E10-2006-48

P. G. Akishin[1], E. P. Akishina[1], S. A. Baginyan[1],
V. V. Ivanov[1], Val. V. Ivanov[1], I. V. Kisel[1,2],
B. F. Kostenko[1], E. I. Litvinenko[3], G. A. Ososkov[1],
A. M. Raportirenko[1], A. G. Soloviev[1], P. V. Zrelov[1],
V. V. Uzhinsky[1]

METHODS FOR EVENT RECONSTRUCTION
IN THE CBM EXPERIMENT

[1] Laboratory of Information Technologies, Joint Institute for
Nuclear Research, 141980 Dubna, Russia
[2] Kirchhoff Institute of Physics, University of Heidelberg, 69120 Heidelberg,
Germany
[3] Frank Laboratory of Neutron Physics, Joint Institute for Nuclear Research,
141980 Dubna, Russia

Акишин П. Г. и др.   E10-2006-48
Методы восстановления событий в эксперименте CBM

   Обсуждаются методы и алгоритмы восстановления событий в эксперименте
CBM. Они разработаны и представлены группой ЛИТ на рабочих совещаниях
коллаборации CBM в 2004–2005 гг.

   Работа выполнена в Лаборатории информационных технологий ОИЯИ.

Akishin P. G. et al.   E10-2006-48
Methods for Event Reconstruction in the CBM Experiment

   Methods and algorithms for event reconstruction in the CBM experiment are
described. All these results have been developed and presented by the LIT team at
the CBM collaboration meetings in 2004–2005.

   The investigation has been performed at the Laboratory of Information Tech-
nologies, JINR.

INTRODUCTION

The CBM Collaboration [1, 2] builds a dedicated heavy-ion experiment to investigate the properties of highly compressed baryonic matter as it is produced in nucleus–nucleus collisions at the Facility for Antiproton and Ion Research (FAIR) in Darmstadt, Germany. The scientific goal of the research program of the CBM experiment is to explore the phase diagram of strongly interacting matter in the region of highest baryon densities. This approach is complementary to the activities at RHIC (Brookhaven) and ALICE (CERN–LHC) which concentrate on the region of high temperatures and very low net baryon densities.

The experimental setup has to fulfil the following requirements: identification of electrons which requires a pion suppression factor of the order of 10^5, identification of hadrons with large acceptance, determination of the primary and secondary vertices (accuracy $\sim 30\,\mu$m), high granularity of the detectors, fast detector response and read-out, very small detector dead time, high-speed trigger and data acquisition, radiation hard detectors and electronics, tolerance towards delta-electrons.

Figure 1 depicts the present layout of the CBM experimental setup. Inside the dipole magnet gap there are the target and a 7-planes Silicon Track-
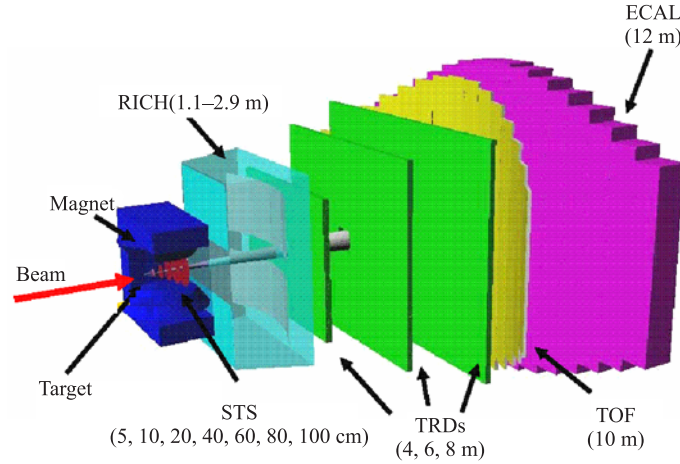

Fig. 1. CBM general layout

ing System (STS) consisting of pixel and strip detectors. The Ring Imaging Cherenkov detector (RICH) has to detect electrons. The Transition Radiation Detector (TRD) arrays measure electrons with momentum above 1 GeV. The

Time-of-Flight (TOF) detector consists of Resistive Plate Chambers (RPC). The Electromagnetic Calorimeter (ECAL) measures electrons, photons and muons. The CBM setup is optimized for heavy-ion collisions in the beam energy range from about 8 to 45 A · GeV. The typical central Au+Au collision in the CBM experiment will produce up to 700 tracks in the inner tracker (see Fig. 2). Large track multiplicities together with the presence of a non-homogeneous magnetic field make the reconstruction of events extremely complicated. It comprises local track finding and fitting in the STS and TRD, ring finding in RICH, cluster reconstruction in ECAL, global matching between STS, RICH, TRD, TOF and ECAL, and the reconstruction of primary and secondary vertices. Therefore, the collaboration performs the extensive analysis of different event recognition and reconstruction methods, in order to better understand the geometry of detectors and to investigate specific features of useful events [2].
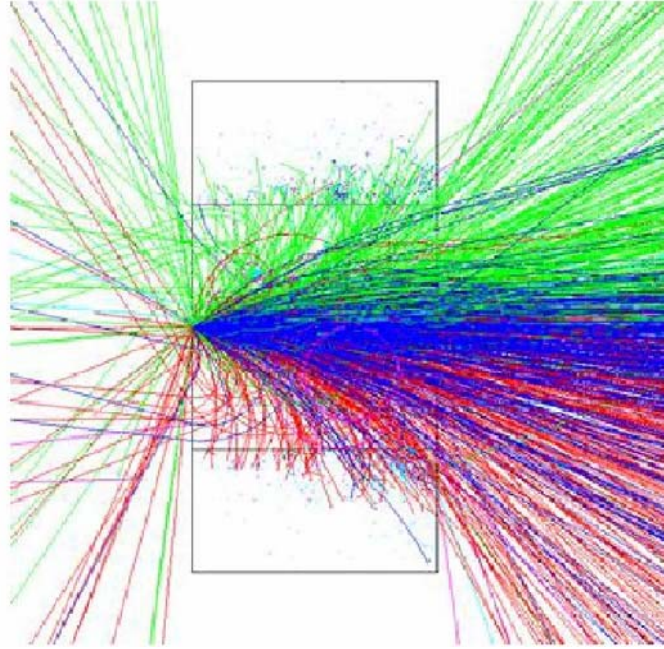

Fig. 2. Visualization of a typical CBM event

1. THE STS TRACK RECONSTRUCTION PROBLEM

The STS track reconstruction problem can be split into track finding and track fitting problems. Different competitive approaches to both track finding and the

reconstruction of the initial track parameters were applied by the LIT specialists. For the track finding 3D track-following and cellular automaton methods have been used. The Kalman filter and global fitting methods like the polynomial approximation are applied to the problem of momentum reconstruction. The Kalman filter was also used for determination of primary and secondary vertices.

1.1. 3D Track-Following Method for STS. The track-following method reconstructs tracks based on the hits measured in the STS tracking stations. The algorithm should be stable with respect to initial vertex coordinates and the STS geometry. We used some approaches known from [3]. The track recognition procedure is accomplished in 3D space on both x–z and y–z projections simultaneously. The procedure alternates between both views, predicting a track position on the next station and searching for hits in the vicinity of the predicted position. Starting from the middle of the target area, this point is sequentially connected with all hits in the first station in y–z view, where tracks are close to straight lines (see Fig. 3). The straight lines driven via these two points are prolonged to the plane of the second station. All hits in an asymmetrical corridor around the intersection point are then used for fitting by a parabola in the x–z view which is prolonged to the next station. Since several prolongations can happen, we set aside corridors around each point predicted on the third station. A similar corridor is set in the y–z view on the third station. If hits are found in these limits, they are attached to the track. We consider the hits belonging to a track, if they are found in both planes. If for some of the track prolongations, we do not find such a correspondence, we delete that prolongation.

The method continues the track prolongation and searching for hits in corridors around the predicted position in both projections simultaneously towards the outer stations. Having tracks found on the first stage we make a selection on the basis of the following criterion: each point can belong to one track only. So, any track with one or more points belonging to some other track is rejected.

Each new parabolic prolongation is done with the corresponding curvature radius r calculated *taking the magnetic field into account* (see Figs. 3 and 4). It is done in the following way. Knowing parameter a from the initial parabola $x = az^2 + bz + c$, we derive the momentum value P_{xz} for the second station as

$$P_{xz} = \frac{0.3 B_y (1 + (2a \cdot z + b)^2)^{3/2}}{2a}. \tag{1}$$

From P_{xz} and the Y-component B_y of the magnetic field known from its map we obtain the approximation of the curvature radius $r = P_{xz}/0.3 \cdot B_y$ at the point (x_2, z_2). That is enough to draw a new parabola to make a prediction on the third station. Further prolongations are made in the same manner.

The high accuracy predictor is based on *special tables with confidence bounds of prediction corridors* depending on the station number. Those tables were calculated by statistics obtained from many thousands simulated events. Calculations

3

have to be done by Monte-Carlo simulations of a considered sample of heavy-ion events in the STS for all stations. Since the conventional approach to use symmetrical 3σ corridors appeared to be too inefficient, we use the distributions of deviations between real hit positions and their predictions for all remaining layers.
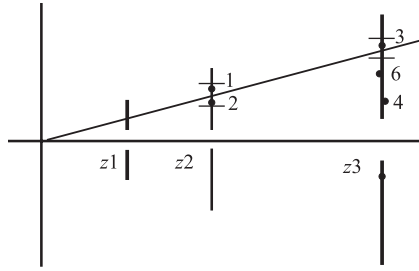

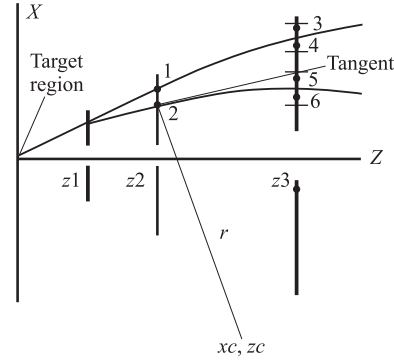Fig. 3. Prediction and search in YOZ view

Fig. 4. Prediction and search in XOZ view taking into account the local track curvature

Magnetic field knowledge gave us a striking result: the distributions and corresponding confidential corridors for prediction got narrowed 6 times. That increases substantially the accuracy of prediction and speed up the track-finding performance.

On the data simulated for the initial STS design based on hybrid pixel stations (see [1]), 3D track-following approach has shown quite a satisfactory efficiency at the level 92–96 % and very low level of ghost tracks. At the same time, for the most recent STS design with three first stations constructed on the MAPS technology and four other stations based on silicon strip wafers the efficiency felt down demanding a more elaborated predictor.

Much better results on the MAPS and strip STS design were obtained by the second track reconstruction procedure based on combination of a cellular automaton application for track-element recognitions and Kalman filter as a predictor.

1.2. Cellular Automaton Based Track Finding. The cellular automaton method [4, 5] creates short track segments (tracklets) in neighbouring detector planes and strings them into tracks (see Fig. 5). Being essentially local and parallel, cellular automata avoid exhaustive combinatorial searches, even when implemented on conventional computers. Since cellular automata operate with highly structured information, the amount of data to be processed in the course

of the track search is significantly reduced. As a rule, cellular automata employ a very simple track model which leads to utmost computational simplicity and fast algorithm. By definition, the reconstructed track is assigned to a generated particle, if at least 70% of its hits have been caused by this particle. A generated particle is regarded as found, if it has been assigned to at least one reconstructed track. If the particle is found more than once, all additionally reconstructed tracks are regarded as clones. The reconstructed track is called a ghost, if it is not assigned to any generated particle (70% criteria). The efficiency of track reconstruction for particles detected in at least four stations is presented in Fig. 6. Tracks of high momentum particles are reconstructed very well with efficiencies of 99.45%, while multiple scattering in detector material leads to a lower reconstruction efficiency of 89.46% for slow particles. The reconstruction efficiency for fast primary tracks with momentum higher than 1 GeV/c is almost 100%, while the efficiency of all fast tracks is slightly lower because of the presence of secondary tracks, originating far downstream from the target region. The total efficiency for all tracks with a large fraction of soft secondary tracks is 96.98%. The clone rate is not a problem for the algorithm (0.01%), the ghost level is at 0.61%.
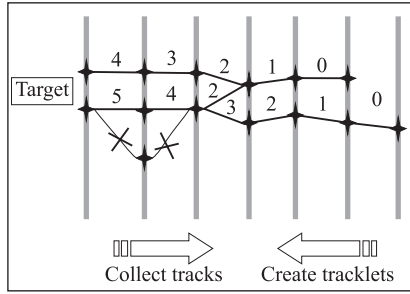

Fig. 5. A simple illustration of the cellular automaton algorithm. It creates tracklets, links and numbers them as possibly situated on the same trajectory, and collects tracklets into track candidates
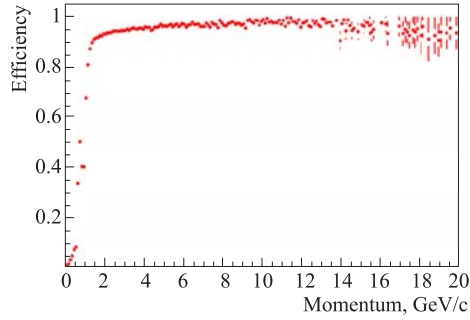

Fig. 6. Track reconstruction efficiency as a function of momentum

2. TRACK AND VERTEX FITTING

Track and vertex fitting have been done using the Kalman filter based procedures [2, 7]. Propagation of tracks in non-homogeneous magnetic field is based on a specially developed analytic formula [8, 9]. Mean relative momentum resolution for all tracks is 0.69%. Secondary tracks from D^0 decay being longer

5

have a slightly better momentum resolution of 0.67%. After the primary vertex is reconstructed, tracks identified as primary can be refitted with an additional constraint to the primary vertex position. This improves their average track momentum resolution to 0.63%.

The *primary vertex* was determined from all tracks reconstructed in the STS excluding those which formed well detached vertices like K_S^0 and Λ decays. The Kalman filter based algorithm reconstructs the primary vertex with the accuracy of 4 μm for the longitudinal and better than 1 μm for transversal components of the primary vertex position.

Precision of the *secondary vertex* parameters obtained in the geometrical vertex fit can be improved by taking into account several assumptions on tracks associated to the vertex. Two types of constraints have been included into the secondary vertex fit: a mass constraint and a topological constraint. The mass constraint is usually applied in the case of one or several combinations of particles in the vertex are known to originate from a narrow width mass state. The topological constraint is used to point a mother particle to the (already reconstructed) primary vertex. The final accuracy is 44.4 μm for the longitudinal and 1.7 μm for transversal components of the secondary vertex position for D^0 decay.

3. MOMENTUM ESTIMATION

Momentum estimation has been realized by two approaches: polynomial approximations and orthogonal polynomial sets.

3.1. Polynomial Approximations. This algorithm reconstructs the particle momentum directly from the hits in the Silicon Tracking System (STS). It consists of two steps. First, the track curve is fitted by a polynomial vector function, using the smoothness of the trajectory. Three types of approximation were applied: polynomials, cubic splines and B-splines. The optimization problem is described by the residual function

$$F = \sum_{i=0}^{N} \left[\left(\frac{\hat{x}(z_i) - x_i}{\sigma_x^i} \right)^2 + \left(\frac{\hat{y}(z_i) - y_i}{\sigma_y^i} \right)^2 \right],$$

where x_i, y_i are the trajectory hits, $\hat{x}(z)$, $\hat{y}(z)$ the coordinate approximations, σ_x^i, σ_y^i are the measurement errors, and N – the number of hits in the tracking system. It should be noted that F is a quadratic functional of the parameters describing the coordinate functions $\hat{x}(z)$, $\hat{y}(z)$. This means that the optimization problem is reduced to a very fast procedure of multiplication of an a priory prepared matrix with the vectors of hit coordinates.

In the second step, the constructed functions $\hat{x}(z)$, $\hat{y}(z)$ are used to determine the approximate value of the momentum. The momentum reconstruction is based

on the equations of motion:

$$x_{zz} = \frac{kq(1+x_z^2+y_z^2)^{\frac{1}{2}}}{p}\{x_zy_zB_x - (1+x_z^2)B_y + y_zB_z\},$$

$$y_{zz} = \frac{kq(1+x_z^2+y_z^2)^{\frac{1}{2}}}{p}\{(1+y_z^2)B_x - x_zy_zB_y - x_zB_z\},$$

where p and q are momentum and charge of the particle, respectively, and B_x, B_y, B_z the components of the nonuniform magnetic field in the point (x, y, z); x_z, y_z and x_{zz}, y_{zz} denote the first and second derivatives of x and y with respect to z, respectively. With the density function

$$f(\alpha, z, \hat{x}(z), \hat{y}(z)) = |x_{zz} - \alpha kq(1+x_z^2+y_z^2)\{x_zy_zB_x - (1+x_z^2)B_y + y_zB_z\}|^2 +$$

$$+|y_{zz} - \alpha kq(1+x_z^2+y_z^2)^{\frac{1}{2}}\{(1+y_z^2)B_x - x_zy_zB_y - x_zB_z\}|^2$$

the approximated value p is derived as the inverse of minimizing functional

$$G(\alpha) = \int\limits_{z_b}^{z_e} f(\alpha, z, \hat{x}(z), \hat{y}(z))g(z)dz,$$

where $g(z)$ is a weight function and z_b, z_e are the z coordinates of the first and last STS detector, respectively. Since $G(\alpha)$ is a quadratic functional of the parameter α, a fast noniterative procedure for the evaluation of p can be constructed.

This algorithm has been applied to simulated tracks in the momentum range 1–10 GeV/c with hits in all seven stations of the STS. Ideal track finding was assumed. The first and second momenta of the relative momentum residual distributions are summarized in Table 1. While in the absence of multiple scattering (MS), the spline approximations give better results, the performance in the presence of multiple scattering is similar for all three approximations ($\sigma_p = 0.75$–0.80 %).

Table 1. Mean and RMS of the momentum residual distributions

Track model	$\langle\frac{\Delta p}{p}\rangle$ [%] no MS	$\langle\frac{\sigma_p}{p}\rangle$ [%] no MS	$\langle\frac{\Delta p}{p}\rangle$ [%] MS	$\langle\frac{\sigma_p}{p}\rangle$ [%] MS
Polynomial	–0.02	0.28	–0.02	0.76
Cubic spline	0.08	0.17	0.09	0.79
B spline	0.01	0.16	0.00	0.78

3.2. Orthogonal Polynomial Sets. The method of accurate momentum reconstruction with orthogonal polynomial sets constructs an explicit function which gives the momentum in terms of measurable quantities: position and direction of a track at the entrance of the spectrometric field and the deflection angle as the effect of the field onto the track momentum [10–12]. This experimental input information can be provided, e. g., by a Kalman filter operating on hits registered in the CBM silicon tracking system (STS).

In inhomogeneous magnetic field φ is a function of p, position and direction of a charged particle at the magnet entrance

$$\varphi = \varphi(X_1, Y_1, A_x, A_y, p), \tag{2}$$

(X_1, Y_1, Z_1) are the points in the first STS, (A_x, A_y) are the tangents of the particle trajectory in this point. The task is to construct the inverse function

$$p = p(X_1, Y_1, A_x, A_y, \varphi), \tag{3}$$

which provides accurate momenta restoration [12].

The procedure consists of two steps:

1) The deflection angles for the given magnetic field are calculated for a set of representative trajectories.

2) The explicit function (3) is then constructed on the basis of these trajectories.

Each trajectory is defined by five variables:

1) $(x_1 = X_1, y_1 = Y_1)$ — coordinates of a point in the first STS;

2) $(x_3 = A_x, x_4 = A_y)$ — tangents of a particle trajectory in the points (X_1, Y_1);

3) $x_5 = 1/(p \cdot c)$, p is the particle momentum, c — speed of light.

Let $[A_i, B_i]$ be the range of ith variable: $i = 1, 2, \ldots, 5$. Each variable is normalized to the range $[-1, +1]$

$$g_i = \frac{2x_i - A_i - B_i}{B_i - A_i} \tag{4}$$

and a discrete number of «*nodes*», according to the Tchebycheff distribution

$$g_i = g_i(\alpha_i) = \cos \frac{(2\alpha_i - 1)\pi}{2N_i}, \quad \alpha_i = 1, \ldots, N_i, \quad i = 1, \ldots, 5$$

is chosen. The set of $N_1, N_2, \ldots, N_5$ determines the collection of fixed trajectories, which are traced through the magnetic field, and the set of corresponding deflections $\varphi(x_1, x_2, x_3, x_4, x_5)$ is calculated.

Let the range of $\varphi(\cdot)$ be $[A_6, B_6]$; $\varphi(\cdot)$ is also normalized to the range $[-1, +1]$:

$$g_6 = \frac{2\varphi - A_6 - B_6}{B_6 - A_6}$$

8

and a discrete number of values g_6

$$g_6 = g(\alpha_6) = \cos\frac{(2\alpha_6 - 1)\pi}{2N_6}, \quad \alpha_6 = 1, \ldots, N_6, \quad N_6 \leqslant N_5$$

is chosen. Now, applying the inverse interpolation, we can calculate the corresponding values of g_5.

Let g_5 be in the form

$$g_5 = \sum_{ijklm} C_{ijklm} T_i(g_1) T_j(g_2) T_k(g_3) T_l(g_4) T_m(g_6), \tag{5}$$

$i = 0, \ldots, N_1 - 1; j = 0, \ldots, N_2 - 1; k = 0, \ldots, N_3 - 1; k = 0, \ldots, N_4 - 1;$
$m = 0, \ldots, N_6 - 1$. The coefficients C_{ijklm} are calculated using the formula

$$C_{ijklm} = \frac{\sum\limits_{\alpha_1\alpha_2\alpha_3\alpha_4\alpha_6} g_{5\alpha_1\alpha_2\alpha_3\alpha_4\alpha_6} T_i(g_1) T_j(g_2) T_k(g_3) T_l(g_4) T_m(g_6)}{(\sum\limits_{\alpha_1} T_i(g_1))^2 (\sum\limits_{\alpha_2} T_j(g_2))^2 (\sum\limits_{\alpha_3} T_k(g_3))^2 (\sum\limits_{\alpha_4} T_l(g_4))^2 (\sum\limits_{\alpha_6} T_m(g_6))^2}, \tag{6}$$

$\alpha_1 = 1, \ldots, N_1; \alpha_2 = 1, \ldots, N_2; \alpha_3 = 1, \ldots, N_3; \alpha_4 = 1, \ldots, N_4; \alpha_6 = 1, \ldots, N_6.$

The total number of «nodes» for which trajectories were calculated was 625: $N_1 = N_2 = N_3 = N_4 = 5$. The trajectories were computed for each sample and for $N_5 = 7$ momentum values in the range 1–10 GeV/c and the deflection angle was determined. For each combination $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$ the momentum variable was calculated by inverse interpolation for $N_6 = 7$ of deflection variables. Then, using (6), the expansion coefficients C_{ijklm} were calculated: total number is $5 \times 5 \times 5 \times 5 \times 7 = 4375$. Lowering the upper limits N_1, N_2, N_3, N_4, N_6, we obtain, without changing the coefficients, a least-squares fit to the computed trajectories. This is a consequence of the Tchebysheff polynomials being orthogonal. The number and significant coefficients can be found by a Fisher test. Our analysis has shown that without loss in accuracy, only 89 coefficients can be used.

In order to estimate the accuracy of the method on data close to real data, we used the GEANT data. Figure 7 presents the distribution of $p - p_c$ and Fig. 8 shows the distribution of $\dfrac{p - p_c}{p}$ for positively charged tracks. One can see from Fig. 8 that the dispersion of the distribution $\dfrac{p - p_c}{p}$ is 0.26%.

It must be noted that this result is obtained for positively charged particles, because the tracing of the basic set of trajectories was realized for positively charged particles. For a small part of tracks ($\approx 10\%$), the parameters of which are out of the range of variables x_1, x_2, x_3, x_4, x_6, we used the approximation of the uniform magnetic field. This reduces the overal resolution to about 0.34%.
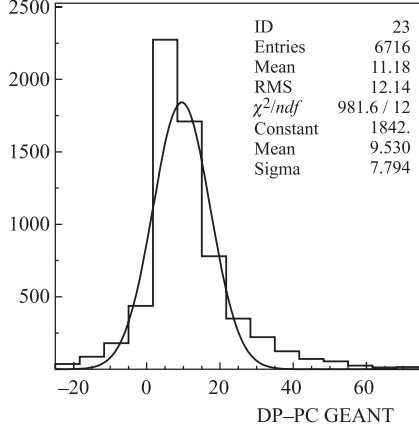
9

ID	23
Entries	6716
Mean	11.18
RMS	12.14
χ^2/ndf	981.6 / 12
Constant	1842.
Mean	9.530
Sigma	7.794

DP–PC GEANT

ID	24
Entries	6716
Mean	0.3918E–02
RMS	0.4754E–02
χ^2/ndf	727.8 / 17
Constant	1818.
Mean	0.4102E–02
Sigma	0.2628E–02

DPP–PC GEANT

Fig. 7. Distribution of $p-p_c$ (in MeV/c) for the GEANT data (for positively charged particles)
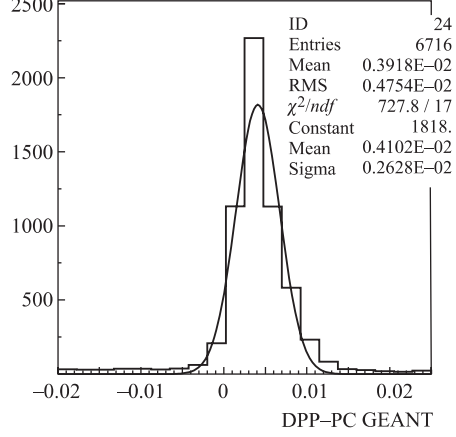
Fig. 8. Distribution of $\dfrac{p - p_c}{p}$ for the GEANT data (for positively charged particles)

In summary, the algorithm provides the possibility to reconstruct the momentum of charged particles registered in the STS system with high accuracy. The accuracy can be further improved by separate momentum reconstruction of particles of different charges and by subdivision of the momentum range into subintervals.

4. PARTICLE IDENTIFICATION WITH THE RICH DETECTOR

The Ring Imaging Cherenkov detector (RICH) is designed to provide electron identification in the momentum range of electrons from low-mass vector-meson decays. A second task of the RICH detector is the p identification for higher momenta in order to improve the K/p separation which quickly deteriorates for $p > 4$ GeV/c if only time-of-flight information is used. Particle identification with the RICH detector is performed by a measurement of the Cherenkov angle/ring radius and the momentum of the particles (see Fig. 10). Assuming that tracks with momentum are provided by the tracking system, the RICH part for particle identification requires the following steps:
- ring finding,
- determination of center and radius of ring/ Cherenkov angle,
- matching of rings with tracks.

All charged tracks being reconstructed by the tracking system are reflected at the mirror in order to give the center of a possible Cherenkov ring (see Fig. 9). The number of these tracks is much larger than the number of particles really

10

producing a ring. In order to combine rings and tracks each ring is to be matched to the track having its extrapolation closest to the calculated ring center.
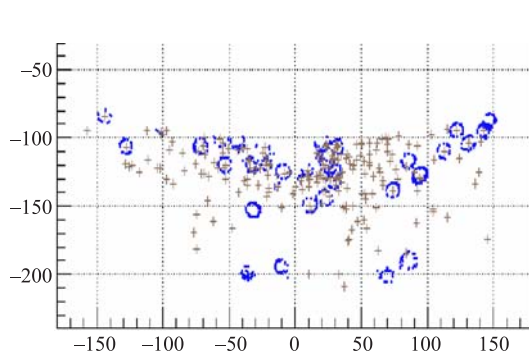

Fig. 9. Simulated image of the CBM RICH photode-tector plane. Hits forming rings to be recognized are marked by points. Track prolongations to this plane are marked by crosses
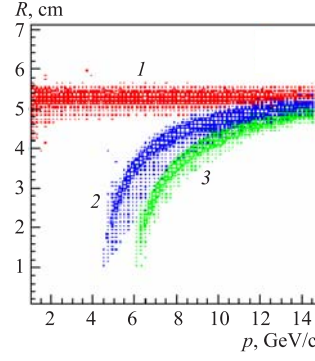
Fig. 10. Ring radius distribution versus the track momentum for $e^{\pm}$ (*1*), $\mu^{\pm}$ (*2*) and $\pi^{\pm}$ (*3*)

Track extrapolation was obtained using CBM Rich Projection Producer class. Each RICH event consists of about 400 track extrapolations and about 1500 hits, nearly 80 rings. So, the key part of any ring-finder is a prior search for all ring containing fragments. It was realized by the coarse histogramming of source data, then by clustering hits in all separate areas of this histogram and by choosing not all, but only hits belonging to each of those clusters.

Several methods for CBM-RICH ring recognition were elaborated on the basis of the fast search for the areas containing RICH hits and then, either by using the information of previously found tracks or, as a standalone program, for unguided ring finding.

4.1. Track Based RICH-Ring Recognition Algorithm. Each track extrapo-lation could be considered as a potential ring center. However, as can be seen from Fig. 9, the high multiplicity of CBM events, especially the great number of secondary particles cause the track-ring matching problem. It is solved by com-bining track and ring with closest distance. We calculate all distances between the ring center predicted by a track and nearest hits. Each time we test these distances to be within prescribed limits. Then, we histogram those distances and look for a maximum. If the sum of maximum bin and two adjacent bins is exceeded a CUT, that means that we found a ring. Natural criteria are also applied:

1) Δr — the difference of fitted radius and simulated radius < 0.5 cm;

2) Δxy — difference of fitted ring center and simulated center < 0.5 cm;

3) min/max potential hits (compare MC hits and associated hits) in the range 0.5–1.5.

11

The total efficiency of track-guided ring finding shown in Figs. 11 and 12 is on the satisfactory level.
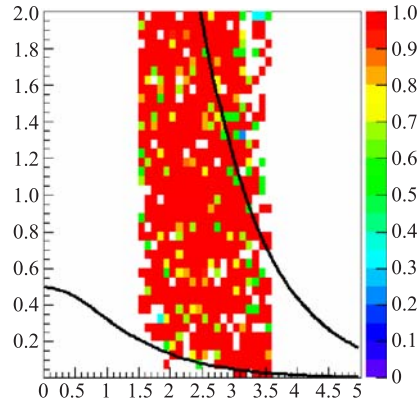

Fig. 11. Ring-finding efficiency for electrons as a function of P_t (GeV/c) (vertical axis) and rapidity
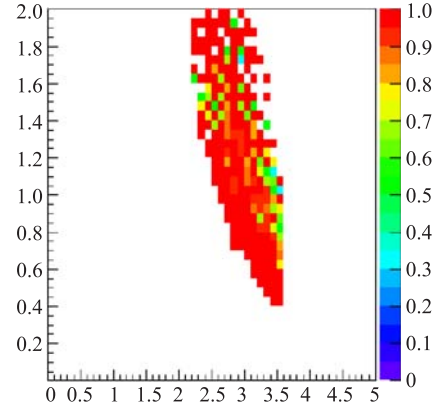

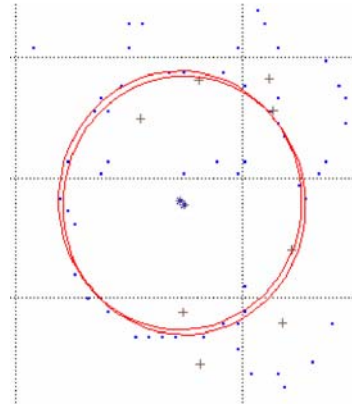Fig. 12. Ring-finding efficiency for pions as a function of P_t (GeV/c) (vertical axis) and rapidity
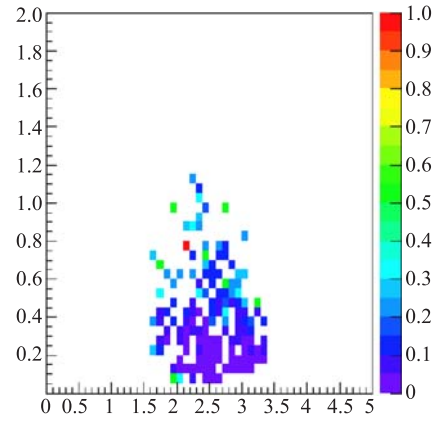

Fig. 13. Two tracks — one ring

Fig. 14. Wrong matches: real rings

Since we use the track extrapolation as a ring-finding predictor, we have the ring-finding and track-ring matching in one. But, if we have two very close

tracks, a mismatching problem arises (see Fig. 13). Fortunately, as could be seen in Figs. 14 and 15, mismatching errors are at the level less than 10%, although such circle problems need to be better studied.
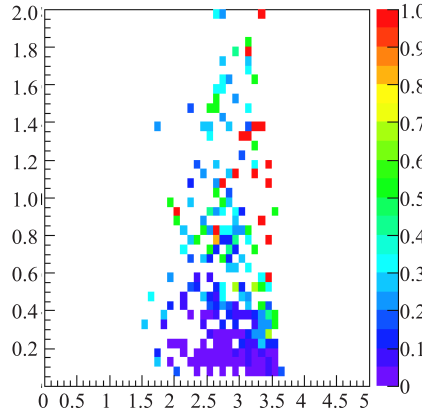

Fig. 15. Matching of fake rings

The second approach to the *unguided ring recognition*, leading to a so-called *standalone ring-finding*, is of importance for various problems of the RICH instrumental testing and for the RICH alignment with other detectors of the CBM setup. Two different methods have been considered: based on the Hough transform and elastic neural net approaches.

4.2. The Hough Transform Approach. It was studied as an option for a standalone ring finder for providing center and radius for each ring. It can be used as well to give an estimate of center and radius of the rings which then can be used as input for the ring fitting. The Hough transform framework is often applied to cope with a low resolution search of rings. The Hough transform is robust to a certain extent concerning topological gaps in rings (semicircles at detector edges) and concerning a high noise background [14, 15]. It converts points of the measurement space, i.e. hits, to points in the parameter space. In case of circles in the RICH detector, the coordinates of the parameter space are the ring centers and their radii. Through three arbitrary signal points a unique circle can be drawn. The resulting ring centers show a wide distribution in the parameter space and nearly fill the full circle of real rings. This effect can easily be understood considering the fact, that the ring center and the radius from the neighboring hits are only vaguely determined. Unfortunately, heavy combinatorics inherented to any Hough transform implementation results in its very high time consuming. Therefore, a procedure was developed to reduce the number of combinations. In

13

the next step of the usual Hough transform strategy, the ring centers and radii are determined by the search for the most populated places in the parameter space. A simple method would consist of collecting all calculated centers (xc, yc) in histograms of proper granularity and defining a cut-off value in the signal height to select real ring centers. The radii corresponding to each center can be then found similarly.

4.3. Elastic Neural Net for Standalone RICH Ring Finding. Standalone finding of rings in the RICH detector is based on the elastic neural net [17, 18]. The method does not require any prior track information and can be used for triggering. Application of the method to the RICH detector of the CBM experiment shows the efficiency of 94.3% and high speed (5.4 ms per event with about 1400 hits in the RICH detector). In view of its computational simplicity and high speed, the algorithm is considered to be further implemented in hardware which can increase the speed by few orders of magnitude.

4.4. The Ring Fitting. The ring fitting algorithm is implemented as the program RFit which estimates parameters of Cherenkov rings over a set of scattered points provided by the RICH detector.

The RFit algorithm is based on minimization of a least square function (LSF), which is a measure of deviation between a sought ring an a given data points (x_i, y_i):

$$\sigma^2 = \frac{\sum\limits_{i=1}^{n} d_i^2}{n}, \tag{7}$$

where $d_i = \left| \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2} - r \right|$, point (x_0, y_0) is the ring center, r is its radius.

RFit uses the MINUIT package [19] to perform a minimization. To be more exact, ROOT::TMinuit class is used[*]. This package was originally written in Fortran by Fred James and part of PACKLIB (patch D506) and has been converted to a C++ class by R. Brun.

This approach is often failed for highly contaminated data, thus, the robust fit must be used [20, 21, 22]. It can be realized by replacing LSF (7) by the weighted LSF:

$$\sigma^2 = \frac{\sum\limits_{i=1}^{n} w_i d_i^2}{\sum\limits_{i=1}^{n} w_i}, \tag{8}$$

---

[*]To use RFit both with ROOT and without it, TMinuit class is separated from the ROOT package by the corresponding modifications and is included to RFit distribution.

where weights w_i are recalculated before the minimization procedure. RFit uses well-known Tukey's bi-square weights:

$$w_i = \begin{cases} \left(1 - \dfrac{d_i^2}{c_T^2 \sigma^2}\right)^2, & d_i < c_T \sigma, \\ 0, & d_i \geqslant c_T \sigma. \end{cases} \quad (9)$$

As a whole, the robust approach can be realized as the iteration procedure:
• setting $w_i = 1$, calculating σ^2 by (8), setting $c_T = 5$;
• while $c_T > 1$ doing the loop:
— calculating w_i by (9) using current values of σ and c_T;
— minimizing σ^2 given by (8);
— reducing c_T by $c_T / = 2$.

This procedure can be speed-up significantly when some rough initial circle parameters are know from either a guidance or a search like the Hough transform. The robustness slows algorithm, but it is inevitable pay for repairing its accuracy and efficiency in highly contaminated cases. RFit supports both standard and robust approaches.

4.5. Simultaneous Fit of Two or More Circles. As one can see from Figs. 9–13, the high multiplicity of events leads often to overlapping close rings which hinders their parameters estimation. So, a problem of simultaneous fitting of two or more circles is arisen. To solve it, one needs to create the equation of such a combined curve. It can be done by multiplying the corresponding number of the circle equations. For instance, for the forth order curve joining two circles, it gives the six parameter equation

$$F(x_i, y_i; a, b, c, d, R_1, R_2) =$$
$$= [\sqrt{(x-a)^2 + (y-b)^2} - R_1][\sqrt{(x-c)^2 + (y-d)^2} - R_2] = 0. \quad (10)$$

The LSM estimation of all parameters requires the search for the global minimum of the nonlinear functional

$$L(a, b, c, d, R_1, R_2) = \sum_i w_i F^2(x_i, y_i; a, b, c, d, R_1, R_2), \quad (11)$$

with the optimal weight function

$$w_{\text{opt}}(t) = \frac{1+c}{1 + c \exp(t^2/2)}. \quad (12)$$

The only parameter c is the ratio of the mean number of noise observations within a strip of the width $\sigma\sqrt{2\pi}$ to the mean number of useful observations in the sample. Thus, it is determined by the contamination of data not in the whole

15

range of the sample, but within its essential part where all useful observations are practically concentrated. The value of c is often roughly known in experimental models.

The solution of nonlinear minimization of (11) was again obtained by the MINUIT program modified to work with robust weights.

Denoting by k the iteration number, one obtains from (12):

$$w_i^{(k+1)} = \frac{1+c}{1 + c\exp(\mathrm{e}_i^{(k)}/\mathrm{var}^{(k)})^2/2}, \tag{13}$$

where

$$(\mathrm{var}^{(k)})^2 = \sum w_i^{(k)}(\mathrm{e}_i^{(k)})^2 / \sum w_i^{(k)}.$$

All variants of ring-fitting programs were carefully tested on a big sample of simulated events and showed the satisfactory accuracy and efficiency.

5. THE TRD TRACK FINDING

5.1. The TRD Geometry and Resolution. The TRD will be located between RICH and TOF detectors (see Fig. 1). One TRD layer consists of one big chamber 4×4 m with a square hole in the center (see Fig. 16). Within the current geometry of the CBM experiment the TRD layers are grouped into 3 TRD stations (Fig. 17) with 3–4 layers each.
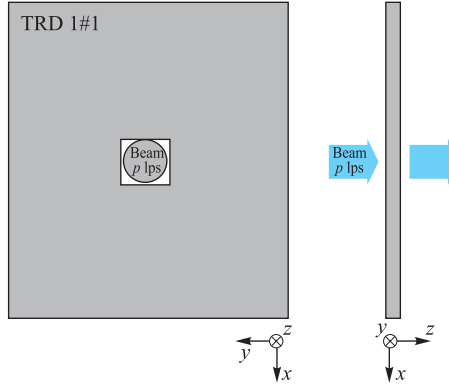

Fig. 16. Schematic layout of one TRD layer

The composition of the TRD chamber is identical for all chambers and is organized via layers of different materials with a total thickness of 6 cm. The TRD stations are placed at 4, 6 and 8 m from the target. Currently, there are
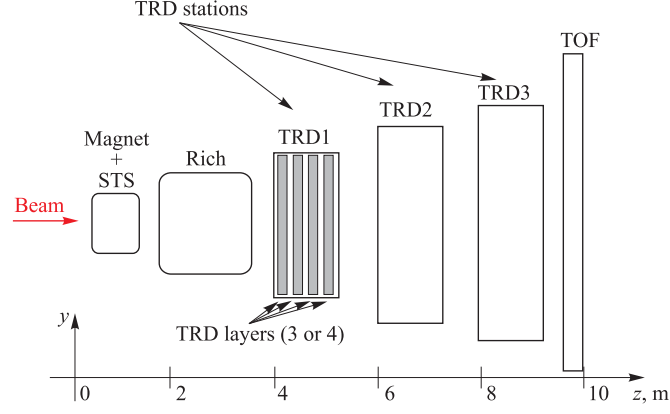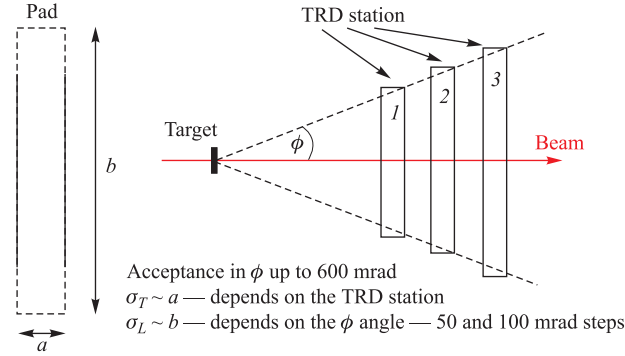
Fig. 17. CBM TRD scheme

Acceptance in ϕ up to 600 mrad
$\sigma_T \sim a$ — depends on the TRD station
$\sigma_L \sim b$ — depends on the ϕ angle — 50 and 100 mrad steps

Fig. 18. Width of pad and resolution across the pad (σ_T) are identical for all pads in each station. Length of pad and resolution along the pad (σ_L) are increasing with the distance from the beam position in discrete steps

two different geometry versions with 3 or 4 TRD layers per station. Each layer has a pad structure for read-out, where pads are rectangular and their orientation alternates between x and y directions from layer to layer (see Fig. 18).

5.2. Statistical Information. Some statistical information of the TRD event is presented on the histograms in Figs. 19, 20 and 21.

Most of the tracks have momenta larger than 1 GeV/c (Fig. 19). In order to eliminate tracks from secondary electrons, we accept only tracks with 12 hits at minimum.

The basic tracking problem is caused by the drastic difference in measurement errors on adjacent layers depending on the corresponding pad size. Their distributions in the (x, z) and (y, z) planes are shown in Figs. 20 and 21 for some
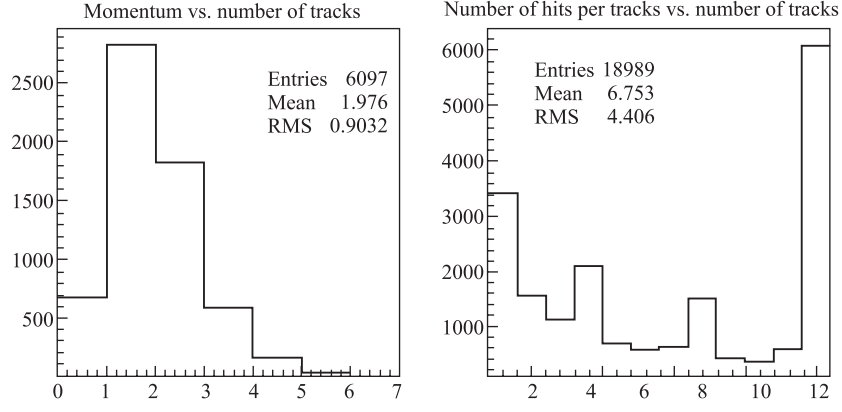
17

Fig. 19. Momentum distribution for tracks in the TRD (for 10 events) and distribution of number of the TRD hits per track (for 10 events)
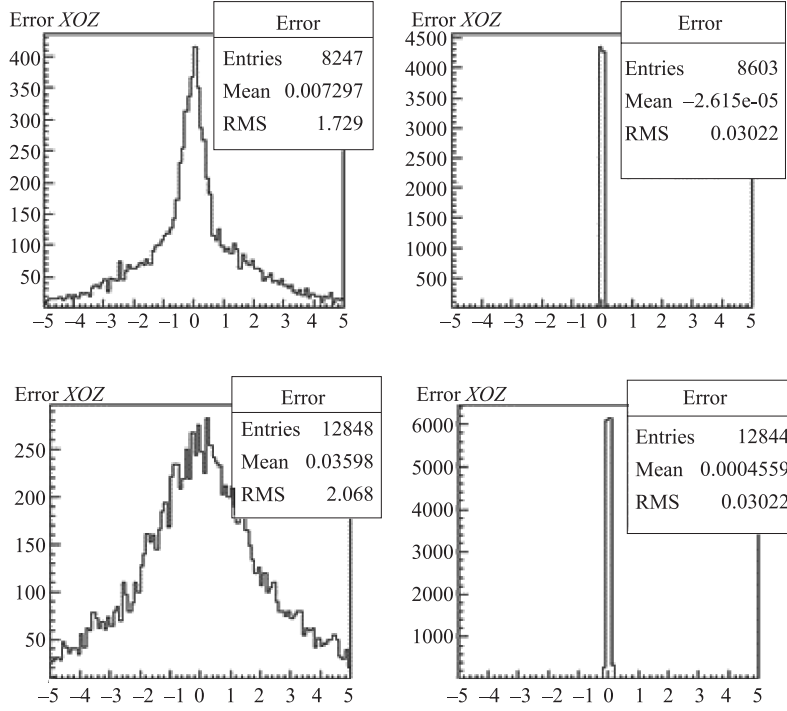

Fig. 20. Measurement errors in the (x, z) plane for the 1st, 2nd, 11th and 12th TRD layers
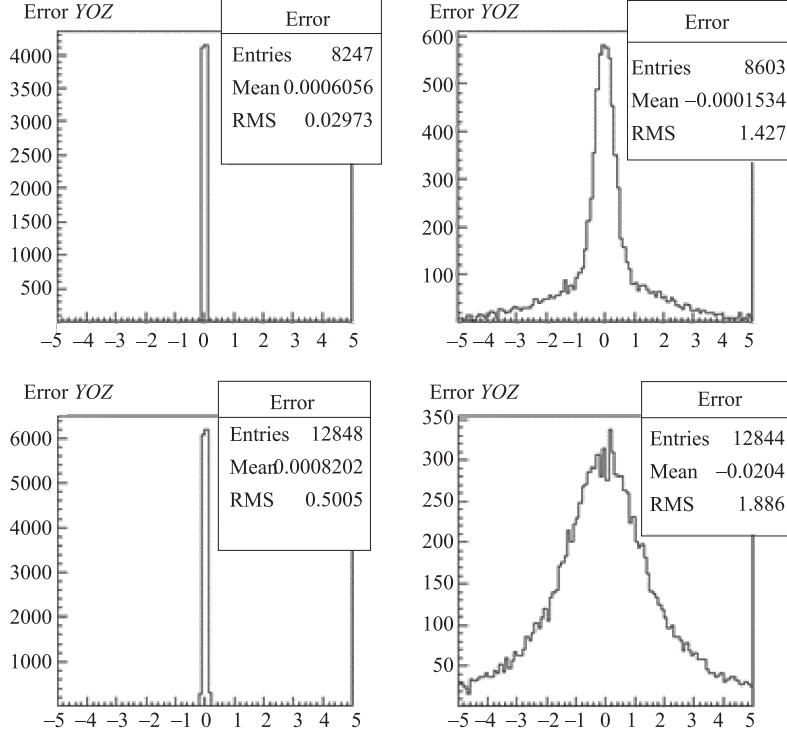
18

Fig. 21. Measurement errors in the (y, z) plane for the 1st, 2nd, 11th and 12th TRD layers

of TRD chambers.

5.3. 3D Track-Following Algorithm for CBM TRD. In order to draw first track-segments between first and second layers in (y, z) and (x, z) views as a preliminary version, we use Monte-Carlo values for the track and its momentum of the entrance of the first TRD station. Further, it can be replaced either by track and momentum prolongation from the STS to TRD, or, in a standalone variant, by the initial search of track segments connecting hits in the first and second stations.

The 3D track-following algorithm for TRD could repeat this one described in Subsec. 1.1 with the obvious simplification caused by the absence of the magnetic field in the TRD area. However, the TRD specifics brought essential tracking problems arising due to:

• errors in TRD layers;

• multiple scattering implies big track deviations from the predicted position (see Fig. 22);

19

• large multiplicity (increasing by secondary electrons from 800 tracks on the 1st station up to 1400 on the last one);
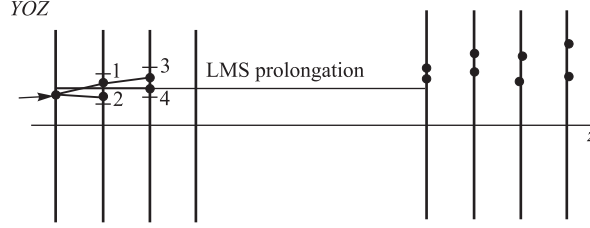
• noisy hits from secondary electrons.

Fig. 22. Track deviation from its predicted position

Therefore, very significant improvement of the track predictor is required. This has been achieved by applying the Kalman filter.

5.4. Kalman Filter Application in the TRD Track Finder. *5.4.1. Kalman Filter Formalism.* Kalman filter is usually applied to the analysis evolution of a linear discrete dynamical system. The system dynamics in a discrete point of time t_k is described by a linear stochastic equation

$$\vec{p_k} = F_k \overrightarrow{p_{k-1}} + \vec{\omega_k}, \tag{14}$$

where $\vec{p_k}$ is the state vector (the vector of the system parameters) of the analyzed system; F_k is the deterministic transition matrix from the state $k-1$ to the state k; $\vec{\omega_k}$ is a random disturbance.

The state vector does not need to be observed directly, but a linear function of the state vector is observed. It is a measurement vector

$$\vec{m_k} = H_k \vec{p_k} + \vec{\xi_k}, \tag{15}$$

where $\vec{\xi_k}$ – the measurement error.

The vectors $\vec{\omega_k}$ and $\vec{\xi_k}$ are Gaussian-independent random variables with Q_k and V_k covariance matrices.

The formula of the one-step Kalman predictor has the following form:

$$\widetilde{\vec{p_k}} = \widetilde{\vec{p_k^{k-1}}} + K_k(\vec{m_k} - H_k \widetilde{\vec{p_k^{k-1}}}), \tag{16}$$

where

• $\widetilde{\vec{p_k}}$ is the estimate of the state vector at the kth step. It represents the mathematical expectation of the state vector $\vec{p_k}$, based on known k measurement vectors: $E(\vec{p_k} \mid \vec{m_k}, \overrightarrow{m_{k-1}}, \dots, \vec{m_1})$.

- $\widetilde{p_k^{k-1}}$ is the predicted value of the state vector at the kth step using the state vector estimated at the previous $k-1$th step: $\widetilde{p_k^{k-1}} = F_k \widetilde{p_{k-1}}$.

K_k is the Kalman transition matrix which transforms the error in the measurement space to the space of the system parameters

$$K_k = C_k^{k-1} H_k^T (H_k C_k^{k-1} H_k^T + V_k)^{-1}, \qquad (17)$$

where

- $C_k^{k-1} = \mathrm{cov}(\widetilde{p_k^{k-1}} - \vec{p_k})$ is the covariance matrix of the predicted state vector

$$C_k^{k-1} = F_k C_{k-1} F_k^T + Q_k. \qquad (18)$$

- $C_k = \mathrm{cov}(\widetilde{p_k} - \vec{p_k})$ is the covariance matrix of the state vector estimated at the kth step

$$C_k = (I - K_k H_k) C_k^{k-1}.$$

5.4.2. Application to the TRD Track Finding. We assume that in the TRD zone the trajectories of charged particles are piecewise lines consisting of fragments of different straight lines. Each fragment in XOZ and YOZ planes is described by two equations

$$x = x_k + t_x(z - z_k),$$

$$y = y_k + t_y(z - z_k),$$

where x_k, y_k are the coordinates of the particle intersection with the kth layer and t_x, t_y are the track slopes formed by multiple scattering in the material between layers. So, we consider the particle trajectory as a linear discrete dynamic system which changes its parameters when it passes each layer.

In order to introduce the Kalman filter related to our problem, we determine the state vector as

$$\vec{p_k} = (x_k, y_k, t_x, t_y, q/p)^T.$$

Then, the corresponding transition matrix F_k is

$$F_k = \begin{pmatrix} 1 & 0 & \Delta z & 0 & 0 \\ 0 & 1 & 0 & \Delta z & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \qquad (19)$$

where $\Delta z = z_k - z_{k-1}$.

The matrix H_k in our case is

$$H_k = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}. \tag{20}$$

The covariance matrix V_k is

$$V_k = \begin{pmatrix} v_{11} & 0 & 0 & 0 & 0 \\ 0 & v_{22} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{21}$$

where v_{11}, v_{22} are the measurement errors of x, y track coordinates in the kth layer.

The covariance matrix Q_k coincides with the multiple scattering matrix. In our case, for thin scattering layer, Q_k has the form

$$Q_k = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & q_{33} & q_{34} & 0 \\ 0 & 0 & q_{43} & q_{44} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{22}$$

where

$$q_{33} = (1 + t_x^2)(1 + t_x^2 t_y^2)\theta_0^2,$$

$$q_{34} = q_{43} = (1 + t_y^2)(1 + t_x^2 t_y^2)\theta_0^2,$$

$$q_{44} = t_x t_y (1 + t_x^2 t_y^2)\theta_0^2$$

and

$$\theta_0 = \frac{13.6 \text{ MeV}}{\beta cp} z \sqrt{\frac{x}{X_0}}[1 + 0.038 \ln \frac{x}{X_0}].$$

The TRD Track Finder algorithm is 3D track-following process where the Kalman filter step by step gives, using formula (8), the estimate of the track parameters on each layer. On the last layer, we obtain the best estimate of the track parameters. Then, using the *smoothing* procedure we correct the estimate of the track parameters on each previous layer.

5.4.3. The Smoother Procedure. The smoothed state vector is introduced as

$$\widetilde{\overrightarrow{p_k^n}} = \widetilde{\overrightarrow{p_k}} + A_k(\widetilde{\overrightarrow{p_{k+1}^n}} - \widetilde{\overrightarrow{p_{k+1}^k}}),$$

where $\widetilde{\overrightarrow{p_n^n}} = \widetilde{\overrightarrow{p_n}}$.

Smoother gain matrix is

$$A_k = C_k F_{k+1}^T (C_{k+1}^k)^{-1}.$$

Covariance matrix of the smoothed state vector is

$$C_k^n = C_k + A_k(C_{k+1}^n - C_{k+1}^k)A_k^T.$$

5.4.4. Preliminary Results. In Fig. 23 one can see the scheme of the TRD track finder. During our calculation of the track finding efficiency we use rather rigid track accepting criterion: a track is considered as accepted, if it contains 12 hits (each hit from each TRD station) and there are no hits from other tracks. As shown in Fig. 23, the efficiency at the level of 70% is given.
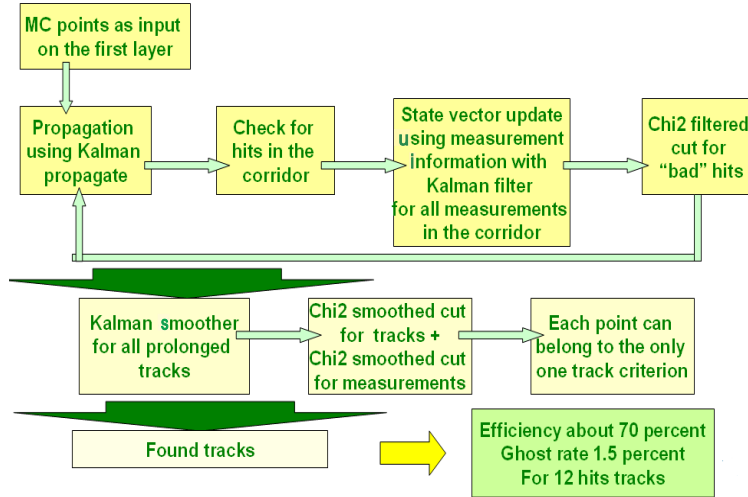

Fig. 23. Scheme of the TRD track finder

The performance of the new track finder program based on the Kalman filter, as a predictor, was tested by comparing the found track states with true Monte Carlo information. To characterize this comparison, we use the χ^2 on the number of freedom degrees (see Fig. 24). We consider also such a common estimation of the found track quality as $pullx = (xrec - xtrue)/\sigma_x$, where $xrec$ is the reconstructed value, and $xtrue$ is its true Monte Carlo value, σ_x is the estimated error, obtained from the covariance matrix of the Kalman procedure (see Figs. 25 and 26).
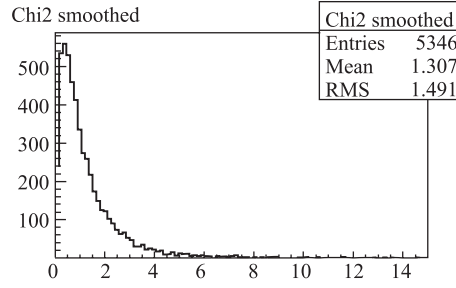
23

Chi2 smoothed

Chi2 smoothed	
Entries	5346
Mean	1.307
RMS	1.491

Fig. 24. χ^2 distribution for accepted tracks

Pull X

Pull X	
Entries	5346
Mean	0.01093
RMS	0.4767

Pull X

Pull X	
Entries	5346
Mean	0.008389
RMS	0.3884

Pull X

Pull X	
Entries	5346
Mean	0.004021
RMS	0.3559

Pull X

Pull X	
Entries	5346
Mean	0.00702
RMS	0.3921
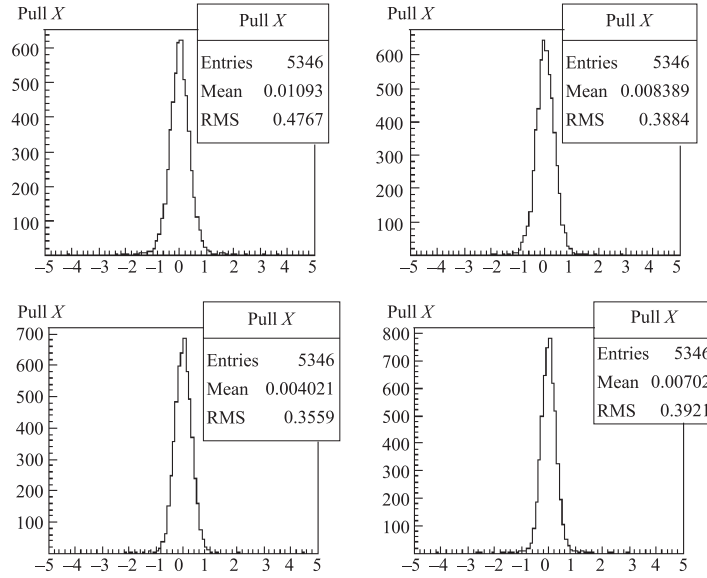
Fig. 25. Pull distribution of X

As one can see from distributions in Figs. 24, 25, 26, if we will apply more liberal accepting criteria (for instance, accepted track can have one common hit with the other tracks) and with better tuning of track finder algorithm parameters, it should give the increase of the total algorithm efficiency.
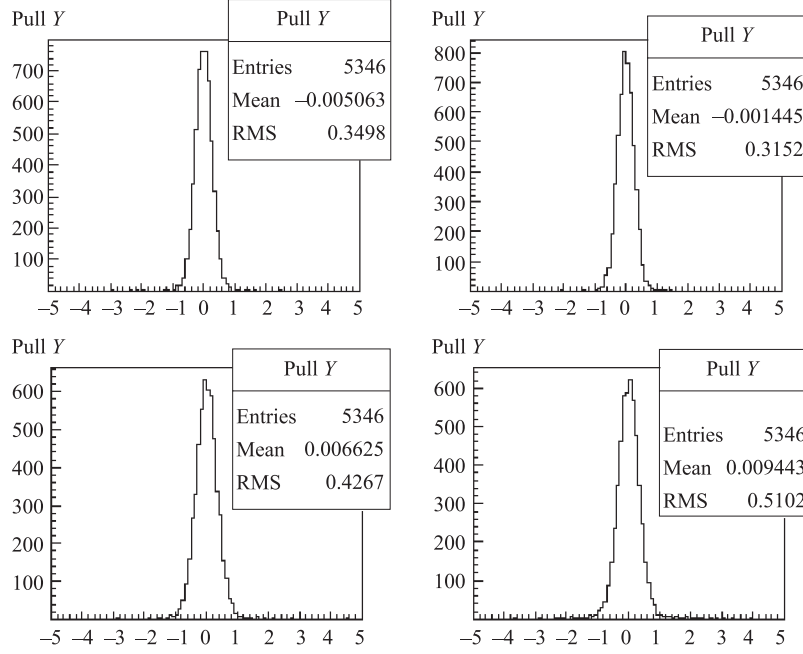
24

Fig. 26. Pull distribution of Y

CONCLUSION

During the years 2004–2005 the JINR LIT group proposed a set of effective methods for event reconstruction in the CBM experiment and developed corresponding algorithms implemented in the CBM framework software.

It includes the following:

• two approaches of track-finding algorithms (based on 3D-following and cellular automaton methods);

• primary and secondary vertex recognition algorithms;

• two different approaches for charged particle momentum restoration;

• Cherenkov ring finding (two approaches: Hough transform, elastic neural net) and robust ring fitting software;

• the TRD track finder with a track predictor based on the Kalman filter.

The majority of these codes is included into the CBM framework and is in use by physicists, although some of them need further development to improve the performance.

25

REFERENCES

1. Letter of Intent for the Compressed Baryonic Matter Experiment. http://www.gsi.de/documents/DOC-2004-Jan-116-2.pdf

2. Compressed Baryonic Matter Experiment. Technical Status Report. Darmstadt: GSI, 2005; http://www.gsi.de/onTEAM/dokumente/public/DOC-2005-Feb-447 e.html.

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

4. *Kisel I.* Fast Tracking in the CBM Experiment // 11th Workshop on Electronics for LHC and Future Experiments (LECC2005), 12–16 September, 2005, Heidelberg, Germany.

5. *Kisel I.* Tracking in the CBM Experiment. // The Workshop on Tracking in High Multiplicity Environments (TIME05), 3–7 October, 2005, Zurich, Switzerland (to appear in Nucl. Instr. Meth. A).

6. *Arnold R. et al.* (NEMO Collaboration). Technical design and performance of the NEMO 3 detector // Nucl. Instr. Meth. A. 2005. V. 536. P. 79–122.

7. *Gorbunov S., Kisel I., Vassiliev I.* Analysis of D0 meson detection in Au+Au collisions at 25 AGeV. CBM note CBM-PHYS-note-2005-001, 23 June 2005.

8. *Gorbunov S., Kisel I.* An analytic formula for track extrapolation in an inhomogeneous magnetic field. CBM note CBM-SOFT-note-2005-001, 18 March 2005.

9. *Gorbunov S., Kisel I.* Analytic formula for track extrapolation in non-homogeneous magnetic field // Proc. of the International Workshop on Advanced Computing and Analysis Techniques in Physics Research (ACAT 2005), May 22–27, 2005, DESY, Zeuthen, Germany (to appear in Nucl. Instr. Meth. A).

10. *Lechanoine C., Martin M., Wind H.* // Nucl. Inst. Meth. 1969. V. 69. P. 122–124.

11. *Azhgirey L. S. et al.* Measurements of the magnetic map of the spectrometric magnet SP12A. JINR Communication 13-86-164. Dubna, 1986 (in Russian).

12. *Bonushkina A. Yu., Ivanov V. V.* Momenta restoration of charged particles detected by the MASPIK spectrometer. JINR Communication P10-93-125. Dubna, 1993 (in Russian).

26

13. *Agakishiev H. et al.* New Robust Fitting Algoritm for Vertex Reconstruction in the CERES Experiment // Nucl. Instr. Meth. A. 1997. V. 394. P. 225–231.

14. *Muresan L., Muresan R., Ososkov G., Panebratsev Yu.* Deformable templates for circle recognition // JINR Rapid Comm. 1[81]-97. P. 27–44.

15. *Baginyan S., Ososkov G.* Finding tracks detected by a drift tube system // CPC. 1998. V. 108. P. 20–28.

16. *Ososkov G.* Novel approach in RICH data handling // Czech. J. Phys. 1999. V. 49/S2. P. 145–160.

17. *Gorbunov S., Kisel I.* Elastic net for standalone RICH ring finding. CBM note CBM-SOFT-note-2005-002, 22 September 2005.

18. *Gorbunov S., Kisel I.* Elastic net for standalone RICH ring finding. // Proc. of the X Intern. Workshop on Advanced Computing and Analysis Techniques in Physics Research (ACAT 2005), May 22–27, 2005, DESY, Zeuthen, Germany (to appear in Nucl. Instr. Meth. A).

19. http://wwwinfo.cern.ch/asdoc/minuit/minmain.html

20. *Huber P.* Robust statistics. N.-Y.: Wiley, 1981.

21. *Ososkov G.* Elastic arm methods of data analysis as a robust approach // Tatra Mt. Math. Publ. 2003. V. 26. P. 291–306.

22. *Chernov N., Kolganova E., Ososkov G.* Robust Methods for the RICH Ring Recognition and Particle Identification // Nucl. Instr. Meth. A. 1999. V. 433. P. 247–278.

Корректор *Т. Е. Попеко*