Письма в ЭЧАЯ. 2016. Т. 13, № 5(203). С. 1020–1027

КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ В ФИЗИКЕ

PERFORMING TRACK RECONSTRUCTION AT THE ALICE TPC USING A FAST HOUGH TRANSFORM METHOD

C. S. Kouzinopoulos [1], *P. Hristov* [2]

CERN, European Laboratory for Particle Physics, Geneva

The Hough Transform algorithm is a popular image analysis method that is widely used to perform global pattern recognition in images through the identification of local patterns in a suitably chosen parameter space. The algorithm can also be used to perform track reconstruction and to estimate the trajectory of individual particles passed through the active elements of a detector volume. This paper presents a fast reconstruction method for the Time Projection Chamber (TPC) of the ALICE experiment at LHC. The method, that combines a linear Hough Transform algorithm with a fast filling of the Hough Transform parameter space, is developed within AliceO^2, the new computing framework of ALICE for Run3.

PACS: 29.85.-c; 07.05.Pj

INTRODUCTION

ALICE (A Large Ion Collider Experiment) is a general-purpose, heavy-ion detector at the CERN Large Hadron Collider (LHC) which focuses on QCD, the strong-interaction sector of the Standard Model. It is designed to address the physics of strongly interacting matter and the quark–gluon plasma at extreme values of energy density and temperature in nucleus–nucleus collisions [1].

A significant increase of the LHC luminosity for heavy ions is expected in Run3 after the Long Shutdown 2 (LS2) period, leading to Pb–Pb collision rates of about 50 kHz and an integrated luminosity $L_{\rm int} = 10$ nb^{-1}. The main physics topics addressed by the ALICE upgrade are precise measurements of heavy flavour hadrons, low-momentum quarkonia, and low mass di-leptons [2]. After the upgrade, continuous read-out will be implemented instead of the classic trigger system for the main detectors, including ITS and TPC, to deal with event pile-up and avoid trigger-generated dead time.

The integration of online and offline data processing requires a common online–offline software framework, AliceO^2 and a common computing facility, the O^2 computing infrastructure, dedicated to both data collection and processing. AliceO^2, that is currently under development for Run3, is built on top of ALFA, the new ALICE–FAIR concurrency framework for high-quality parallel data processing and reconstruction on heterogeneous computing systems [3].

[1] E-mail: charalampos.kouzinopoulos@cern.ch

[2] E-mail: peter.hristov@cern.ch

Since approximately 92.5% of the data rate for Pb–Pb collisions at 50 kHz will be generated by the ALICE TPC, one of the main processing challenges of the O^2 computing infrastructure will be to achieve an efficient TPC track reconstruction. This paper presents one possible algorithm to perform track reconstruction for the TPC, a fast Hough Transform method. The method is based on prior work [4] where it showed good and stable track reconstruction performance over the entire range of expected event multiplicities in central Pb–Pb collisions; it is currently being implemented as part of AliceO^2.

1. ALICE TPC DETECTOR FOR Run3

TPC is the main tracking detector in the central barrel of ALICE. Together with the ITS, TRD and TOF detectors, it provides charged particle momentum measurement, particle identification, and vertex determination with sufficient momentum resolution, two-track separation and dE/dx resolution for studies of hadronic and leptonic signals in the region $p_t < 10$ GeV/c and pseudorapidities $|\eta| < 0.9$ [5, 6].

TPC is the largest detector of this type, with an overall active volume of about 90 m^3. It employs a cylindrical gas volume with an inner radius of approximately 84 cm, an outer radius of about 250 cm, and an overall length of about 5 m in the beam axis. It is divided into two half volumes of equal size, separated by a central electrode at high voltage to generate the drift field.

The readout plane is azimuthal segmented; the chambers are mounted into 18 trapezoidal sectors at each end plate, each sector covering 20° in azimuth. The readout at each sector is divided into two chambers, the Inner Readout Chamber (IROC) and Outer Readout Chamber (OROC), resulting in a radial segmentation of the readout plane, as can be seen in Fig. 1. On each side of the central electrode, the primary ionization electrons drift towards the end plate, where their signal is measured by approximately 550,000 readout pads (5504 pads per IROC and 9856 pads per OROC on each of the 36 TPC sectors).

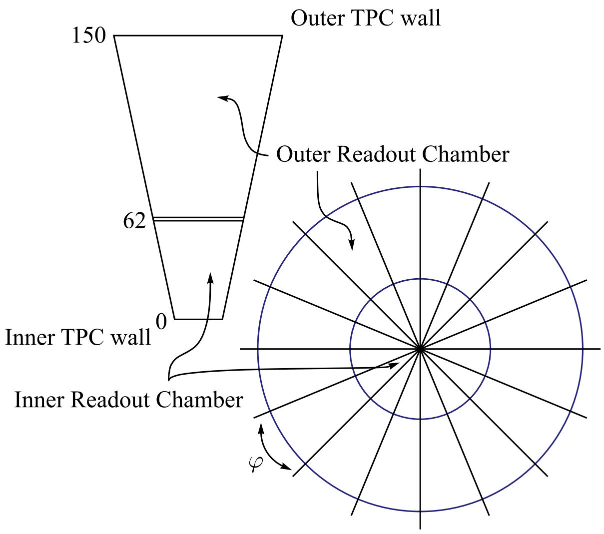

Fig. 1. A schematic view of the current layout of the TPC sectors and readout chambers

Currently, the signal measurement is performed via conventional Multi-Wire Proportional Chambers (MWPCs) with cathode pad readout, while the readout chambers operate with an active bipolar gating grid. This schema though has a principal rate limitation to about 3.5 kHz and thus is not suitable for Run3. For this reason, it will be replaced by a multistage Gas Electron Multiplier (GEM) detectors system as charge amplifier. GEMs operate reliably in high-rate applications and provide intrinsic ion blocking capabilities that avoid massive charge accumulation in the drift volume from back-drifting ions and prevent excessive space-charge distortions, enabling therefore the TPC to operate in a continuous, un-gated readout mode at collision rates of 50 kHz [6]. The upgraded readout chambers will employ stacks of four GEM foils for gas amplification and anode pad readout, with three GEM foils per layer in the OROCs and a single large GEM foil per layer for the IROCs.

The signals coming from the readout pads on the readout plane are passed to the front-end cards (FEC), digitized and processed by a DSP chip and are subsequently transferred to a Common Readout Unit (CRU). From the CRU, the individual data fragments are ordered by their geometrical position on the pad rows and are forwarded to the O^2 computing infrastructure on a pad-by-pad basis.

2. THE O^2 COMPUTING INFRASTRUCTURE

The O^2 computing infrastructure will support both online synchronous data reduction and asynchronous and iterative data processing [2]. An overview of the infrastructure, from the generation of the raw data on the detector level till the final data storage on the Storage Servers, is depicted in Fig. 2.

The computing infrastructure consists of two different categories of computing nodes, the First Level Processors (FLPs) and the Event Processing Nodes (EPNs). Approximately 250 FLPs constitute the front-end of the O^2 infrastructure; their role will be to read out the raw data samples as produced by the ALICE detectors over the front-end links, to perform low-level standalone processing (clusterization, masking, calibration), and to compress the data by up to a factor of 2.5 by converting raw to clusterized data and applying lossless compression algorithms. The resulting data will subsequently be merged into data packages called Sub-Time Frames (STFs), that are temporary stored locally before being transmitted to the EPNs over a switching network. Each FLP will collect data from a specific subdetector only, the produced STFs therefore will contain only partial information from the physics event that was buffered in the subdetector electronics.

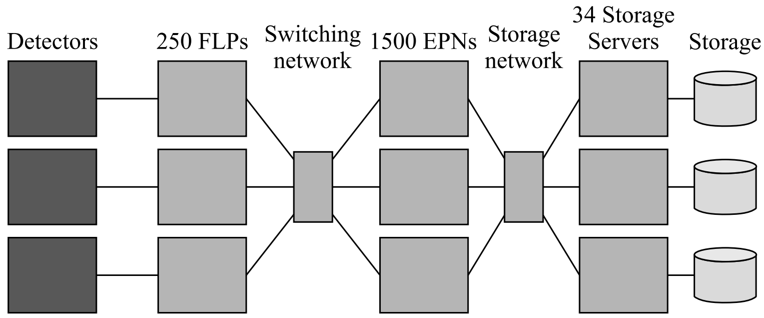

Fig. 2. High-level overview of the data aggregation, from the fragments read-out from the detectors to the full Time Frames stored on the Storage Servers [2]

Around 1500 EPNs will aggregate from the FLPs all the STFs corresponding to the same event in order to form complete Time Frames. The EPNs will also perform additional processing on the data, including track and event-level reconstruction, as well as further compression of the data volume up to an estimated factor of 8 with lossy data reduction by discarding noninteresting clusters. To reduce data traffic contention when several FLPs are sending to the same EPN, a traffic shaping mechanism will be used, which will buffer the STFs at the FLPs and use a delayed transmission schema to the EPNs [7].

The data dispatch and building components needed by both FLPs and EPNs are implemented using ALFA, as a part of the AliceO^2 computing framework. ALFA is a modular set of packages including a data transport layer and the capability to coordinate multiple data processing components. Processing tasks are organized into topologies, consisting of independent processes called *devices* that communicate on either inter-process or distributed fashion using asynchronous message queues. The devices are grouped into three categories:

• sources — devices without inputs that feed the processing pipeline with data;

• message-based processors — devices that operate on messages without interpreting their content;

• content-based processors — devices that access and process the data, employing user algorithms.

One such task that will be part of the AliceO^2 computing framework as a *content-based processor* device is a track reconstruction.

3. USING THE HOUGH TRANSFORM METHOD FOR TRACK RECONSTRUCTION

The task of track reconstruction is to reconstruct the path which a charged particle took through the active components of a detector by combining the hits as registered by the readout pads to tracks. From the parameters describing the tracks, the transverse momentum can be obtained. By extrapolating tracks to the inner part of the detector, secondary vertices can be reconstructed; by extrapolating tracks to the outer part of the detector, the entering point into the calorimeter can be calculated so that measurements in the calorimeter can be matched to the particle which produced the track [8].

One method that can be used for track reconstruction on the TPC is Hough Transform [9] (HT), an algorithm commonly used in computer vision applications to determine all the possible lines in an image based on a given set of image pixels. When used for track reconstruction, HT performs global track recognition by the identification of local patterns in a parameter space and mapping image space points to parameter space curves.

Since the solenoid magnetic field of ALICE is almost constant, the trajectories of the particles originating from a primary vertex on the beam axis are helices, if multiple scattering and energy losses in the detector volume are neglected. The particles follow a circular trajectory in the transverse plane; reconstructing a track is then equivalent to finding the parameters that describe their trajectory. Different methods can be used for parameterization, but an efficient way is to transform the particle trajectories to straight lines by using the conformal mapping [10] transformation. This leads to uncorrelated variables that are closely related to the geometry of the TPC and to simple mathematical calculations.

As described in [4], two circles can be defined inside a TPC sector, the first lying near the middle edge of the sector, while the second one lying near the outer edge. Then, as shown

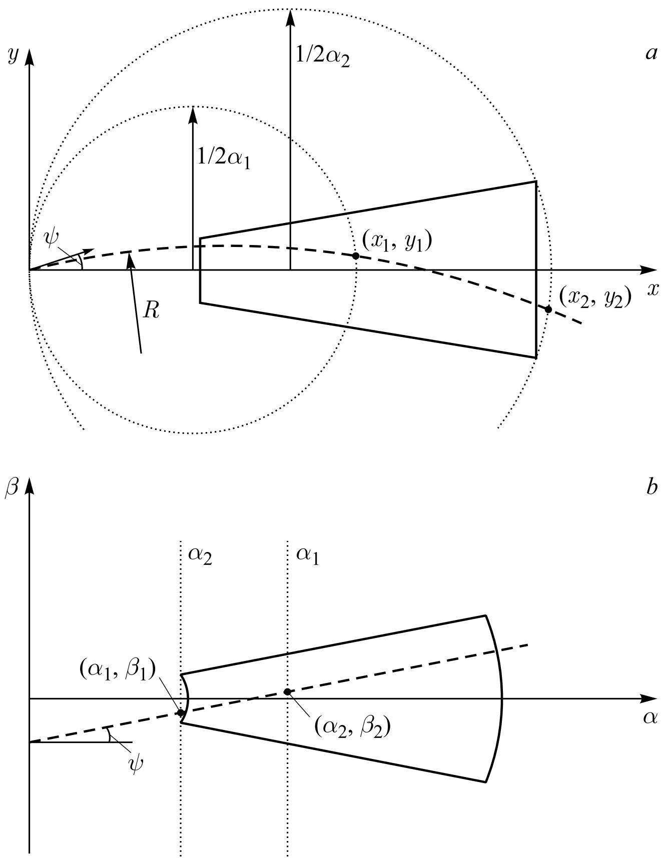

Fig. 3. *a*) A schematic view of a TPC sector on the Cartesian cluster system. The dashed line represents a track crossing the sector. The two defined circles are also visible. *b*) The same view in the conformal mapping space. The track and the two circles are visible as straight lines [4]

in Fig. 3, *a*, the track crossing the TPC sector can be represented by two points (x_1, y_1 and x_2, y_2) on these circles. The circles are given by the constants α_1 and α_2:

$$\alpha^1 = \frac{x_1}{r_1^2} = \frac{x_1}{x_1^2 + y_1^2}\, \alpha^2 = \frac{x_2}{r_2^2} = \frac{x_2}{x_2^2 + y_2^2}. \tag{1}$$

The track can be parameterized using the variables β_1 and β_2:

$$\beta^1 = \frac{y_1}{r_1^2} = \frac{y_1}{x_1^2 + y_1^2}\, \beta^2 = \frac{y_2}{r_2^2} = \frac{y_2}{x_2^2 + y_2^2}. \tag{2}$$

Then, for each space point along the track trajectory it is true that

$$\frac{\dfrac{y}{x^2 + y^2} - \beta_1}{\dfrac{x}{x^2 + y^2} - \alpha_1} = \frac{\beta_2 - \beta_1}{\alpha_2 - \alpha_1}. \tag{3}$$

If A and B are defined as

$$A = \frac{\dfrac{y}{x^2+y^2}(\alpha_1 - \alpha_2)}{\dfrac{x}{x^2+y^2} - \alpha_2}, \qquad B = \frac{\dfrac{x}{x^2+y^2} - \alpha_1}{\dfrac{x}{x^2+y^2} - \alpha_2}, \tag{4}$$

then Eq. (3) transforms into a straight line in the conformal mapping parameter space

$$\beta_1 = A + B\beta_2. \tag{5}$$

A track contained fully inside the TPC sector is represented in the parameter space by a series of straight lines covering the whole azimuthal angle around the track parameters position and thus producing a round peak. This allows one to achieve good track reconstruction efficiency, to minimize the amount of fake track candidates, and to extract uncorrelated track candidate parameters [4].

Since the offset A and the slope B of the line depend only on the Cartesian coordinates (x, y) of the digit and the constants α_1 and α_2, they can be precalculated for each of the TPC pads and stored to look-up tables to speed-up the execution of the method.

4. OVERVIEW OF TRACK RECONSTRUCTION ON THE O^2 INFRASTRUCTURE

The HT method presented in this paper will be executed on the O^2 computing infrastructure to perform track reconstruction using a series of individual processing steps; a high-level overview of the procedure can be seen in Fig. 4.

The HT device will receive as an input clusterized data. For this reason, it will be executed on the FLPs after the TPC raw data (pad, arrival time and signal amplitude information) are converted to clusters by a one-dimensional FPGA algorithm. On the first run, HT will undergo a preprocessing phase to allocate and setup all the necessary memory structures of the method, as well as to calculate the A and B parameters and to store them to look-up tables, as described in Sec. 3 above.

The track reconstruction will be performed on the FLPs on a per TPC sector basis. The FLPs will not have information for a full TPC sector, since the STFs will contain only partial

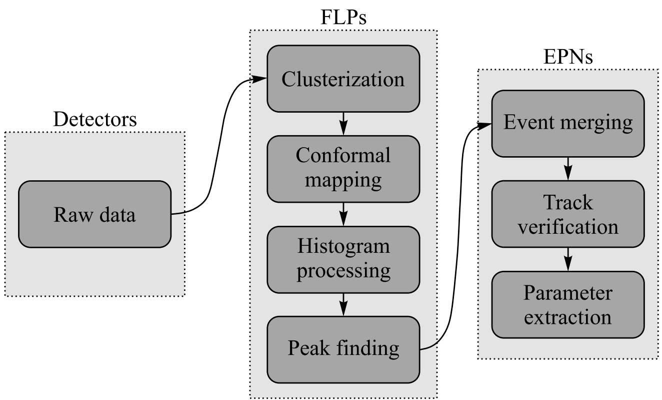

Fig. 4. Overview of the main steps in the HT based reconstruction for the TPC

information from the physics event but instead each FLP will have access to only parts of the TPC sector data. For this reason, a *partial* reconstruction will be performed at this stage with a final merging being performed at a later stage on the EPNs.

Due to the nature of the continuous readout, the FLPs will not have information on the event timestamps. The *event extraction* phase, where tracks and secondary vertices attribute to primary vertices and their association to trigger timestamps from the FIT detector, will be executed as the final step of the asynchronous Time Frame reconstruction on the EPNs. Instead, a time window will be used on the FLPs to separate clusters from different events based on arbitrary timestamps. The candidate tracks produced by the HT device at this stage will have to pass through a verification phase on the EPNs.

The HT method will then perform the conformal mapping transformation on the cluster data; each cluster with a charge greater than a specified threshold will be transformed to a line in the parameter space. Two different parameter space storage methods will have to be considered for the implementation of the device:

• A series of two-dimensional histograms. The histogram bin size and limits must be chosen to obtain the best compromise between the efficiency of the track reconstruction, the resolution and the time performance.

• A key-value pair map container, a solution used in [8]. The use of a map container can reduce the necessary memory space needed for the algorithm but can be slower in terms of CPU utilization.

The value of all the histogram bins/map entries that are crossed by the line are increased by a certain weight. In the original implementation of the method [4], a weight corresponding to the count of the occupied TPC pad rows along the track was preferred over a constant weight or a weight equal to the charge count of the point; all three methods will be reevaluated for the current implementation. The bins with the highest weight correspond to the point where all parameters intersect. Since there can exist bins with the same weight, the adjacent bins to the one with the local maxima have also to be inspected. The bins found in this way give an approximation of the track parameters.

The resulting data structures will be passed to a peak finder on the FLPs to locate all neighbour filled bins/entries and form track candidate peaks. The resulting tracks are *candidates* and will be verified or rejected on the EPNs, after the Time Frames are aggregated.

The partial peaks/tracks of the individual peak finders will be transmitted to the EPNs as a part of the STFs. Each EPN will have full information on a specific event from all TPC sectors and in addition the time reference from the ITS and TRD detectors. The output of the partial HT executions on the FLPs will be merged while processing aggregated Time Frames.

• Track candidates situated too close to the entrance or exit of the data chunk of a given FLP will have to be merged.

• Track candidates situated too close to the entrance or exit of a given TPC sector will also have to be merged.

• Track candidates formed from clusters of different events will have to be carefully reevaluated and possibly rejected as fakes.

The track parameters obtained with this method are only an estimate of the original ones, due to the binning of the Hough space. To improve the estimation of the track parameters, a simple track fitting algorithm will be implemented at the EPNs.

As the last step, the tracks from all TPC sectors are collected on the EPNs and stored in an event-summary data-block.

CONCLUSIONS

This paper presented the HT method that can be used to perform track reconstruction for the ALICE TPC for Run3. The method is based on prior work for the ALICE experiment, showing good and stable track reconstruction performance.

HT is currently developed as a content-based processor device, as part of the AliceO^2 software framework and will be executed on the O^2 computing infrastructure.

REFERENCES

1. *The ALICE Collab.* The ALICE Experiment at the CERN LHC // J. Instr. 2008. V. 3, No. 08. P. S08002.
2. *Buncic P., Krzewicki M., Vande Vyvre P.* Technical Design Report for the Upgrade of the Online–Offline Computing System // Tech. Rep. CERN-LHCC-2015-006. ALICE-TDR-019. Geneva: CERN, 2015.
3. *Al-Turany M. et al.* The New ALICE–FAIR Software Framework // J. Phys.: Conf. Ser. 2015. V. 664, No. 7.
4. *Cheshkov C.* Fast Hough Transform Track Reconstruction for the ALICE TPC // Nucl. Instr. Meth. A. 2005. P. 35–39. ALICE-INT-2005-024.
5. *Dellacasa G. et al. (ALICE Collab.).* ALICE: Technical Design Report of the Time Projection Chamber. CERN / LHCC 2000-001 ALICE TDR 7. 2000.
6. Upgrade of the ALICE Time Projection Chamber. Tech. Rep. CERN-LHCC-2013-020. ALICE-TDR-016. Geneva: CERN, 2013.
7. *Rybalchenko A. et al.* Efficient Time Frame Building for Online Data Reconstruction in ALICE Experiment // J. Phys.: Conf. Ser. 2015.
8. *Heinze I.* Development of a Hough Transformation Track Finder for Time Projection Chambers. Ph. D. Dissertation. Hamburg Univ., 2013.
9. *Duda R. O., Hart P. E.* Use of the Hough Transformation to Detect Lines and Curves in Pictures // Commun. ACM. 1972. V. 15, No. 1. P. 11–15.
10. *Yepes P.* A Fast Track Pattern Recognition // Nucl. Instr. Meth. A. 1996. V. 380, No. 3. P. 582–585.