E10-2007-88

S. Lebedev*, G. Ososkov, C. Hoehne[1]

RING RECOGNITION IN THE CBM RICH DETECTOR

[1]Gesellschaft für Schwerionenforschung mbH, Darmstadt, Germany

*E-mail: salebedev@jinr.ru

Лебедев С., Ососков Г., Хене К. E10-2007-88

Распознавание колец черенковского излучения
в RICH-детекторе эксперимента CBM

В данной работе описываются два алгоритма для поиска колец черенковского излучения в RICH-детекторе эксперимента CBM: автономный алгоритм (требуется информация только об отчетах RICH-детектора) и алгоритм, использующий информацию из вершинного детектора STS. Проблема нахождения ложных колец и ее решение с использованием искусственной нейронной сети также обсуждается в данной работе. Разработанные алгоритмы были протестированы на большой статистике смоделированных событий и включены в CBM Framework.

Работа выполнена в Лаборатории информационных технологий ОИЯИ.

Сообщение Объединенного института ядерных исследований. Дубна, 2007

Lebedev S., Ososkov G., Hoehne C. E10-2007-88

Ring Recognition in the CBM RICH Detector

Two algorithms of ring recognition, a standalone ring finder (using only RICH information) and an algorithm based on the information from vertex tracks, are described. The fake ring problem and its solution using a set of two-dimensional cuts or an artificial neural network are discussed. Results of a comparative study are given. All developed algorithms were tested on large statistics of simulated events and were then included into the CBM framework for common use.

The investigation has been performed at the Laboratory of Information Technologies, JINR.

Communication of the Joint Institute for Nuclear Research. Dubna, 2007

1. INTRODUCTION

Two algorithms of ring recognition for the CBM RICH detector [1] will be described in this paper: a standalone ring finder (using only RICH information) and an algorithm based on the information from vertex tracks. In the first approach the Hough transform (HT) is combined with preliminary area clustering in order to decrease combinatorics. The second, track-based (TB) method, is much simpler, but it depends a lot on the quality of the track extrapolation. Since the ring recognition results are seriously affected by many fake rings formed by random combinations of hits, two alternative approaches to solve the fake ring problem are proposed: either by applying a set of two-dimensional (2D) cuts or by using an artificial neural network.

The note is organized as follows: In Section 2, we introduce the CBM RICH detector, the event reconstruction scheme, and formulate the problem of ring recognition. In Section 3, we discuss the steps of our algorithm in detail. Starting from an HT overview and its special features for ring finding, we give details of the implemenation of the HT algorithm. Then we discuss how the HT computation was speeded up by a coarse histogramming and a proper clustering algorithm. Next, ring center and radius recognition using histrograms filled by the HT are described. Also in this section we describe our second, TB algorithm and discuss its drawbacks. In Section 4, we explain the fake ring problem and propose a solution using an artificial neural network. In Section 5, results applying both methods are presented. Finally, in Section 6, we present conclusions and an outlook.

2. A BRIEF OVERVIEW OF THE CBM RICH DETECTOR

The RICH detector in CBM will serve for electron identification from lowest momenta up to 10–12 GeV/c needed for the study of the dielectronic decay channel of vector mesons. In the current CBM detector layout (Fig. 1) the RICH would be positioned behind the magnet with the silicon tracking system (STS/MVD) and in front of the first transition radiation detector (TRD). The RICH detector used for this investigation was filled with nitrogen ($\gamma_{\rm th} = 41$), the radiator length was 2.5 m, the mirror radius 4.5 m, and as photodetector either a proposal from IHEP

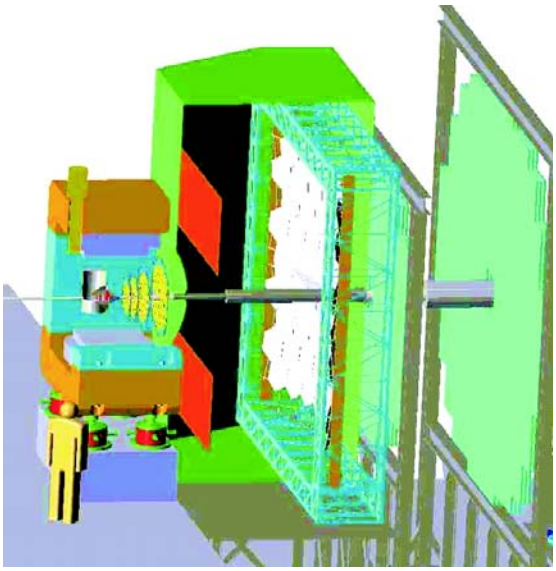

Fig. 1. Layout of the RICH detector to be positioned behind a large aperture dipole magnet with a silicon tracking system inside. The RICH will be followed by several transition radiation detectors serving for further electron identification and tracking

(Protvino) or MAPMTs from Hamamatsu (H8500-03) was implemented. This setting provides electron rings with about 6 cm radius and 40 or 22 hits/ring, respectively. As typical reactions for CBM, central Au+Au collisions at 25 AGeV beam energy were simulated.

The event reconstruction in the RICH detector includes several steps (see Fig. 2). First, STS tracks are extrapolated to a virtual plane in front of the

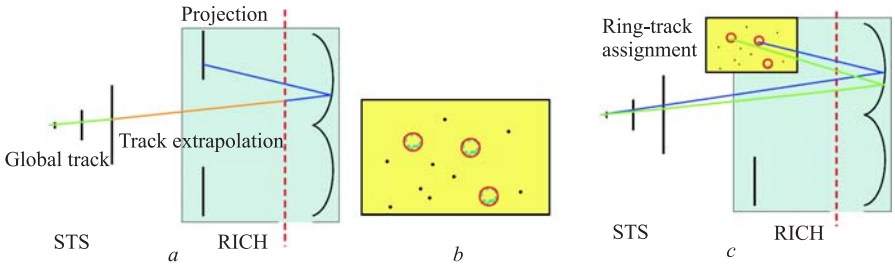

Fig. 2. *a*) Schematic sketch of the STS and the RICH detector, track extrapolation and track projection onto the photodetector plane; *b*) schematic sketch of RICH hits and found rings; *c*) ring-track matching

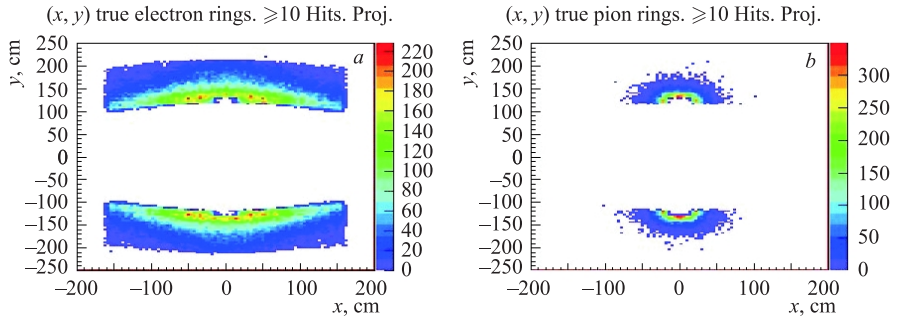

Fig. 3. Distribution of ring centers on the photodetector plane for electron rings (a) and pion rings (b)

mirrors. These tracks are then projected onto the photodetector plane providing track positions in this plane. The second step begins with the so-called hit producer which digitizes the MC points. Next, rings will be searched for; i. e., certain hits will be grouped to a ring. A ring fit will provide more precise values of its radius and center. The last step is to match the found rings with the extrapolated tracks in order to add them to the bank of global tracks. The main problems of ring recognition are [2]:

- Large number of hits (about 100–120 rings with approximately 22–40 hits/ring in each central Au+Au collision at 25 AGeV), many track projections from the STS (appr. 600 charged tracks per central collision (Au+Au, 25 AGeV)).

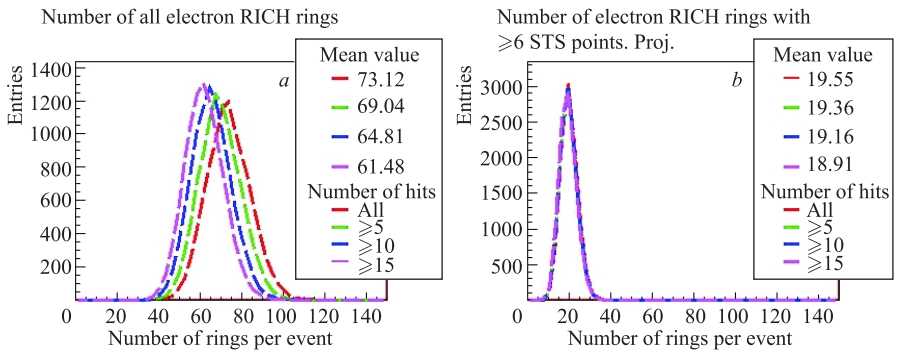

Fig. 4. Number of electron rings per event (central Au+Au collisions at 25 AGeV plus $5e^+$ and $5e^-$ added at the primary vertex): all electron rings (a), electrons from the primary vertex (b)

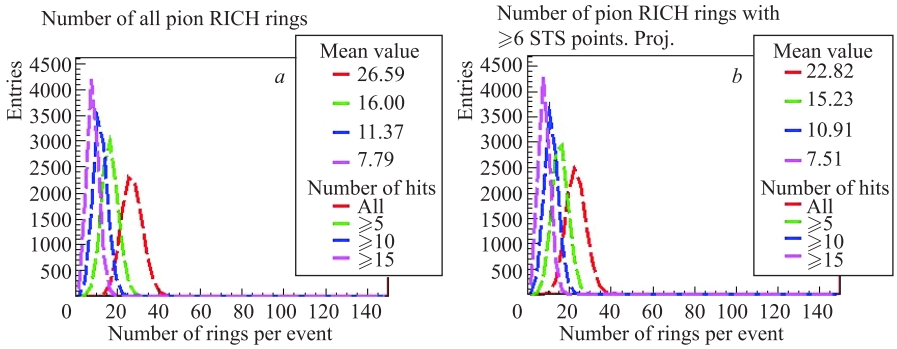

Fig. 5. Number of pion rings per event: all pion rings (*a*), rings from pion from the primary vertex (*b*)

- High ring density especially close to the beam (central part of photodetector plane) (see (Fig. 3).
- Overlapping rings.
- Low number of hits and small radius for pion rings (Cherenkov threshold 5.6 GeV/c) (see Fig. 3, *b*).

For the ring recognition routines it is important that most of the electron rings have more than 15 hits per ring.

For central Au+Au collisions at 25 AGeV beam energy the situation described in the following gives a typical picture: More than 70% of the electron rings are from secondary electrons produced elsewhere in the CBM detector (Fig. 4).

Due to the high Cherenkov threshold and the momentum distribution of pions having their maximum far below 5 GeV/c, about half of the rings from pions have less than 5 hits and only 25% have more than 15 hits per ring. Most of the pion rings have their track projection in the photodetector plane (Fig. 5).

3. ALGORITHMS

3.1. Ring Finding Based on Extrapolated Tracks. This approach uses additional information from the STS. Each track extrapolation from the STS on the photodetector plane is considered as potential ring center (a predictor) (Fig. 6). The distances between this predictor and all closest hits taken from an area determined by a fixed radius are calculated and filled into a histogram depicted in Fig. 7.

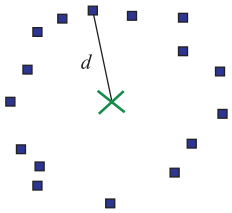

Fig. 6. Ring candidate

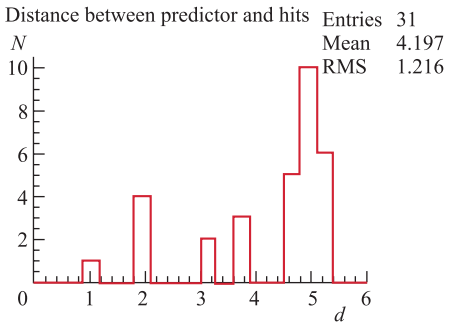

Fig. 7. Histogram of distances

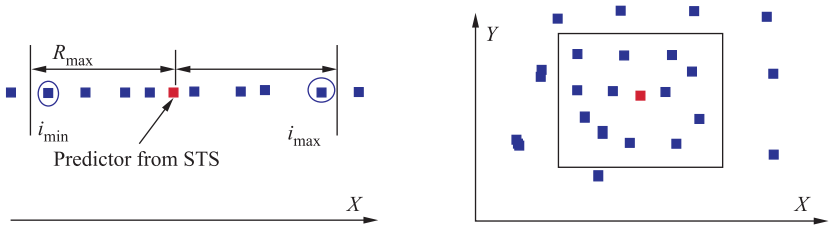

Fig. 8. Finding all hits in corridor defined around the predicted point

This histogram allows one to decide whether a ring was found: The highest peak corresponds to a ring candidate. This peak is determined by first searching for the bin with the highest value, and then summing up the content of this and the two neighboring bins. If the result exceeds a prescribed cut, the ring candidate is accepted, otherwise it is rejected. The center of gravity of these three bins is calculated to evaluate the ring radius. The found ring center is considered as coordinate of the predictor.

As for all track extrapolations the distances to all hits have to be calculated, one gets a large number of combinatorics. In order to reduce combinatorics the following algorithm is applied, as illustrated in Fig. 8. First, all hits are sorted by their X coordinate. Then, one loop over all predictors and a corridor is made in X on both sides of the predictor which is equal to the maximum radius ($R_{\max}$, in our case $R_{\max} = 7$ cm). As all hits are preliminarily sorted, it becomes very simple and fast to find the first and the last index of the hits in the corridor ($i_{\min}$ and $i_{\max}$). Then all hits in this corridor are checked by their Y value.

3.2. Hough Transform Overview. The Hough transform [3, 4] is a standard method for shape recognition in digital images, e. g., finding straight lines, circles, ellipses. Using the HT, one searches for curves which pass through a sufficient

number of points of interest. A set of curves on a plane is specified by the parametric equation $F(a_1, a_2, a_3, ..., a_n, x, y) = 0$**,** where F is some function, a_1, a_2,... are parameters of the set of curves, and x, y are the coordinates on the plane. The parameters of the curves form the space of parameters (or Hough space), each point in this space (specific value of parameters a_1, a_2,...) corresponds to some curve.

3.3. Ring Center and Radius Evaluation. The parametric equation of a circle is $(x-a)^2+(y-b)^2 = R^2$, where a and b are the coordinates of the circle center and R is the radius. The equation $F(a, b, R, x, y) = (x-a)^2 + (y-b)^2 - R^2$ therefore describes a set of circles**.** With any three points (a triplet) one can unambiguously determine a circle, i. e., these three parameters (a, b, R). This can be done via the following system of equations:

$$\begin{cases} (x_1-a)^2 + (y_1-b)^2 = R^2; \\ (x_2-a)^2 + (y_2-b)^2 = R^2; \\ (x_3-a)^2 + (y_3-b)^2 = R^2, \end{cases}$$

where x_1, y_1, x_2, y_2, x_3, y_3 are the coordinates of the 1st, 2nd, 3rd point, correspondingly. Solving these equations, one gets the following values for the center coordinates and the radius:

$$a = \frac{1}{2}\frac{(x_2^2 - x_3^2 + y_2^2 - y_3^2)(y_1 - y_2) - (x_1^2 - x_2^2 + y_1^2 - y_2^2)(y_2 - y_3)}{(x_2 - x_3)(y_1 - y_2) - (x_1 - x_2)(y_2 - y_3)},$$

$$b = \frac{1}{2}\frac{(x_1^2 - x_2^2 + y_1^2 - y_2^2)(x_2 - x_3) - (x_2^2 - x_3^2 + y_2^2 - y_3^2)(x_1 - x_2)}{(x_2 - x_3)(y_1 - y_2) - (x_1 - x_2)(y_2 - y_3)},$$

$$R = \sqrt{(x_1 - a)^2 + (y_1 - b)^2}.$$

In practice, (a,b,R) are not calculated for all triplets, but a preliminary search of significant areas is performed. This is necessary because of the huge number of hits and the resulting strong growth of combinatorics and calculation time. Furthermore, this helps to reduce incorrect combinations of triplets in advance; therefore, it also reduces the number of false (noise) points in the space of ring centers. For a fast search of triplet candidates the hits are sorted at the beginning using the algorithm which was described before in the context of the track based ring finder.

The HT algorithm (see Fig. 9) works in several iterations. The number of iterations is set by the user, also the HT parameters can be changed for each iteration. In the first iteration, the algorithm finds all «good» rings, which have many hits and therefore high peaks in the ring center histogram. After each

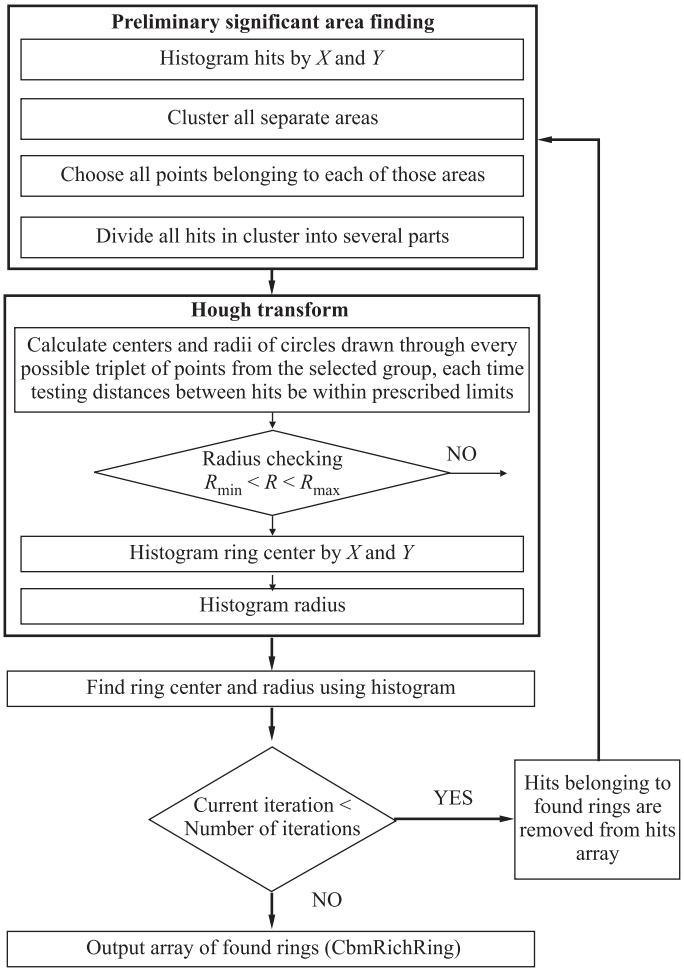

Fig. 9. HT algorithm flowchart

iteration all hits which belong to a found ring are removed from the array of hits. Each iteration is divided into several steps.

First, a coarse histogramming of the source data by X and Y position is done, e. g., in Fig. 10 a histogram with 50×50 cells is presented.

Next, all separated areas are clustered and the points belonging to each of those areas are selected. A fragment of this histogram is shown in Fig. 11. One can see that the clustering splits this fragment into three areas corresponding to

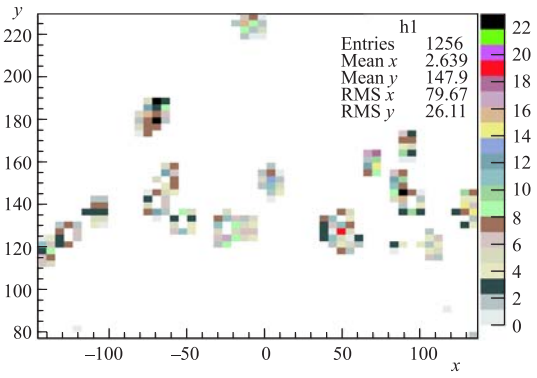

Fig. 10. Histogram of RICH hits, coarse binning

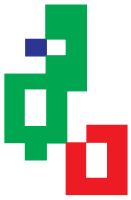

Fig. 11. Three clusters of RICH hits in the coarse histogram

three rings, two of them belong to one cluster. Then, the hits of each cluster are divided by a fixed number (2 or 3), which allows one to reduce combinatorics.

The next step is to calculate the center and radii of circles drawn through every possible triplet of points $(x_i, y_i;\ i = 1, 2, 3)$ from the selected group, using the formulas which were described above. Each time the distances between points and obtained radii of a triplet are tested to be within prescribed limits. In this way the hits are transformed to the parameter space (a, b, R), i. e., the HT is applied. The ring centers $(X_{\text{center}}, Y_{\text{center}})$ obtained by this step are collecetd in a two-dimensional histogram, see Fig. 12 for a binning of 100×100 cells. The averaged radii for each triplet from the given histogram bin are also calculated.

From 2D histogram shown in Fig. 13 all bins with a content less than a given limit are eliminated. This limit differs from iteration to iteration; i. e., it is reduced in each step.

Ring candidates are selected by again clustering all separate areas of $\{x_c, y_c\}$ and choosing all centers belonging to each of those areas. The approximate center and radii of ring candidates in these areas are extracted according to the following rules:

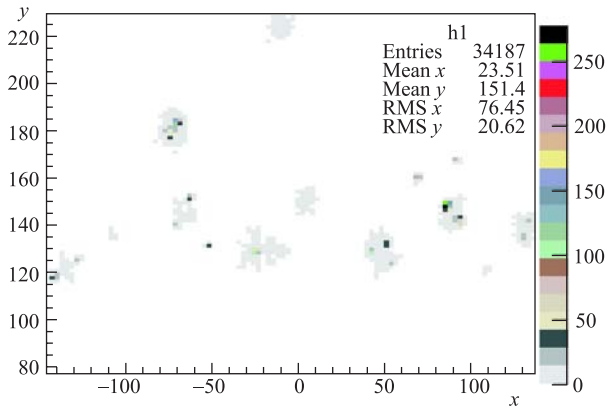

Fig. 12. Histogram of ring centers calculated in the HT

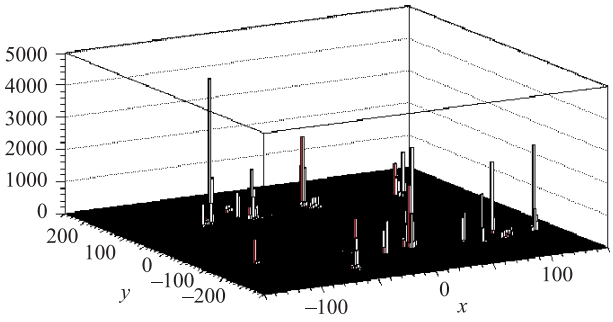

Fig. 13. 2D histogram of ring centers

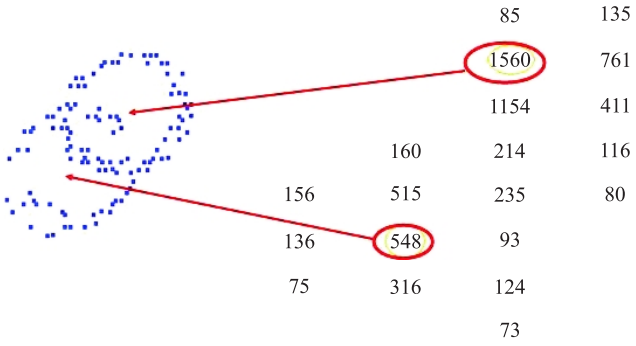

Fig. 14. Number of centers accumulated in the bins (numbers on the right) and corresponding rings (left)

1. Initiate the search by marking all (x, y) bins as non-searched.
2. Look for a group of bins with maximum values ($x_{\max}$, $y_{\max}$) in these non-searched bins selection.
3. Calculate the center of this group of bins as the center of gravity and mark these bins as searched.
4. Go on until exhausting all non-searched bins.

As an example, one area and the corresponding rings are shown in Fig. 14. The contents of every bin (numbers on the right) present the number of triplet centers accumulated in the given bin during the histogramming (HT) process.

Two maxima are seen corresponding to two ring centers.

4. REJECTION OF FAKE RINGS

4.1. Overview. The ring finder finds not only «true» rings but also «fake» rings by random combinations of hits in the photodetector plane. These fake rings have to be rejected after the ring finding is done. A typical scenario for the reconstruction of fake rings is the «stealing» of hits from nearby rings, see Fig. 15.

In order to reject reliably these fake rings, a set of parameters of found rings had to be selected which could be used for fake rejection. For this task, parameters had to be found which differ for fake and true rings as much as possible. Then, a set of cuts based on these parameters was developed to reject fake rings, however, trying not to drop down the ring finding efficiency for «good» rings. Finally, seven parameters were selected. In Fig. 18 these parameters are shown for

- true electron rings (dashed line),
- true pion rings (dotted line).

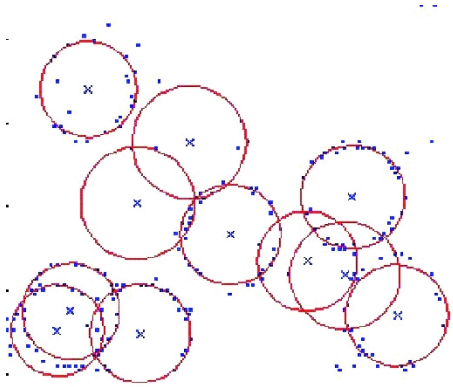

Fig. 15. Example of typical fake rings

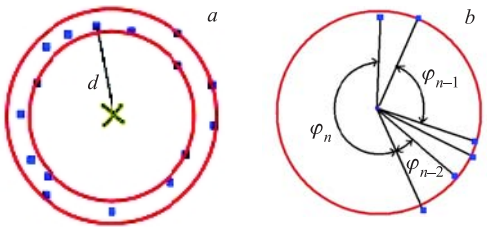

Fig. 16. Narrow corridor around the ring (*a*) and angles between neighbouring hits (*b*)

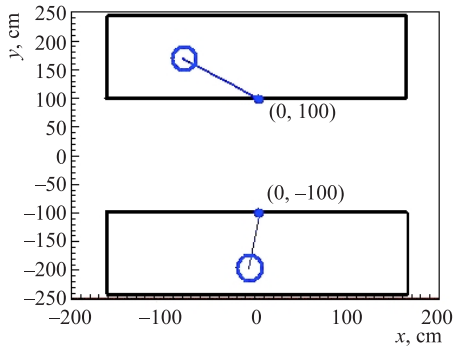

Fig. 17. Radial position on photodetector plane

The following parameters were calculated and are presented in Fig. 18:

- a) All distances between a found ring center and the hits belonging to the ring (see the figure) were calculated and filled in a histogram. Then, the maximum peak in the histogram was selected and all hits lying within a certain, narrow corridor were counted, as shown in Fig. 16, *a*. In this way we calculate the number of hits ($\mathrm{TB_{sum}}$) which lie in a narrow corridor around the ring radius.
- b) Number of hits in the ring. One can see in Fig. 18, *b* that true electron rings have more hits than fakes.
- c) Distance between the closest track projection and ring center. Most of fakes lie rather far from the extrapolated tracks.
- d) The sum of three biggest angles between neighbouring hits. A typical feature of fake rings is the non-uniform distribution of hits along the ring. This nonuniformity is quantified by calculating the angles between neighbouring hits and selecting the three maximum angles (see Fig. 16, *b*).
- e) The chi-squared criterion χ^2 of the ring fit. For rings χ^2 is determined

as follows:

$$\chi^2 = \frac{\sum_{i=1}^{N} (\sqrt{(x_c - x_i)^2 + (y_c - y_i)^2} - R)^2}{N - 3},$$

where x_c, y_c are the coordinates of the ring center, and x_i, y_i are the coordinates of the ith hit, N is the number of hits per ring, R is the radius.

- f) Position on the photodetector plane (radial position). A large number of fake rings are found in the inner part of the photodetector plane due to the large density of hits and rings in this region.
- Ring radius (not shown). As was mentioned before, electron rings typically have a radius of about 6 cm.

The first attempt of using a set of one-dimensional cuts on each of these parameters resulted in an insufficient rejection of fakes and a large loss in efficiency. Therefore, in a second attempt, a set of two-dimensional cuts was developed.

Table 1. 2D cuts

Number of 2D cut	Cut description
1	$\mathrm{TB_{sum}} \leqslant 12$ and Radial position < 36
2	The biggest angle > 2.5 and Radial position < 36
3	$\mathrm{TB_{sum}} \leqslant 12$ and Ring-track distance > 1.2
4	$\chi^2 > 0.4$ and Radial position < 40
5	$\chi^2 > 0.6$ and $\mathrm{TB_{sum}}$ / Number of hits < 0.42
6	Ring-track distance > 1.0 and Radial position < 40

Here, each 2D cut is a combination of two 1D cuts. Pairs of parameters are combined and filled in 2D histograms for fake rings, true electron and pion rings separately (see, e. g., Fig. 19). Then, cuts are made in this 2D space, which remove the main part of the fake rings. One example of such a pair of parameters is shown in Fig. 19: in the upper row without, in the lower with applying a chosen cut.

Results of using the sample of 2D cuts as summarized in Table 1 are presented in Section 5.

4.2. Artificial Neural Networks. An alternative approach to a reliable rejection of fake rings is the application of an artificial neural network (ANN). By input values of ANN it is necessary to define whether the ring is correctly found or not. We used the standard ROOT class — TMultilayerPerceptron, which is an implementation of multilayer perceptron. In our case, the ANN consists of seven

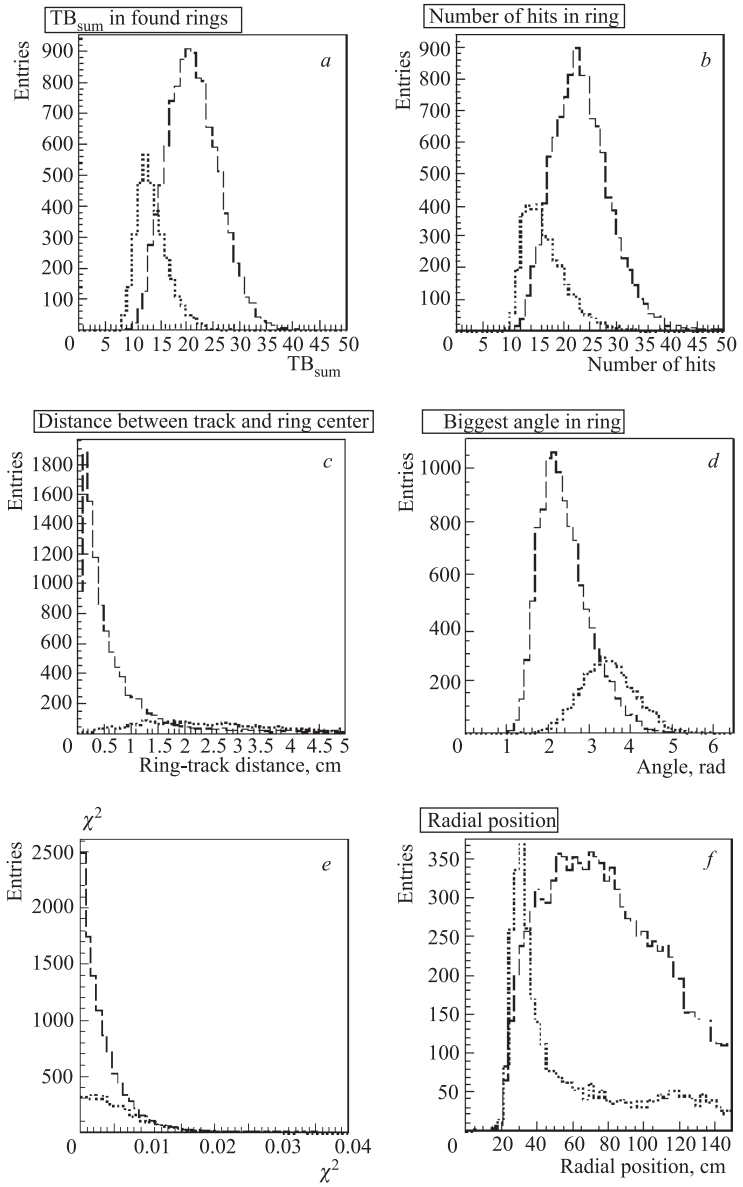

Fig. 18. Parameters selected to distinguish between good (dashed line) and fake (dotted line) rings, for explanation see text. (Hamamatsu type PMT was simulated)

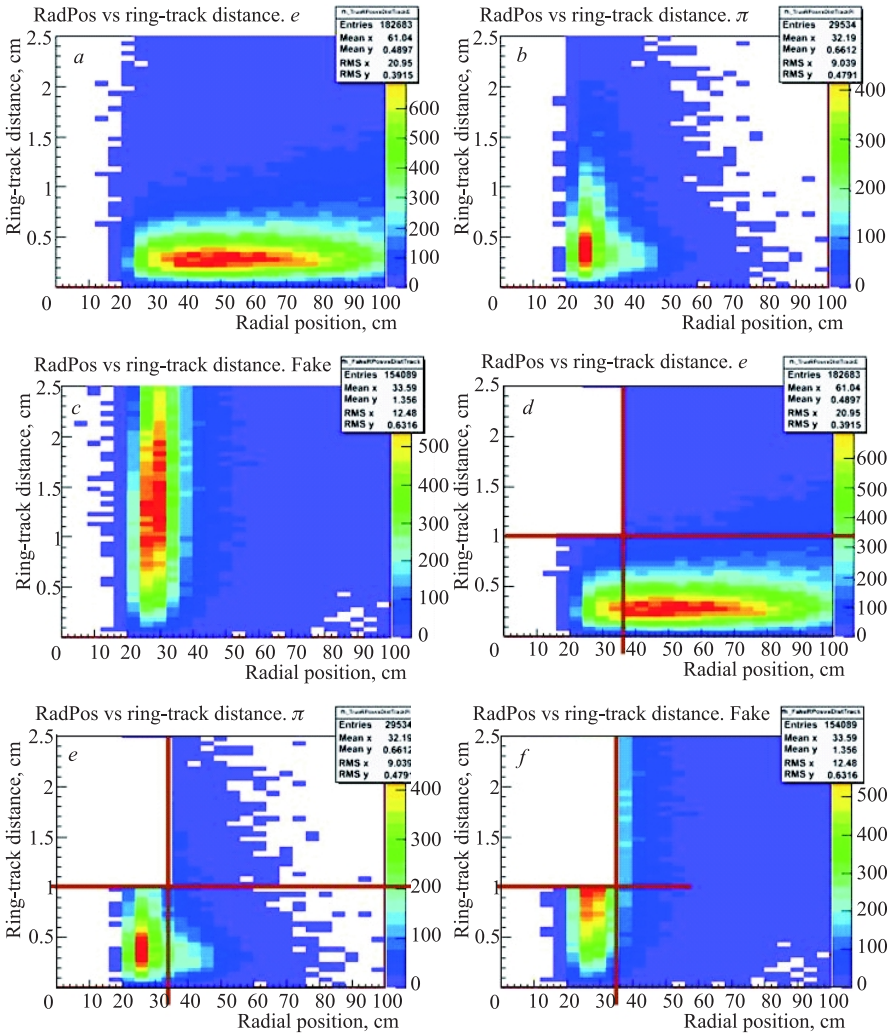

Fig. 19. Example of 2D histograms for a parameter pair combination (radial position vs distance between ring and closest track projection) From left to right: electron, pion and fake rings. The lower row shows the selected cut

input neurons corresponding to the seven parameters which were described above, 30 hidden neurons and one output neuron. In the ANN training, the output value of fake rings was set to «one» and the one for true, i. e., correctly found rings

to «zero». The training of the ANN was based on the back propagation error method and used 20000 samples for training (10000 true rings and 10000 fake rings). For testing we used 300000 rings, 150000 fake and 150000 true rings. Note that the output neuron value is not binary: for fake rings this value lies in the region around 1, and for correctly found rings it concentrates around 0. Therefore, it is necessary to make a cut on the output value, here we choose 0.6 for separating good and fake rings. Histogram of ANN output values is depicted in Fig. 20. The chosen threshold is shown by arrow. This cut results in:

- Number of fake-like true rings is 9467 (prob. 2nd kind error = 6.3% = 9467/150000 · 100%).
- Number of true-like fake rings is 6107 (prob. 1st kind error = 4.0% = 6107/150000 · 100%).

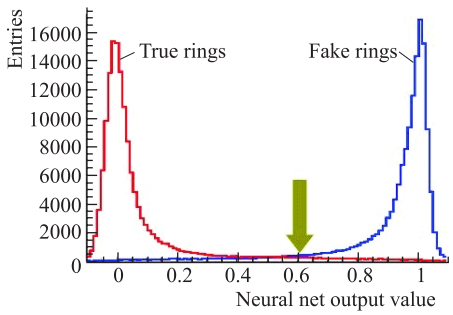

Fig. 20. ANN output value for true and fake rings for the test sample of 300000 rings

In Fig. 21 a two-dimensional histogram of the same parameter pair selection as in Fig. 19 is shown, this time presenting the ring selection done by applying the ANN and a cut value of 0.6.

5. RESULTS

In the following, results of the ring recognition routines will be presented as well as of the fake ring rejection applying the methods introduced above.

Efficiencies and ring quality can be classified using MC information: Fake rings are defined by having less than 60% of hits belonging to one MC ring. For all non-fake rings, the MC information allows one to check whether, e. g., this ring was then matched to the correct track.

The ring finder efficiency is classified by three parameters:

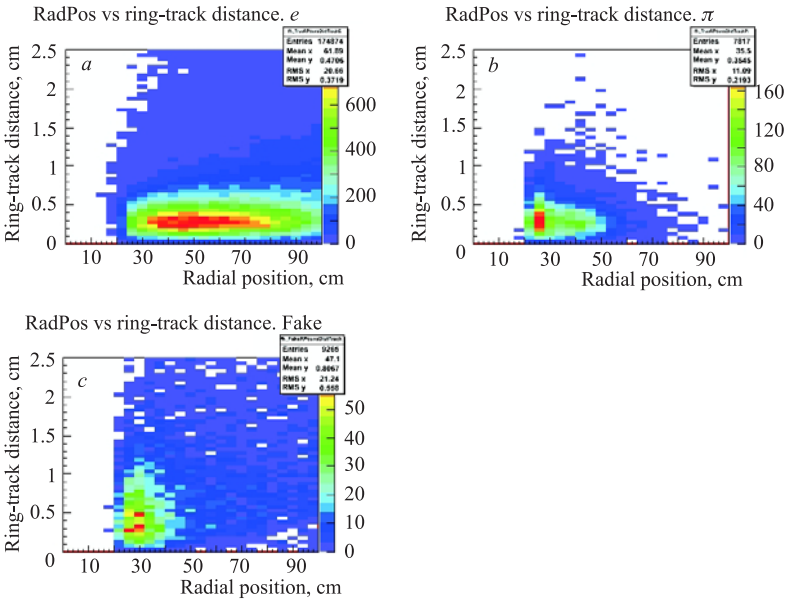

Fig. 21. Example of 2D histograms for a parameter pair combination (radial position vs distance between ring and closest track projection): electron (*a*), pion (*b*) and fake rings (*c*)

- Efficiency of electron ring finding, defined as

$$eff = \frac{N_{\text{rec}}}{N_{\text{ac}}} \cdot 100\%,$$

  where N_{rec} — number of correctly reconstructed primary electron rings (which have more than 60% hits corresponding to one MC ring), N_{ac} — number of reconstructable primary electron rings (more than 5 hits in RICH).
- Number of fake rings per event.
- Number of clone rings. A clone ring is defined as a ring which was found more than once.

For the results presented in the following, the simulation parameters as listed below were used:

- 10000 central Au–Au collisions at 25 GeV (besides $5e^+$ and $5e^-$ with uniform (θ, p_t, ϕ) distributions were added at the vertex in order to enhance the statistics for electrons from the primary vertex).
- cbmroot2, release JUN 2006.

- L1 STS track finder and Kalman Filter extrapolation to RICH.
- RICH geometry: N2 radiator; Protvino type PMT or Hamamatsu H8500-03 flat panel MAPMT.
- TAU radius fit and ring radius correction for geometrical distortions.
- Ring-track assign by closest distance approach.

5.1. Efficiency of Ring Finding. The efficiency of electron ring finding using the HT algorithm in dependence on p_t and rapidity is shown in Fig. 22. The mean efficiency is 95.36%, the average number of fakes per event is 15.41, of clones it is 7.07. The histogram in panel *b* shows the distances between ring center and closest track projection. All distances are shown by the red line, in blue are the distances for wrong ring-track matches or matches with fake rings, and in green are the true rings correctly matched with their track. Many wrong ring-track matches, in particular below 2 cm distance, stem from track matches with fake rings.

The efficiency of electron ring finding using the track-based algorithm in dependence on p_t and rapidity is shown in Fig. 23. The mean efficiency is 77.03%, the average number of fakes per event is 9.98, of clones it is 3.23. Please note that this ring finding method already includes ring-track matching. In contrast, for the HT additional efficiency losses for ring-track matching have to be accounted for a fair comparison. This efficiency of ring-track matching will be discussed below (Subsection 5.3).

5.2. Fake Ring Rejection Efficiency. After ring finding the routines for rejecting fake rings have been applied, here results will be presented using the HT ring finder. The efficiency of electron ring finding including the application

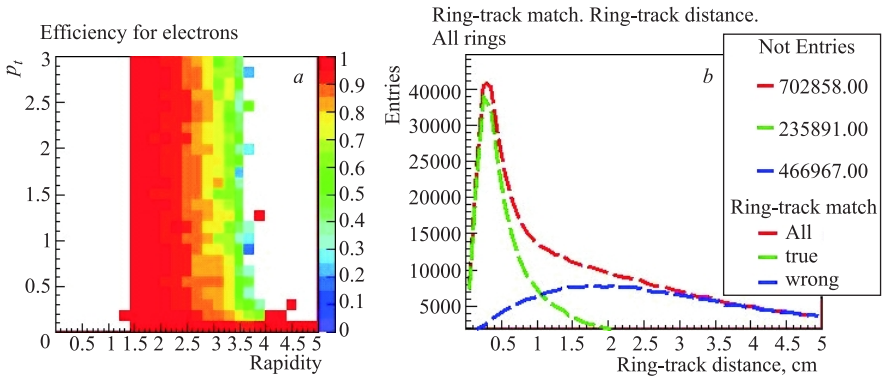

Fig. 22. *a*) Ring finding efficiency for electrons from the primary vertex in dependence on p_t and rapidity (HT algorithm, Protvino PMT). *b*) Ring-track matching, the closest distance between ring and track

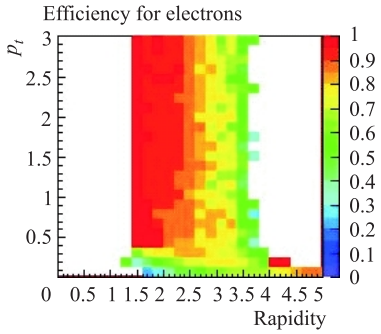

Fig. 23. Ring finding efficiency for primary vertex electrons in dependence on p_t and rapidity (track-based algorithm, Protvino PMT)

of the set of 2D cuts as introduced above is shown in Fig. 24 in dependence on p_t and rapidity. In panel *b* the distances between ring center and closest track projection are shown after applying the cuts. The mean efficiency is 91.39%, the average number of fake rings per event still remaining in the ring sample is 2.59 per event, of clones it is 2.23. Comparing these numbers with the ones given in the previous section, we conclude that the application of the cuts results in:

- 4.16% loss in efficiency for electrons,
- 5.94 times less fake rings,
- 3.17 times less clone rings,
- roughly 2 times less wrong ring-track matches.

The efficiency of electron ring finding including the application of the neural network with a cut on 0.6 is shown in Fig. 25 in dependence on p_t and rapidity. In panel *b*, the distances of rings and tracks after this cut are shown as well. The mean efficiency is 91.34%, the average number of fake rings remaining in the sample is 0.91 per event, of clones it is 1.24 per event. Again, comparing these numbers with the ones after the HT ring recognition alone, we conclude that the ANN results in:

- 4.21% loss in efficiency for electrons,
- 16.93 times less fake rings,
- 5.70 times less clone rings,
- 4 times less wrong ring-track matches.

Table 2 summarizes the efficiencies and fake and clone ring rates after applying the neural net ring selection with different cuts on the output value of ANN, here called N_n. The efficiencies in this table were obtained for 1000 events.

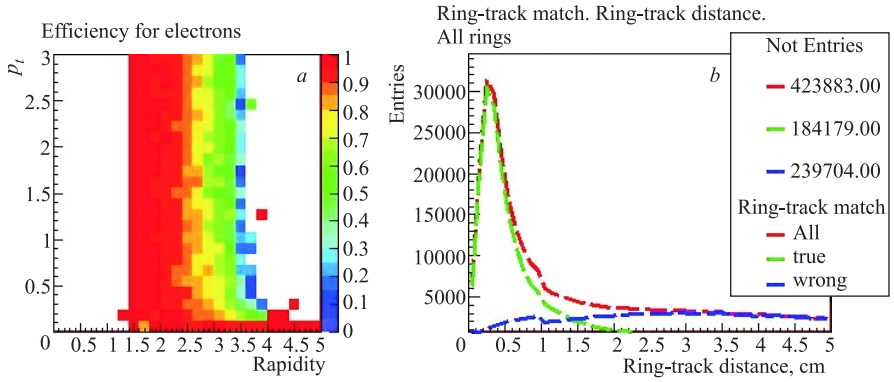

Fig. 24. *a*) Ring finding efficiency for primary electrons in dependence on p_t and rapidity (HT algorithm + 2D cuts, Protvino type PMT). *b*) Ring-track matching, the closest distance between ring and track (HT algorithm + 2D cuts, Protvino type PMT)

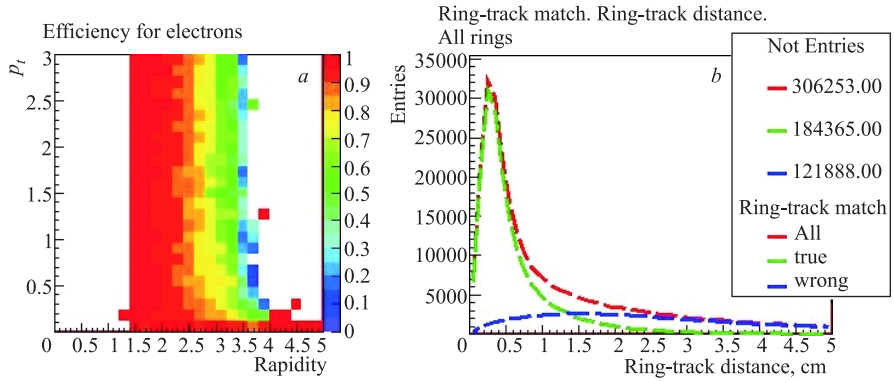

Fig. 25. *a*) Ring finding efficiency for primary vertex electrons in dependence on p_t and rapidity (HT algorithm + ANN cut 0.6, Protvino type PMT). *b*) Ring-track matching, the closest distance between ring and track (HT algorithm + ANN cut 0.6, Protvino type PMT)

To summarize, we conclude that

- Ring finding efficiency and fake ring rate are strongly correlated, the higher the efficiency, the more fake rings are in the sample, and vice versa.
- Neural net showed the best results in ring selection.
- ANN is rather flexible for the application of ring finding in physics analysis, because with only one parameter the efficiency or purity can be easily tuned.

Table 2. Primary electron efficiency and fake and clone ring rates per event after applying ANN with different cuts on the output value (HT algorithm, Protvino type PMT)

	No cut	$N_n > 0.8$	$N_n > 0.7$	$N_n > 0.6$	$N_n > 0.5$	$N_n > 0.4$	$N_n > 0.3$
Electrons, %	95.56	94.60	93.80	92.27	90.34	88.92	86.83
Fakes/event	15.64	4.65	2.98	1.60	1.42	1.01	0.64
Clones/event	7.15	2.76	2.16	1.70	1.58	1.40	1.1

5.3. Ring-Track Matching Efficiency. If a standalone ring finder is used, found rings have to be matched to tracks in a second step. As discussed above, currently this is done by selecting the closest track. Rings to be used in the analysis are finally selected by applying a cut on this distance. Again, the combined ring reconstruction efficiency and purity are strongly correlated and depend on this cut. The efficiencies after applying different distance cuts are presented in Table 3 using the following abbreviation: D is the distance between ring center and the closest track projection [cm]. $D > 1.0$, e. g., then means that all rings are rejected for which the distance between ring-center and track was larger than 1 cm. The efficiency of «HT» already includes the fake ring rejection based on the ANN with requiring an output value below 0.6.

Table 3. Efficiency vs different ring-track distance cut (HT algorithm, Protvino type PMT)

	HT	$D > 3.0$	$D > 2.0$	$D > 1.5$	$D > 1.0$	$D > 0.75$	$D > 0.5$
Electrons, %	91.45	87.33	85.33	78.58	73.77	66.33	52.14
Fakes/event	0.92	0.73	0.68	0.53	0.46	0.36	0.23
Clones/event	1.28	0.77	0.72	0.62	0.58	0.52	0.43
Wrong matches (all rings/event)	12.32	5.45	4.20	2.22	1.63	1.00	0.58
Wrong matches (electron rings/event)	0.73	0.45	0.41	0.30	0.26	0.20	0.14

With a cut allowing for 1cm distance at maximum, a total electron identification efficiency of about 74% can be reached in the RICH detector having in addition on the order of 0.5 fake rings and 0.5 clones rings per event, and about 0.3 wrong matches.

6. SUMMARY AND OUTLOOK

1. Two efficient algorithms for the CBM RICH data processing were developed: a standalone ring finder (using only RICH information) and an algorithm based on the information from vertex tracks.

2. After a study of the reasons of fake ring appearance and their features, a set of criteria for their rejection was proposed. On the basis of this set two approaches were studied for a reliable rejection of fake rings: one based on 2D cuts and the other on neural network.

3. All developed algorithms were tested on large statistics of simulated events and were then included into the CBM framework for common use.

The algorithms can be used to test the RICH performance if implementing different photodetectors, e. g., the proposal from IHEP (Protvino) or a MAPMT from Hamamatsu. In particular, they allowed one to extract the RICH performance in its current layout in terms of pion suppression and electron identification. The routines can be further used for optimization of the detector setup. Also, they were used to establish electron identification routines which were applied for a rather realistic simulation of the di-electron spectra including full event reconstruction. The feasibility of a measurement of both the low-mass vector meson and the J/ψ could be shown.

REFERENCES

1. *CBM Collaboration.* Compressed Baryonic Matter Experiment. Technical Status Report, GSI, Darmstadt, 2006.

2. *Ososkov G. A., Polanski A., Puzynin I. V.* Modern Methods of Experimental Data Processing in High Energy Physics // Part. Nucl. 2002. V. 33. Issue. 3. P. 676–745.

3. *Hough P. V. C.* Method and Means for Recognizing Complex Patterns, U.S. Patent 3,069,654, 1962.

4. *Toft P.* The Radon Transform. Theory and Implementation: PhD Thesis, Department of Mathematical Modelling, Section for Digital Signal Processing, Technical University of Denmark, 1996; http://www.ei.dtu.dk/staff/ptoft/ptoft.html

Received on June 14, 2007.

Редактор *Е. И. Кравченко*

Подписано в печать 02.10.2007.
Формат 60 × 90/16. Бумага офсетная. Печать офсетная.
Усл. печ. л. 1,5. Уч.-изд. л. 2,01. Тираж 290 экз. Заказ № 55908.

Издательский отдел Объединенного института ядерных исследований
141980, г. Дубна, Московская обл., ул. Жолио-Кюри, 6.
E-mail: publish@jinr.ru
www.jinr.ru/publish/