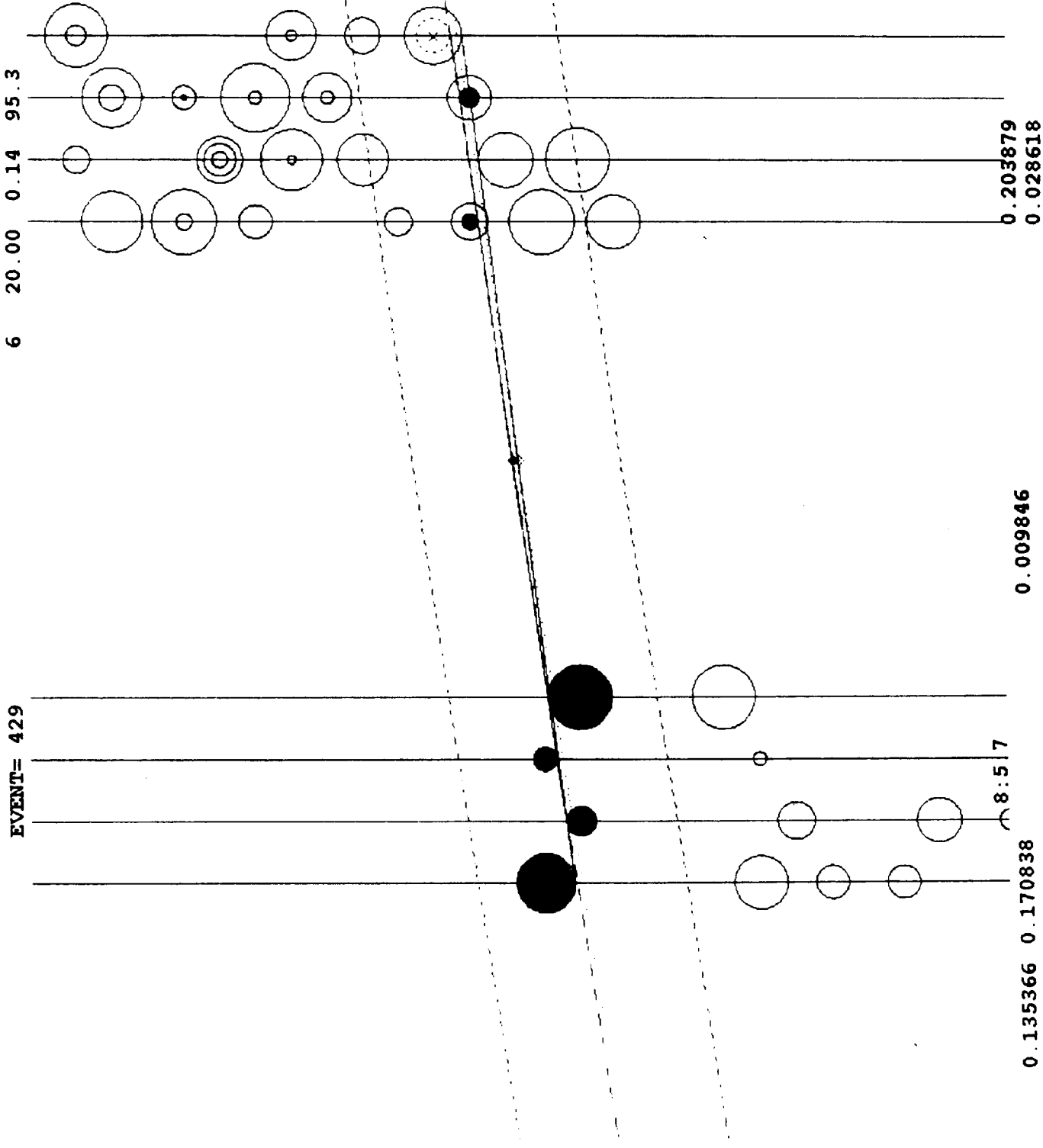
EVENT= 429

6 20.00 0.14 95.3



8:8:8
0.007723

Fig. 7



0.007723

0.203879
0.028618

0.009846

8:57

0.135366 0.170838

Fig. 8

3885 events in "BIG" chambers

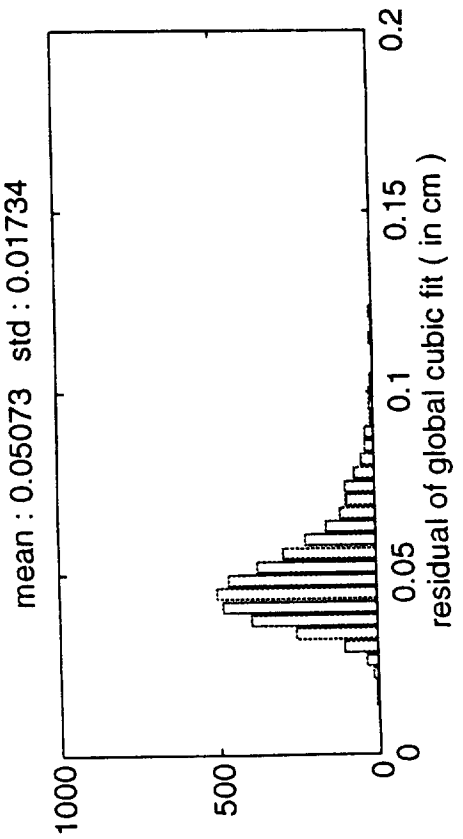
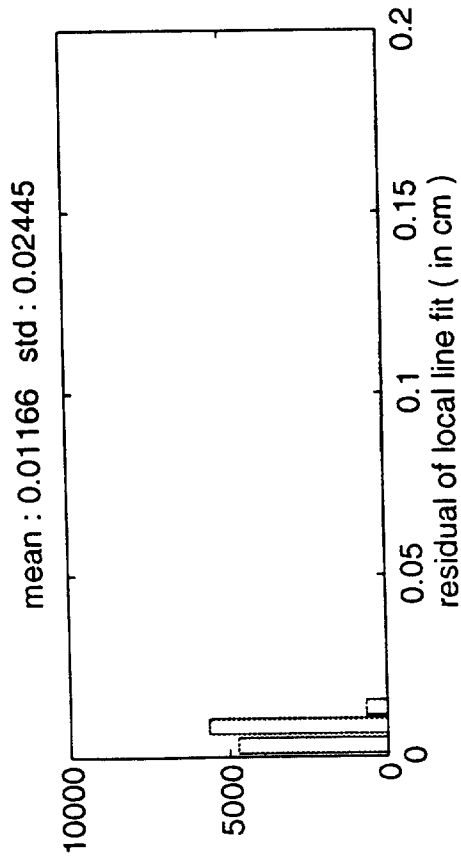
80 < phi < 100 & 0 < eta < 0.8

of hits : mean : 30.31 std : 12.86

of events with sigma > 0.2 : 7

remaining events : 3878

of events with ITERATIONS : 39



746 events in "SMALL" chambers

65 < phi < 70 & 0 < eta < 0.1

of hits : mean : 28.62 std : 14.05

of events with sigma > 0.25 : 7

remaining events : 739

of events with ITERATIONS : 6

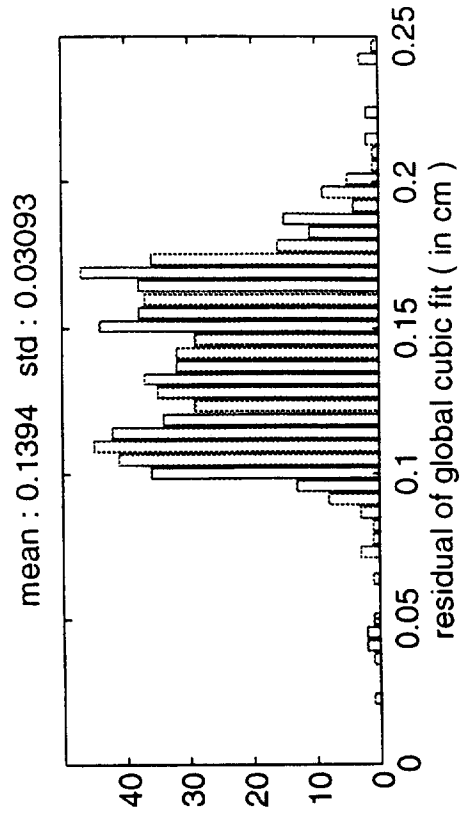
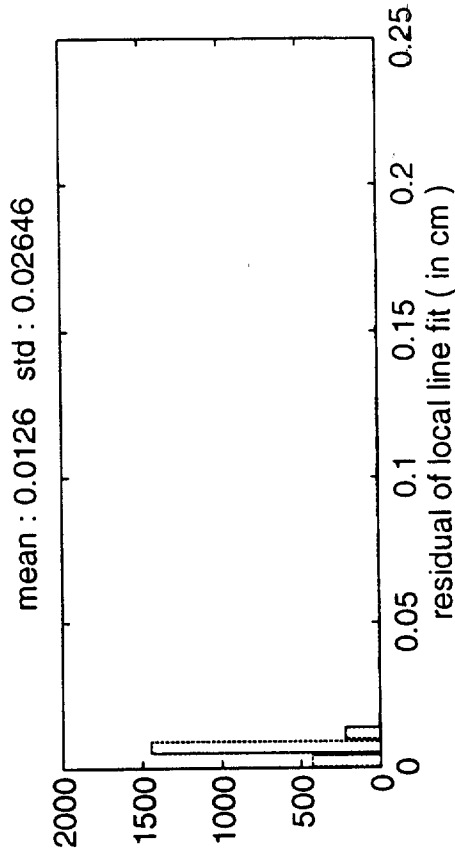


Fig. 9