AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE
WYDZIAŁ FIZYKI I INFORMATYKI STOSOWANEJ

KATEDRA ODDZIAŁYWAŃ I DETEKCJI CZĄSTEK

Projekt dyplomowy

Exploration of adaptive Hough algorithm variants for charged particles tracks finding for the ATLAS experiment for the LHC Run 4

Eksploracja adaptacyjnych wariantów algorytmu Hougha do wyszukiwania torów cząstek naładowanych dla eksperymentu ATLAS dla LHC Run 4 .

Autor:	Stefan Horodeński
Kierunek studiów:	Fizyka Techniczna
Opiekun pracy:	dr hab. inż. Tomasz Bołd, prof. AGH

Kraków, 2023

Contents

Chapter 1

Introduction

The result of the data-collecting process in the tracking detectors, such as ATLAS at Large Hadron Collider at CERN [1], is an array of hits that indicate spatial coordinates of the sensor, where the active medium has been ionized which generates a measurable signal. Reconstructing the parameters describing the particle (such as transverse momentum – p_t, pseudorapidity – η, angle of particle emission – φ_0 or impact parameter of the track – d_0), which is the main objective of the particle tracking, requires connecting space-points originating from the same particle. This is a straightforward task for single particle events but gets noticeably more complex for real-life events in which a multitude of charged particles pass the detector at once. For data-taking processes at LHC Run 1 to Run 3 pile-up is estimated at 40–60 interactions per bunch crossing. Run 4 will provide data characterized by up to 200 interactions in each bunch crossing while maintaining the same high collision frequency equal 40 MHz [2]. HL–LHC (high luminosity LHC) is expected to commence in 2028. It will collect roughly 5–10 times more data over the collider's operation. ATLAS and CMS detectors will be upgraded during the Long Shutdown to handle the increased data volume. At the same time, the data collection systems need to be upgraded to cope with increased data volume. The online events filtering will require the particle track reconstruction and yet that will be more challenging due to increased pile-up. The thesis aims to show detailed studies and optimizations of the Hough algorithm which is one of the promising approaches to track reconstruction, summarize the algorithm used for solutions finding, and assess the efficiency and overall performance of that approach.

1.1 The Large Hadron Collider

The Large Hadron Collider (LHC) is the world's largest particle accelerator, where protons and/or heavy ions collide. The main components of the accelerator are two, almost 27 km long beam pipes, equipped with $9,593$ superconducting magnets curving the particle trajectory and focusing the beams. At the LHC two counter-rotating particle beams are accelerated by an electric field inside 16 radiofrequency cavities operating at frequency 400 MHz [4]. Each cavity reaches the potential of the electric field equal 2 MV which enables reaching the target proton energy of 6.5 TeV from the 450 GeV injected protons. Accelerated particles are grouped in bunches, each containing around 1.2×10^{11} protons at the start. Simultaneously

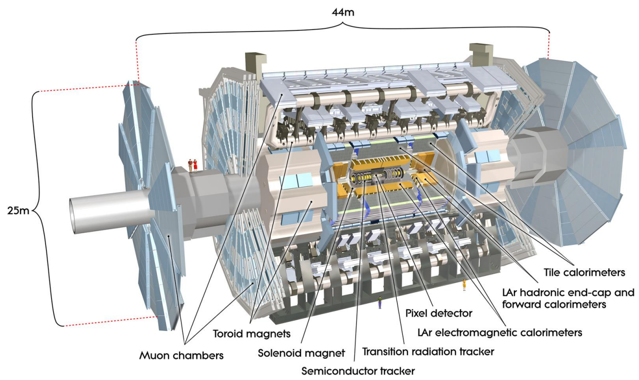


Figure 1.1: ATLAS detector layout [3]

2,808 bunches are present in each of two proton beams. Particle beams are brought to collisions in four points, each corresponding to one of four main detectors: ATLAS, CMS, ALICE, or LHCb. Bunch–crossing is observed every 25 ns which translates into collision frequency equal to 40 MHz.

The technology used in the current run (Run 3) allows colliding protons with the center of mass energy equal $\sqrt{s} = 14\,\mathrm{TeV}$ and luminosity of $1 \times 10^{34}\,\mathrm{cm^{-2}\,s^{-1}}$, and respective energy of 5.36 TeV per nucleon and peak luminosity of $1 \times 10^{27}\,\mathrm{cm^{-2}\,s^{-1}}$ for heavy ions (Pb).

The energy of colliding particles results from gradual acceleration in a cascade system of linear and circular accelerators. Linear accelerator 4 (Linac 4) is the source of protons for various CERN experiments, it delivers 160 MeV hydrogen (H^-) ions to Proton Synchrotron Booster (PSB), which before injection to this accelerator are stripped of two electrons. The PSB accelerates protons to 2 GeV before injection to Proton Synchrotron which pushes the beam to 260 GeV. The last pre-acceleration step is performed in the Super Proton Synchrotron (SPS) which provides 450 GeV protons which consecutively are injected clockwise and counterclockwise into the Large Hadron Collider [5].

1.2 The ATLAS detector

ATLAS is the largest (in terms of physical dimensions) detector operating at LHC with a length equal to 46 m and 25 m diameter [6]. To evaluate the momentum of particles, the detector elements are immersed in a magnetic field generated by three systems: central solenoid, barrel solenoid, and end-cap solenoid. The magnetic field generated by the central solenoid equals to 2 T and generated by both toroids equals to 4 T.

The main detector elements of ATLAS are the inner tracker (IT), liquid argon calorimeter, tile calorimeter, and muon detectors. The inner detector measures the momenta of all the charged particles, the liquid argon calorimeter measures the energy deposition of hadrons as well as charged particles, and the tile calorimeter is dedicated to measurements of hadrons. The inner tracker is crucial for particle tracking in high–energy physics since data collected from these detectors is used as input to the algorithm evaluating parameters of generated particles.

ATLAS detector covers full azimuth angle (φ_0), range of polar angle (θ) expressed as pseudorapidity (η) which is covered by ATLAS ranges between different detector systems. The inner detector provides tracking for $|\eta| < 2.5$, the calorimeter system – $|\eta| < 4.9$ and the muon spectrometer performs in pseudorapidity in the range $|\eta| < 2.7$ [7].

1.3 Inner detector

The inner detector is composed of pixel detectors, semiconductor trackers (SCT), and transition radiation trackers (TRT). Figure 1.2 presents the layout of inner detector elements. Pixel detectors are the first detection layer located just 3.3 cm from the interaction point [8]. This part of the inner detector is made of 92 million pixels, organized into four layers of silicon pixels, which provide data on spatial coordinates of the interaction point with accuracy 10 µm. Besides the barrel layers, also three end-cap discs are used. Pixel detectors located in the very center of the ATLAS detector have the greatest spatial resolution to differentiate between different particles in the zone of the highest track density. The high particle count also induces significant radiation damage to the active detector components. The signal yield after the Run 2 has reduced by 25% and is expected to decrease further to 50% by the end of Run 3 [9]. The pixel detector contributes to tracking performance for pseudorapidity in the range $|\eta| < 2.5$ [10].

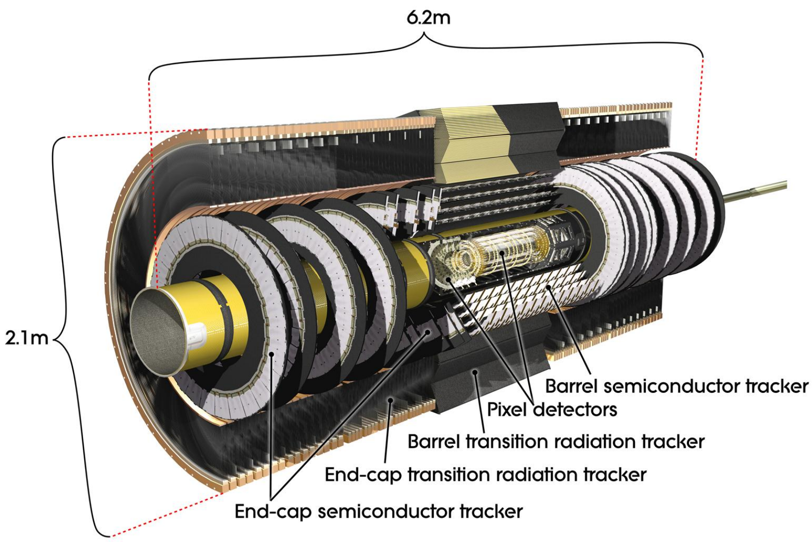


Figure 1.2: Layout of the ATLAS Inner Detector [11]

The consecutive layer of the inner detector is the semiconductor tracker, which consists of 6 million micro-strips. The detector design ensures that each particle crosses at least four layers of these detectors, which provide accuracy of 25 µm. Four thousand modules consisting of six million channels compose this layer of the ATLAS detector. For particles originating in the beam–interaction regions, the design of the semiconductor tracker assures that the track usually has eight measurements, which translate into four space-points [12]. Modules are located almost parallel to the beam axis and the magnetic field. In the end-cap discs, the strip direction is positioned according to the radial direction.

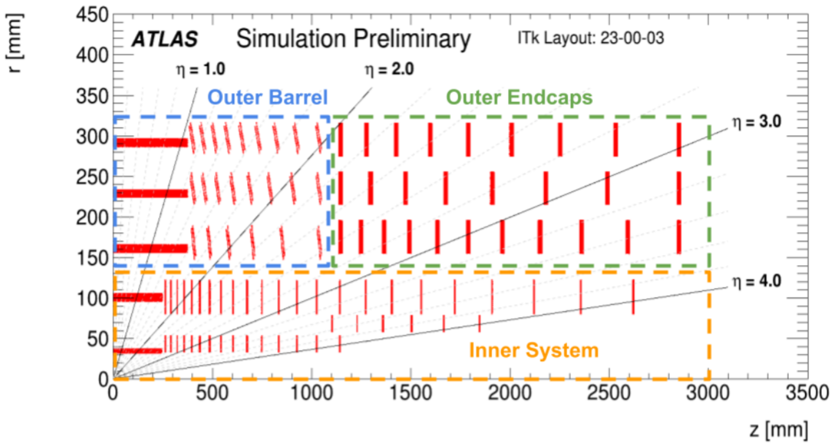


Figure 1.3: Schematic picture of the layer's position in the ATLAS Inner Tracker

The last detector layer inside the inner tracker is the Transition Radiation Tracker. Contrary to the two previous layers it does not rely on semiconductor detectors but rather on gas detectors. Three hundred thousand drift tubes form this detector layer. Each tube is filled with gas which is ionized by traveling charged particles. Diameter of tubes is only 4 mm, which translates into detection accuracy of 0.17 mm [8]. This type of detector in addition to the reconstruction of particle track trajectory, like two previous layers, contributes to the identification of ionizing particles, based on transition radiation.

1.4 ATLAS Inner Tracker upgrade

To handle increased data volume at each bunch crossing, and especially to cope with increased track density close to the vertex position, for HL–LHC the Inner Detector will be replaced by the Inner Tracker (ITk) [13]. The replacement detector system should be designed so that it can operate in increased particle flux and deposited dose in the sensors. Simulations of the new semiconductor components have demonstrated that tracking efficiency for an average interaction bunch crossing equal to 200 will be comparable to the Inner Detector tracking efficiency for 38 interactions per bunch crossing. The schematic layout of the Inner Tracker is depicted in figure 1.4.

The ITk will be an all-silicon detector composed of two main types of sensors: silicon strips and pixel sensors. Due to high charged particle flux, two inner pixel layers were designed to be replaced with around 2000 fb^{-1} of data collected. Four barrel layers and 12 endcap discs are composed of strip sensors. The detector is divided into a barrel region and two endcaps. Such an ITk geometry extends tracking coverage to $|\eta| < 4.0$. ITk will abandon the Transition Radiation Tracker.

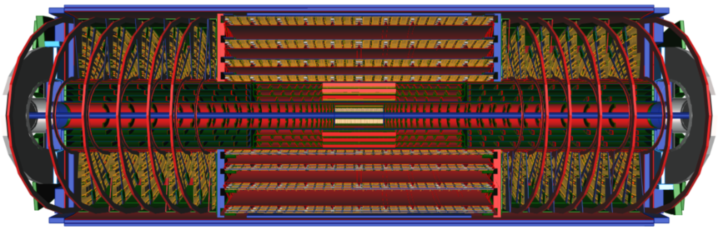
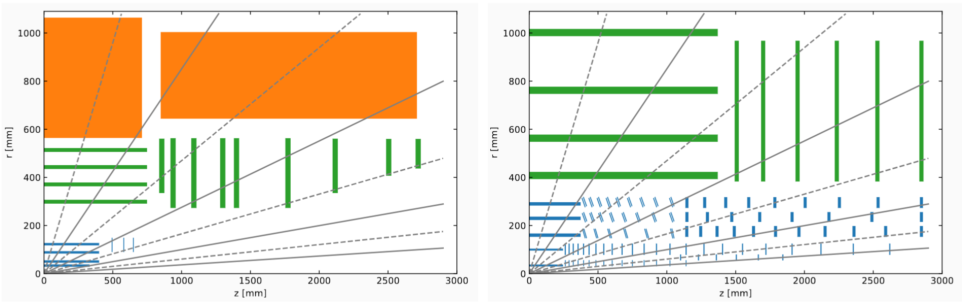


Figure 1.4: The ATLAS Inner Tracker [13].



(a) ATLAS Inner Detector [14].

(b) ATLAS Inner Tracker [14].

Figure 1.5: Comparison of the ID and ITk layout.

1.5 Particle tracking

Charged particle traversing detector volume ionizes materials which generates a signal that is accordingly analyzed by the electronics. After the readout, the output of the detector can be thought of as a set of hits indicating space coordinates of points where particle detection has been recorded. The hits are used to form space-points. For pixel detectors, space-points can be defined as the center of a cluster of pixels with the signal. In the case of strip detectors, such as Semiconductor Tracker, space-point coordinates are calculated from the intersection point of strips on adjacent layers. Thus, the resolution for a strip detector, besides the inherent uncertainty of a single strip, depends also on the incident angle of the track with respect to the module surface.

The goal of particle tracking is to extract information about detected particles (such as transverse momentum – p_t, the azimuth angle of particle emission – φ_0, pseudorapidity) from the gathered measurements. Particle transverse momentum can be estimated from track curvature in the magnetic field. For the single events (only one particle in each event), connecting space-points into a track is straightforward – all detection points originate from the same particle. Still, complications such as inhomogeneous magnetic field as well as misalignment and miscalibrations must be taken into consideration. For high pile-up events, which is the reality at the ATLAS detector, first space-points originating from one particle must be identified before particle parameters can be estimated [15].

Chapter 2

Hough algorithm

Hough transform (HT) is a technique used e.g. in image analysis and digital image processing to isolate particular shapes within the image. The transformation equation is based on the parametric form of the searched shapes, so the procedure is primarily used to detect regular shapes like lines, curves, or circles [16]. In the area of physics, the Hough transform can be used to investigate microscopic crystalline structures from patterns obtained using Electron Backscatter Diffraction [17]. Application of the Hough transform includes locating tracks in the detector volume and extraction of the estimates of particle charge, transverse momentum, and azimuth angle. This thesis, besides presenting the algorithm in the field of high energy physics aims at presenting the Adaptive Hough transform algorithm (ADT), which is an upgrade of the pure Hough transform which optimizes memory consumption, required for particle tracking. The procedure is based on a recursive algorithm that investigates prospective areas of the parameter space and skips most of the parameter space.

2.1 Introduction to the Hough transform

The core idea of the Hough algorithm is to transform a point in the real space (x, y) into a line in the Hough space (a, b). Likewise, a curve or a given shape can be depicted as a point in the Hough space. Real space is defined as a set of points that are measured in an experiment. In the case of high–energy physics and track reconstruction, these points are space-points indicating all sensor ionization that caused a generation of a signal.

In the original paper describing the transformation published by Paul Hough in 1962 [18], the author suggested that lines in real space should be parameterized by their slope and intercept, thus a representation of a line

$$y = a \cdot x + b. \tag{2.1}$$

in real space is a point (a, b) in Hough space. Such a transformation is depicted in the figure 2.3. The point coordinates along the x axis are determined by the slope of the line and the y axis coordinates by the line intercept [18].

However, such a parametrization for lines whose slope approaches infinity tends to have x coordinate also approaching infinity, which makes it unusable for most of

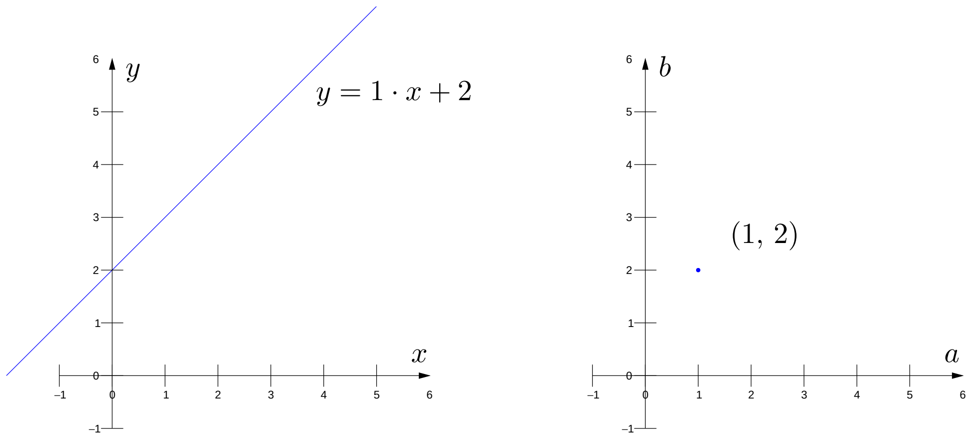


Figure 2.1: Slope–intercept straight line parametrization in the Hough transform.

the applications. To overcome this difficulty Richard O. Duda and Peter E. Hart in 1972 put forward alternative line parametrization which limits the range of points in the Hough space [19]. Slope and intercept parametrization in the Hough space is replaced with the distance to the line from the origin of the coordinate space and the angle between the line connecting the origin to the line and the x axis. That parametrization is presented in the figure 2.2.

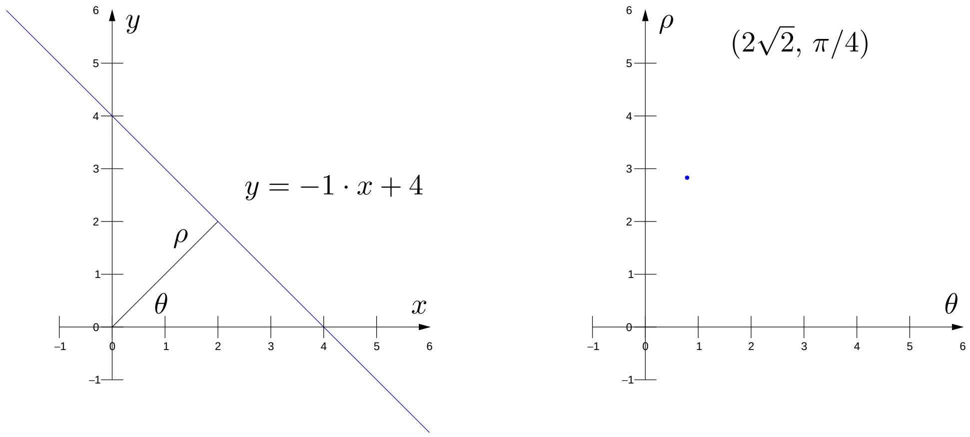


Figure 2.2: θ–ρ straight line parametrization in the Hough transform.

Likewise, a point in the real space $(x,\ y)$ can be transformed into a curved line in the Hough space $(\rho,\ \theta)$ according to the equation:

$$\rho = x \cdot \cos(\theta) + y \cdot \sin(\theta). \tag{2.2}$$

This curve can be understood as a collection of points, each representing one of infinitely many lines crossing the point, which has been transformed by the Hough equation.

Pictures of points in the real space forming a line, generate a collection of curved lines in the Hough space which intersect in a single point – (ρ_s, θ_s). This point corresponds to the parameters of the line that passes through all of the points in the real space – figure 2.4.

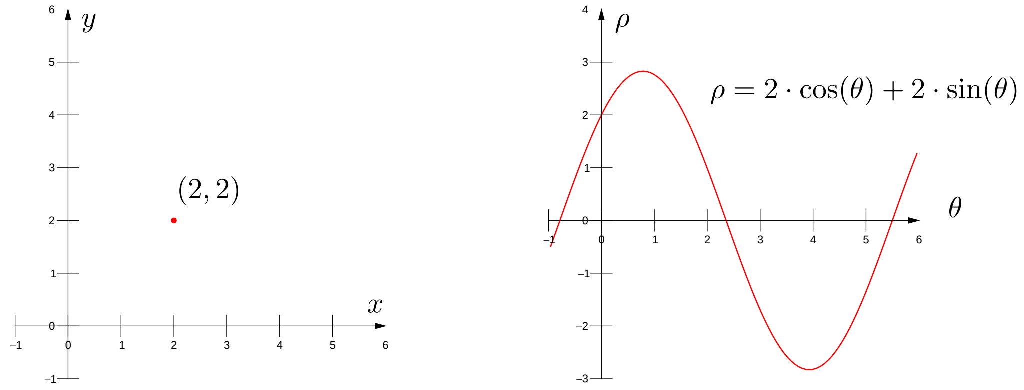


Figure 2.3: θ–ρ single point parametrization in the Hough transform.

The equation 2.2, provides a useful tool for the calculation of straight-line parameters, however, due to the presence of the magnetic field in the detector volume, their trajectories are curved forming a helix. Applying the Hough transform requires a form of the transforming equation that is fit for that purpose. The appropriate equation is derived from charged particle interaction with the magnetic field.

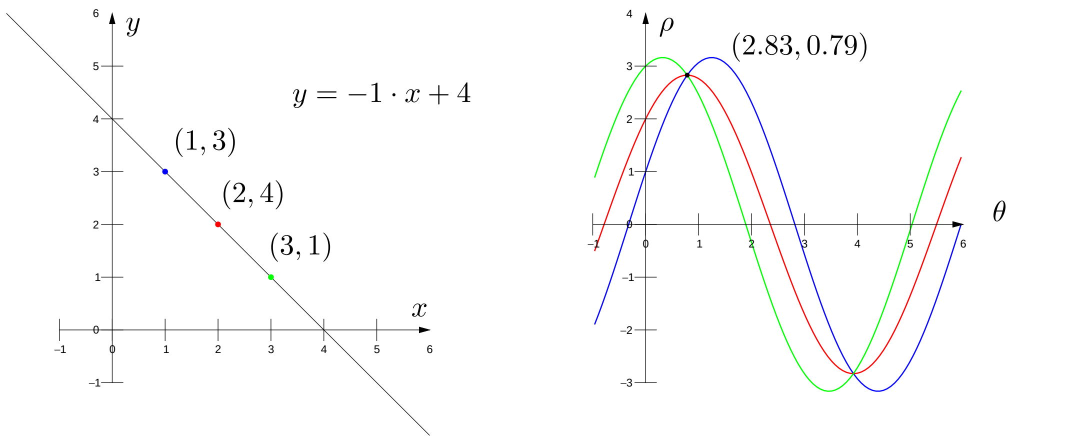


Figure 2.4: θ–ρ parametrization for points aligning into a straight line in the Hough transform.

2.2 Hough transform for charged particles

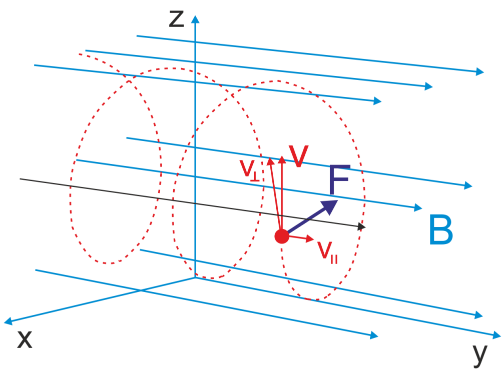
The form of Hough transform used for particles tracks parameters estimation, results from the Lorentz equation, which governs the motion of charged particles in the electromagnetic field

$$\boldsymbol{F}_L = q(\boldsymbol{E} + \boldsymbol{v} \times \boldsymbol{B}), \tag{2.3}$$

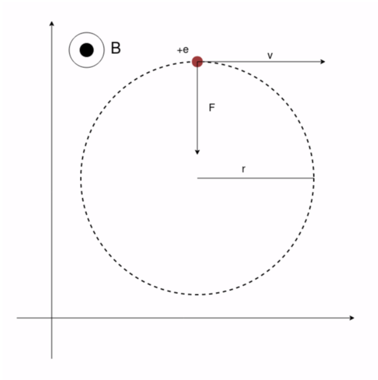
where F_L – Lorentz force, B – magnetic field induction, E – electric field and v denotes velocity of the particle [20]. The equation in the scalar form and assuming no electric field (which is the approximation for the detector) takes on the form

$$F_L = q \cdot v \cdot B \cdot \sin(\beta), \tag{2.4}$$

where β corresponds to the angle between velocity and magnetic induction vectors. It implies that the Lorentz force is proportional only to the velocity component perpendicular to the magnetic field – $v_\perp$. The parallel component only affects the pitch of the helix – figure 2.5a. Thus, to determine transverse momentum we do not have to take under consideration the propagation of the particle along the z axis, analysis of the trajectory curvature in the xy plane is sufficient to estimate p_t and charge sign – figure 2.5b.



(a) 3D charged particle trajectory in the uniform magnetic field [20]

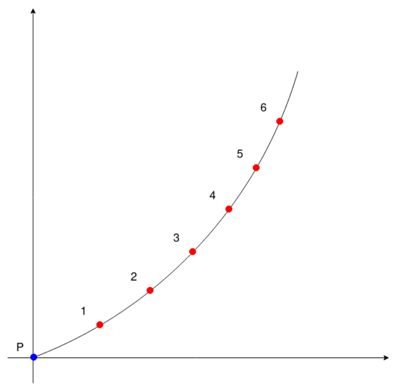


(b) 2D projection on the plane perpendicular to the magnetic field induction vector, of the charged particle helix in the magnetic field.

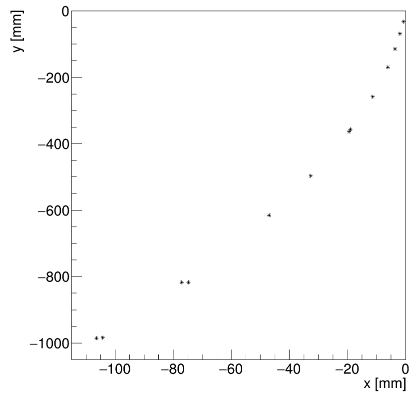
Figure 2.5: Charged particle trajectory in a uniform magnetic field.

The described algorithm transforms each space-point into a curve (which in this case is a straight line). Space-point is defined as a set of coordinates where a charged particle caused ionization of the sensor volume which resulted in the generation of a signal. We anticipate that the equation must include the magnetic field induction since this parameter determines the radius of the helix. For the single particle event, the only ionization and consecutively the only space-point detection is due to the effect of that one particle, and all detection points must follow the two–dimensional projection of the helix on the xy plane. A schematic picture of that problem is depicted in the figure 2.6a. Figure 2.6b presents the set of space-points for a single muon particle.

The derivation of the equation used in the Hough transform for the charged particle is based on the Lorentz force acting as centripetal force as well as geometrical consideration of the problem. The first relationship can be presented as



(a) Schematic image of the space-point alignment with the particle trajectory.



(b) Example of the space-points coordinates on the xy plane for single μ particle, $p_t = 3.304\,\mathrm{GeV}$, $\varphi_0 = -1.592\,\mathrm{rad}$, $q = +1$, $\eta = -1.428$.

Figure 2.6: Distribution of the space-points in thee xy plane.

$$\begin{aligned} F_L &= F_c \\ q \cdot v_\perp \cdot B \cdot &= \frac{m \cdot v_\perp^2}{R} \\ R &= \frac{m \cdot v_\perp}{q \cdot B}, \end{aligned} \tag{2.5}$$

which can be further simplified by substituting $m \cdot v_\perp$ with transverse momentum p_t. This step implies basing the calculations on classical mechanics which might seem questionable since particles in the range of a few GeV are to be tracked using this method. However, this simplified model still returns a precise enough estimate of particle parameters. Also, it was assumed in the calculation that the impact parameter d_0 (minimal distance of the helix trajectory from the $(0, 0)$ point in the xy plane) does not significantly deviate from 0. It is assumed that all the particles originate from the $(0, 0)$ point.

Parameters of the particle which can be the subject of the analysis using the set of space-points (figure 2.6b) are charge, transverse momentum, and angle of particle emission in the xy plane. Hence, radius – R in the formula 2.5 must be expressed using the angle of particle emission in the xy plane. For the transformation of the space-point coordinates to a line in the parameter space, the equation must also include the point parametrization. For each point, we can calculate the distance to the point $(0, 0) - r$, as well as the angle between the x axis and the section connecting the space-point and the center of the coordinate system. These values are defined as follows

$$r = \sqrt{x^2 + y^2}, \tag{2.6a}$$

$$\varphi = \arctan\left(\frac{y}{x}\right). \tag{2.6b}$$

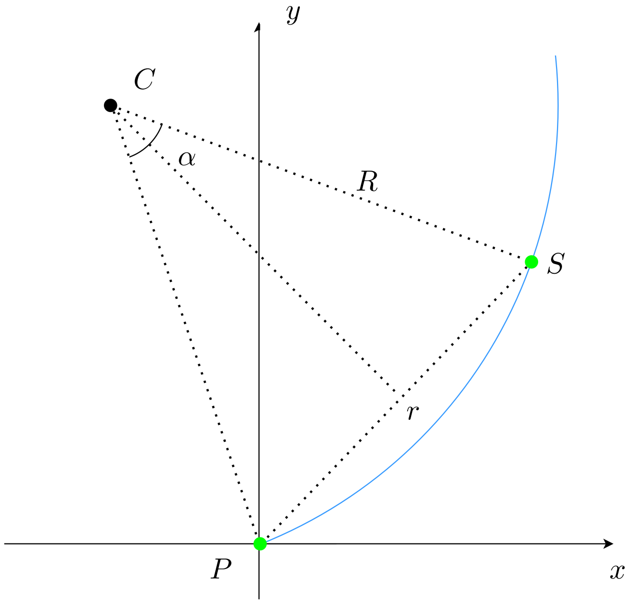


Figure 2.7: Parametrization of the particles trajectory – C – center of the helix's projection circle, R – radius of that circle, P – primary vertex – spatial coordinates where particle originates, S – analyzed space-point, α – the $\angle PCS$ angle, r – sections length between points P and S

The used convention for the coordinate system assumes that the z axis is parallel to the beam direction and that axes x and y are perpendicular.

Isosceles triangle built on the points P – particle production point, S – analyzed space-point and C – the center of the circle (2.7) corresponding to the projection of the helix onto the xy plane, allows to express radius R as a function of r and angle α

$$\sin\left(\frac{\alpha}{2}\right) = \frac{r/2}{R}. \tag{2.7}$$

Our goal in the following steps is to express angle α using φ and φ_0 – the angle of particle production.

To link the angle α with angles φ and φ_0 tangent line to the particle trajectory in the point P has been added (figure 2.8). Based on this new object φ_0 can be defined as the tangent of the slope coefficient of that line. The angle between the tangent line section connecting points P and S is denoted as δ. The angle of the isosceles triangle located next to the side perpendicular to the triangle axis of symmetry equals $\pi - \pi/2 - \alpha/2$. Thus δ equals to

$$\delta = \frac{\pi}{2} - \left(\pi - \frac{\pi}{2} - \frac{\alpha}{2}\right) = \frac{\alpha}{2}. \tag{2.8}$$

To substitute the value of α, we assume that α is small so that we can simplify the equation 2.7 by evaluating the following limit

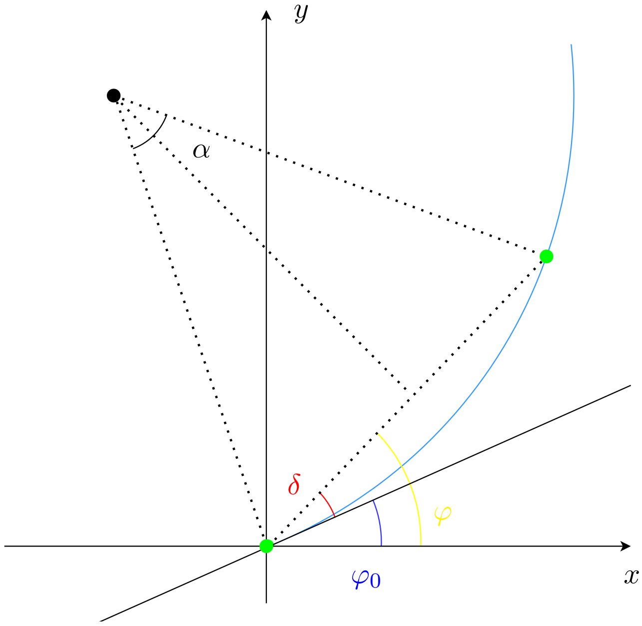


Figure 2.8: Parametrization of the particles trajectory – δ– angle between trajectory tangent at the point P and $\overline{PS}$ section, φ– angle between $\overline{PS}$ section and x axis, φ_0– angle of the particles emission

$$\lim_{\alpha \to 0} \sin\left(\frac{\alpha}{2}\right) = \frac{\alpha}{2}. \tag{2.9}$$

It is a justified approximation because the algorithm was designed to detect and parameterize high transverse momentum tracks. Analysis of the transform performance in the following chapters is concentrated on muons and pions with transverse momentum in the range $[1\,\text{GeV}, 10\,\text{GeV}]$. For such high–energy particles, we can assume that α is close to zero even for significant values of r due to the large radius R of the circular helix projection on the xy plane.

From the analysis of the figure 2.8, we can conclude that the angle δ is equal to the difference between angles φ_0 and φ. Combining this observation with equations 2.8 and 2.9 leads to

$$\sin\left(\frac{\alpha}{2}\right) \approx \frac{\alpha}{2} = \delta = \varphi - \varphi_0. \tag{2.10}$$

That equation is the searched relationship between α and angles φ_0 and φ which can be substituted into equation 2.7

$$\frac{r}{2 \cdot R} = \sin\left(\frac{\alpha}{2}\right) \approx \frac{\alpha}{2} = \varphi - \varphi_0 \quad \longrightarrow \quad R = \frac{r}{2(\varphi - \varphi_0)}, \tag{2.11}$$

that in turn defines radius R which subsequently can be merged with equation 2.5

$$\frac{r}{2(\varphi - \varphi_0)} = \frac{p_t}{q \cdot B} \tag{2.12}$$

which after additional reordering yields the formula of the Hough transform suitable for the charged particles moving in the uniform magnetic field.

$$\varphi_0\left(\frac{q}{p_t}\right) = \varphi - A \cdot r \cdot \frac{q}{p_t}. \tag{2.13}$$

The parameter A present in the equation is a constant value that incorporates information about the magnetic field induction as well as the system of units used.

To ensure a more natural visualization of the accumulator in the parameter space, x axis is assigned to the φ_0 and y axis to the q/p_t value. The transforming equation used in such a coordinate system is of the form

$$\frac{q}{p_t}(\varphi_0) = \frac{1}{rA}\varphi_0 - \frac{1}{rA}\varphi. \tag{2.14}$$

Chapter 3

Adaptive Hough algorithm

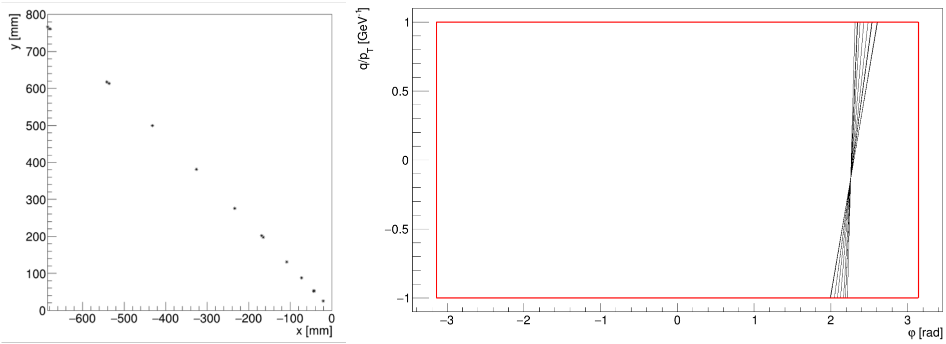
In the HT algorithm, once parameter space is populated by the lines corresponding to measurements in the image domain a peak-finding procedure is used. Typically the parameter space is quantized/histogramed/pixelized and peaks are found in by looking for bins/pixels of the value above the threshold. In consequence, this classical approach to peak finding assumes checking every single subsection of the parameter space while looking for the solutions, which is greatly a suboptimal approach. The main goal of that thesis is to demonstrate the application of the so-called Adaptive Hough Transform (AHT) algorithm, which is a modified implementation of the HT for the solution parametrization. This approach has been suggested as a means to reduce memory space consumption, at the same time retaining the flexibility of the traditional algorithm.

The results presented in the following chapters were acquired based on the Hough transform implementation prepared by Piotr Libucha and described in the thesis 'Development of Hough transform implementation for helix finding on graphic cards processors' [21].

3.1 Filling in the accumulator

The line drawing for each space-point as well as finding the solution takes place in the object called an accumulator. It is defined as a part of the parameter space $(\varphi_0, q/p_t)$ where we intend to search for the solutions. Its size and coordinates are determined by a range of solutions that we are interested in. For the analysis described in this thesis size of the accumulator is $q/p_t \times \varphi = 2 \times 2\pi$. This translates to the detection of particles in the full azimuth angle and the $[1\,\mathrm{GeV}, 10\,\mathrm{GeV}]$ transverse momentum range. The accumulator will be subject to division described in the following sections.

The transformation of the space-points to the parameter space using the equations 2.14 fills in the parameter space with lines. For each space-point we obtain a single line in the accumulator. An example of such a transformation for the single muon particle is presented in the figure 3.1. According to the equation 2.14, all lines have a positive slope. They intersect in one point indicating the parameters of the particle. Because of a high value of the transverse momentum and therefore high slope (lines are nearly vertical), we can anticipate a high uncertainty of q/p_t estimate. Because points of interest in the analysis were high–energy particles, the curvature of the trajectory was minimal which makes it hard to estimate the transverse momentum of the particle. However, because they form a nearly straight line, φ_0 could be determined to a very high precision.



(a) Example of the space-points coordinates on the xy plane for single μ particle, $p_t = 6.9\,\mathrm{GeV}$, $\varphi_0 = 2.26\,\mathrm{rad}$, $q = -1$, $\eta = 0.12$.

(b) Hough transformation of the 3.1a space-points.

Figure 3.1: Hough transformation of single charged particle example in uniform magnetic field.

As it can be noticed in the figure 2.4 and 3.1 for a set of points that forms a given shape, in the parameter space we observe points indicating the parameter form of the sought shape. So the next step is to locate areas where the lines cross which indicates solutions candidates. Multiple procedures to identify solutions have been put forward, but this chapter aims to describe the algorithm that was used for the calculations described in the thesis – the AHT.

The procedure to which this chapter is devoted is frequently described as the voting process since we try to localize solutions candidates by counting the lines that cross a given section. A section is treated as a solutions section once enough lines have been counted in the analyzed area of the accumulator.

3.2 Classical approach

The most straightforward approach to detect solution coordinates would be to iterate over all the possible sections in the accumulator. The size and number of these sections are determined by the precision we wish to obtain from the calculation. Also, as it can be noticed in figure 3.1, there are regions where it is needless to count the number of lines ($\varphi_0 < 1\,\mathrm{rad}$). Nevertheless, the classical approach analyses each of these regions of the accumulator.

In addition, a single line passes through a certain number of cells, which is a small fraction of the total number of cells that had to be subject to the entire voting procedure. The fraction is even smaller for greater algorithm precision indicating that many hopeless cells will be analyzed which do not contain a solution.

3.3 Solution finding

The most crucial difference between the AHT compared to the classical approach is the varying size of the cell for which we perform line counting. Instead of analyzing every single accumulator section, we group them and reject a collection of sections. In practice, this idea is executed by zooming in on the most interesting area of the accumulator until we reach the desired precision of the algorithm or reject the collection of sections.

The AHT is a recursive procedure, however, to design the algorithm that complies with accelerator devices like GPU, the stack of accumulator sections, has been used.

Each section is parameterized based on its size in each dimension – `xSize` and `ySize` as well as coordinates of the lower left corner – `xBegin` and `yBegin`. The coordinates of the upper right corner are (`xBegin + xSize`, `yBegin + ySize`). Analogously for the remaining two corners.

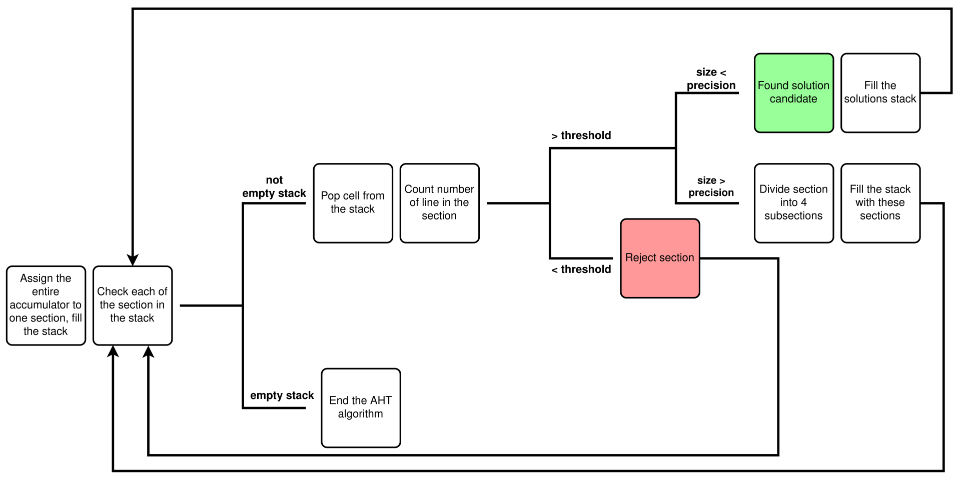


Figure 3.2: Block diagram for the adaptive Hough transform.

The AHT algorithm is explained using the block diagram presented in figure 3.2. The iterative procedure consists of the following steps

1. at the beginning, fill the section with the entire accumulator, define the coordinates of that section and its dimensions, and populate the stack with the section;

2. pop a section from the stack, transform each space-point into a line following the 2.14 equation, and verify whether the line lies within the section. For the given equation describing the line – $q/p_t(\varphi_0)$, coordinates of the lower left corner of the cell – (`xBegin`, `yBegin`) and the size along each axis – `xSize` and `ySize`, the simplest check of line location within the cells requires ensuring that q/p_t(`xBegin`) $<$ `yBegin + ySize` and q/p_t(`xBegin + xSize`) $>$ `yBegin`. Such a test is simplified from the most generic approach by assuming the positive slope of the line, according to the equation 2.14;

3. count the total number of lines within the cell, if it is greater than the `threshold` value, reject that cell, go to step 2,

4. assuming the number of lines is above the `threshold` value, check whether the section size is greater than the required precision

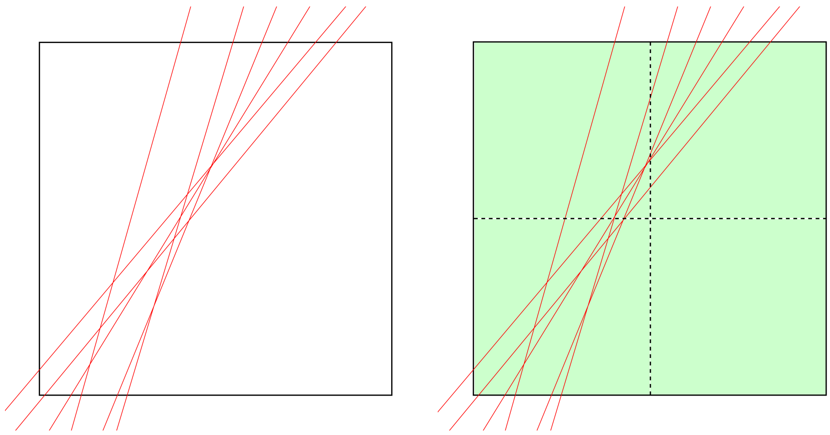
   - if the section size in both coordinates is smaller than the precision, you found a solution candidate – populate the solution buffer, parameters of the found particle: $\varphi_0 =$ `xBegin + xSize/2`, $q/p_t =$ `yBegin + ySize/2`;
   - if the section size is greater than the precision in both coordinates, divide the section into four subsections, new size in each dimension is `xSize/2` and `ySize/2`, populate the stack with new sections, this division mode has been presented in the figure 3.3, coordinates of the lower left corner for each of the four subsections are

     (a) (`xBegin`, `yBegin`)
     (b) (`xBegin + xSize/2`, `yBegin/2`)
     (c) (`xBegin`, `yBegin + ySize/2`)
     (d) (`xBegin + xSize/2`, `yBegin + ySize/2`)

   - if the section size is greater only in one of the coordinates divide the section into two subsections along the coordinate in which the section size is above the precision

     (a) for division in the x coordinate, size of the subsections are `xSize/2` and `ySize`, coordinates for the lower left corner are (`xBegin`, `yBegin`) and (`xBegin + xSize/2`, `yBegin`)
     (b) for division in the y coordinate, size of the subsections are `xSize` and `ySize/2`, coordinates for the lower left corner are (`xBegin`, `yBegin`) and (`xBegin`, `yBegin + ySize/2`)

5. continues execution of steps 2 – 4 until section buffer is empty, proceed then to the next event, for the last event end the algorithm.



(a) (b)

Figure 3.3: Division of a section into four subsections.

3.4 Optimal algorithm parameters

The end–of–loop condition for a section containing sufficiently many lines is that the section's width in both coordinates is smaller than the predetermined precision. Precision, alongside the threshold, is the key parameter that impacts the algorithm efficiency and total number of solutions. The optimal threshold can be, however, straightforwardly assigned starting with the detector geometry, and the number of detector layers that a charged particle traverses. The optimal threshold parameter value has been chosen to be equal to 6 for the application to the single muon events and pile-up events.

The optimal choice of the φ_0 and q/p_t precision is a tricky issue and that requires more in–depth analysis. The granularity of the algorithm can limit and impact efficiency and the number of solutions in a few different ways. Too small granularity (too few divisions in either coordinate) decreases the resolution of the algorithm, we are less capable of distinguishing solutions located nearby. That is a worrying aspect, especially in the case of pile-up. In addition, too small granularity reduces selectivity, we are more likely to find sufficiently many lines to declare the section a solution candidate. This directly increases the odds of finding false positive solutions.

Excessively high granularity can on the other hand significantly hinder solution finding. Due to imperfections in evaluating space-point coordinates, lines do not cross as ideally as has been presented in figure 2.4. Thus, too small a section can fail to accommodate all the Hough lines in the vicinity of the parameter coordinates reflecting the truth particle parameters. This effect noticeably reduces the algorithm efficiency. Furthermore, too high precision requires a higher number of iterations (the memory consumption only moderately depends on this choice).

3.5 Example of the AHT evaluation

The steps that must be followed to find the solution candidate using the AHT have been described in the section 3.3. That section aims to provide a better understanding of the procedure using a simplified model assuming only four divisions.

The figures in this section show only a simplified model of the AHT. It does not cover the order in which sections are analyzed as well as populating and popping sections from the stack. The threshold for that example is equal to 5. As for the case of the charged particle and transforming equation in the form 2.14, all lines have a positive slope. The number of divisions required to reach desired precision in φ_0 equals 4, in q/p_t – 3. Each figure shows a progression of the AHT for another division in both coordinates simultaneously. Gray color indicates the section rejected in the previous step, red – rejected in the current step and the green color represents the section that meets the requirement of the number of lines within that section and will be further analyzed in the following steps, or sections which are solutions candidates. Figures 3.4 and 3.5 indicate divisions in each step of the procedure.

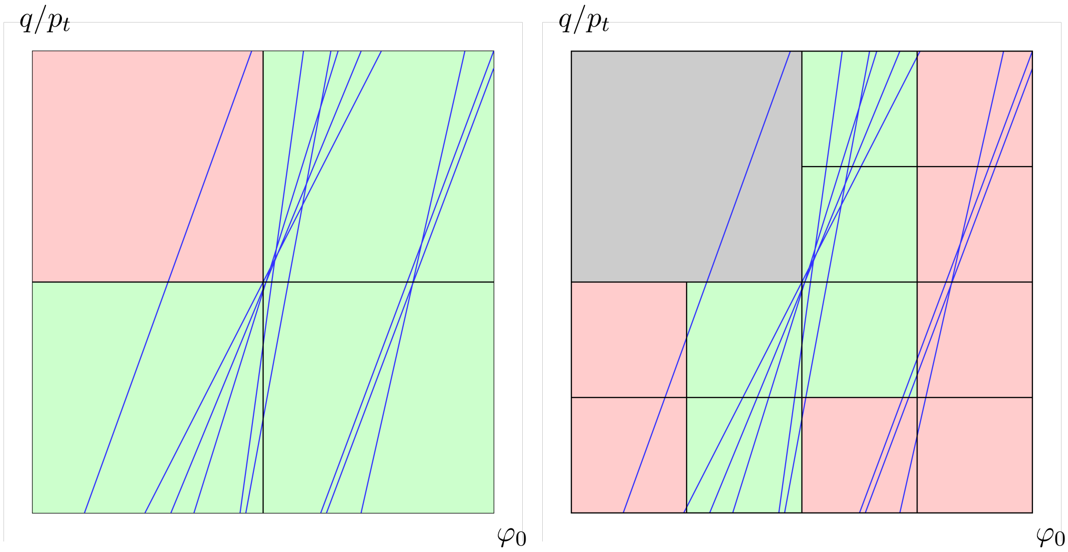


Figure 3.4: An illustration of the AHT progression in the scan of the accumulator space.

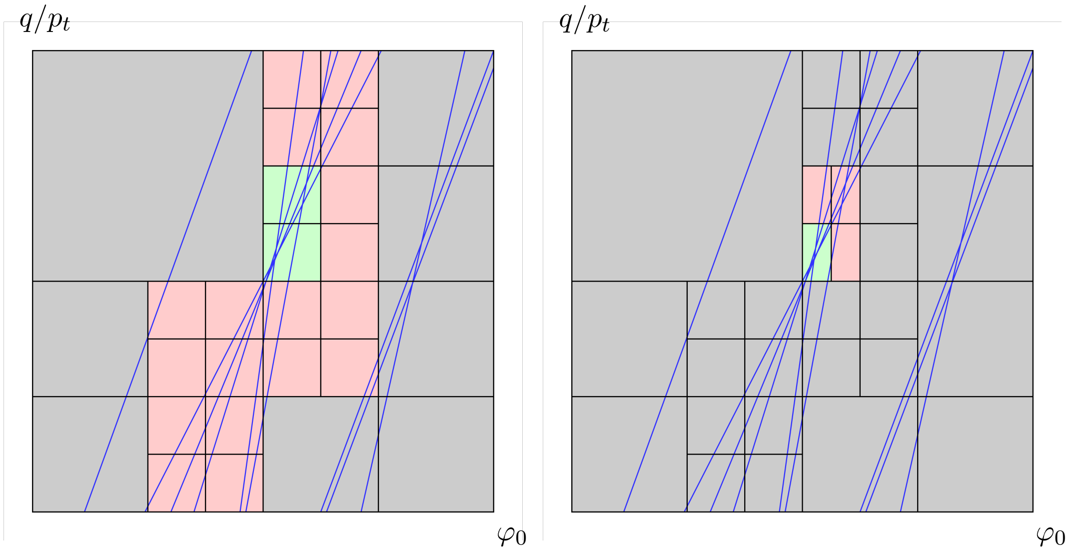


Figure 3.5: An illustration of the AHT progression in the scan of the accumulator space. Continuation of the previous image.

In the three first steps, each subsection has been divided into four sections. It is not, however, the case for the last step. Because we reached set precision in q/p_t, only division in φ_0 is performed.

Chapter 4

Data description

The adaptive Hough transform has been tested in single muon (μ), as well as pile-up events, for which the average number of pile-up interactions was 200. A single μ dataset was used primarily to verify the algorithm performance for simple cases, where all the space-points are associated with a single particle. Also, this part of the analysis allowed us to choose optimal parameters and develop new optimization techniques described in the section 6.

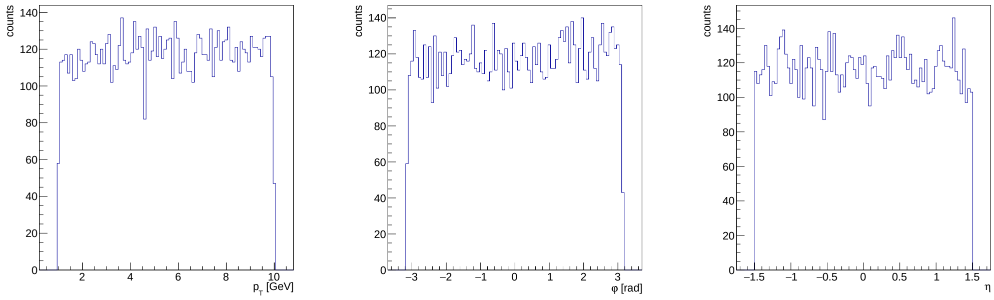
For the study of the algorithm, performance simulated response for the Open Data Detector (ODD) [22] in the ACTS toolkit [23] has been used. The detector geometry is similar to the ATLAS detector, so this analysis is a good test for the proposed algorithm.

For all the space-points types (single μ, and pile-up), the detector covers almost the full range of azimuth angle, from $2°$ to $178°$. However, in the analysis, to avoid further optimization required by space-points originating from barrel and disc geometries, a narrower range of polar has been used, from $25.2°$ to $154.8°$, which translates to pseudorapidity in the range $\eta \in [-1.5, 1.5]$. Detector covers full azimuth angle $\varphi_0 \in [\pi, \pi]$.

The magnetic field induction is assumed to be uniform in the entire detector volume and equal to $2\,\mathrm{T}$.

For the single events, to test the algorithm performance in the simplified condition, transverse momentum, angle of emission, and pseudorapidity follow a uniform distribution – figure 4.1.

The main goal of the analysis was to test the efficiency of the AHT for pileup events. This dataset intends to simulate the output of the particle detection for real-life cases. On average around $1,900$ are observed in a single event, however, only 500 particles are in the p_t range from $1\,\mathrm{GeV}$ to $10\,\mathrm{GeV}$ – figure 4.3.

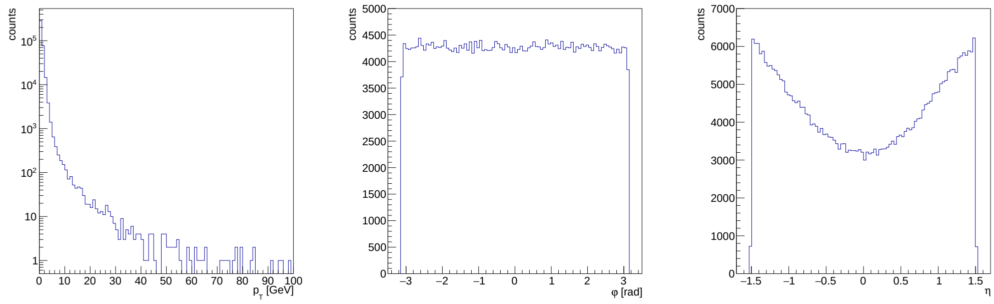


(a) Distribution of transverse momentum – p_t.

(b) Distribution of the angle of particle emission – φ_0.

(c) Distribution of pseudorapidity – η.

Figure 4.1: Distribution of particle parameter in the single μ events sample.



(a) Distribution of transverse momentum – p_t.

(b) Distribution of the angle of particle emission – φ_0.

(c) Distribution of pseudorapidity – η.

Figure 4.2: Distribution of particle parameter in the pile-up sample.

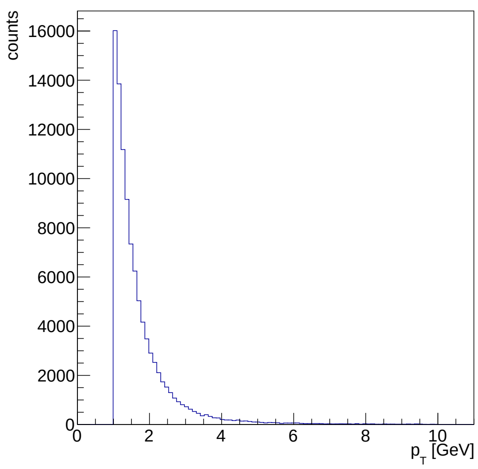


Figure 4.3: Distribution of transverse momentum in the range $p_t \in [1\,\mathrm{GeV}, 10\,\mathrm{GeV}]$ in the pileup sample.

On average a single μ events contain 13.7 space-points, for pile-up, this number increases to $25,500$, which indicates an increased complexity of the real case. 0.16% of particles in the pile-up dataset are muons, 70.64% are pions.

A more general problem that will need to be solved in reality includes the estimation of the d_0. To limit the study to the simplest case we had to ensure that non–zero value of that parameter does not impact the algorithm efficiency. To do so, space-points located in the layers located closest to the particle beam have been removed. Points meeting either of the requirements

- $r \in [0\,\mathrm{mm}, 150\,\mathrm{mm}]$ and $z \in [-550\,\mathrm{mm}, 550\,\mathrm{mm}]$
- $r \in [0\,\mathrm{mm}, 200\,\mathrm{mm}]$ and $z \in [-900\,\mathrm{mm}, -600\,\mathrm{mm}]$
- $r \in [0\,\mathrm{mm}, 200\,\mathrm{mm}]$ and $z \in [600\,\mathrm{mm}, 900\,\mathrm{mm}]$

were not included in the further calculations. In addition, to prevent analysis of events with space-point in the barrel and disc geometries region, single-muon events with at least one space-points in the end-cap region

- $r \in [0\,\mathrm{mm}, 200\,\mathrm{mm}]$ and $z \in [-1600\,\mathrm{mm}, -600\,\mathrm{mm}]$
- $r \in [0\,\mathrm{mm}, 200\,\mathrm{mm}]$ and $z \in [600\,\mathrm{mm}, 1600\,\mathrm{mm}]$
- $r \in [200\,\mathrm{mm}, 1200\,\mathrm{mm}]$ and $z \in [-3000\,\mathrm{mm}, -1200\,\mathrm{mm}]$
- $r \in [200\,\mathrm{mm}, 1200\,\mathrm{mm}]$ and $z \in [1200\,\mathrm{mm}, 3000\,\mathrm{mm}]$

were also rejected.

Data corresponding to the single μ events, saved as the `root` file consists of $10,000$ events, for pile-up events, 200 events have been analyzed.

Chapter 5

Performance for single μ event

Muons, as the least ionizing particles are the easiest for particle tracking. They are also not scattered as easily as hadrons and they are not subject to significant Bremsstrahlung radiation, like electrons, therefore their trajectory closely resembles a helix. Because of that, they are a perfect object for initial algorithm testing. We expect very high efficiency and a very low rate of fake solutions.

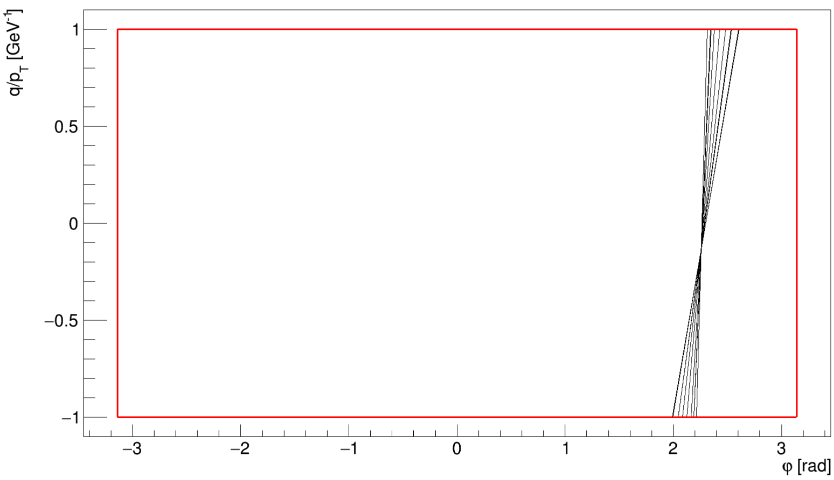


Figure 5.1: Filled accumulator for a single μ event.

Figure 5.1 displayed an accumulator filled with Hough lines transformed according to the equation 2.14. All the lines cross in a single point which indicates solution candidate parameters. There are no spurious lines that are not associated with that particle, which could generate fake solutions. For such an initial accumulator, the AHT algorithm is performed. Consecutive steps are depicted in the figure 5.2 (left). The darker the color filing a section the smaller its dimensions, and the higher the section precision or selectivity. The region of the darkest section corresponds to the point of line crossing in the accumulator 5.1. The zoomed–in part of the accumulator is presented in the figure 5.2 (right). Blue rectangles indicate section borders for the cells where the solution candidate has been found. That image demonstrates one of the issues that had to be solved for better algorithm performance. We expect to observe ideally one Hough solution per single truth particle. However, in the vicinity of the line–crossing point we observed 7 additional duplicate solutions. Significant

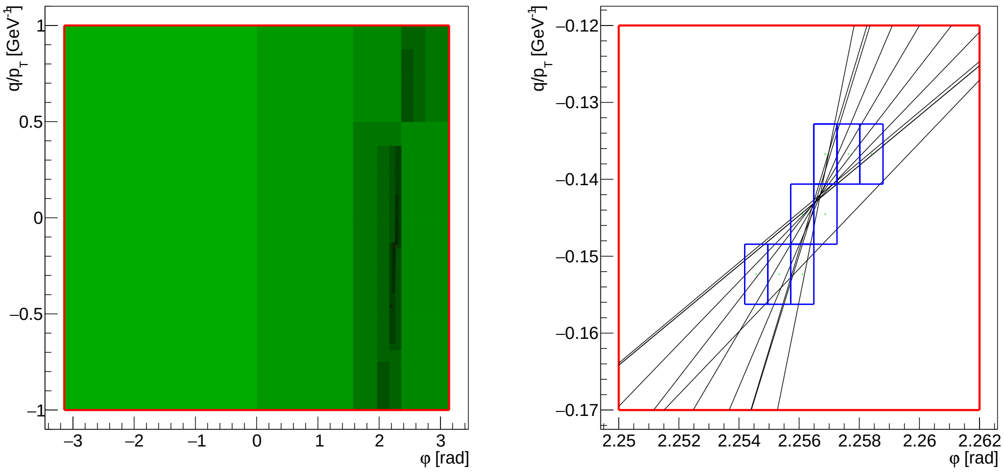


Figure 5.2: (left) Progression of the AHT algorithm, the darker the section color the further in the recursive division, (right) section of the accumulator where solutions have been identified, blue rectangles correspond to the solution candidate sections, particle parameters: $q/p_T = 0.143\,\mathrm{GeV}$, $\varphi_0 = 2.26\,\mathrm{rad}$.

effort during the investigation of the algorithm capabilities, especially for the pile–up events, has been put into reducing the number of spurious solutions. On average, for single μ events, for one truth particle falls 9.92 Hough solutions.

5.1 Optimal algorithm parameters

To evaluate the optimal set of precision parameters, efficiency, and number of solutions for a single truth particle a wide range of the accumulator precision parameters have been calculated. Let the precision in φ_0 and q/p_t be denoted as respectively ξ_{φ_0} and ξ_{q/p_t}. The number of divisions in each coordinate (n^{div}) can be expressed as

$$n^{acc}_{\varphi_0} = \mathrm{cell}\left[\log_2\left(\frac{\Delta^{acc}_{\varphi_0}}{\xi_{\varphi_0}}\right)\right] \tag{5.1}$$

$$n^{acc}_{q/p_t} = \mathrm{cell}\left[\log_2\left(\frac{\Delta^{acc}_{q/p_t}}{\xi_{q/p_t}}\right)\right] \tag{5.2}$$

where Δ^{acc} denotes the accumulator size in one of the coordinates. It is curious to note that the same efficiency and number of solutions will be obtained for a range of precision parameter values. It is the case for as long as ξ is within the (ξ_{min}, ξ_{max}) range where

$$\xi_{max} = \frac{\Delta^{acc}}{2^{n^{div}-1}} \tag{5.3}$$

$$\xi_{min} = \frac{\Delta^{acc}}{2^{n^{div}}} \tag{5.4}$$

Thus, a much more accurate approach requires expressing precision as the necessary number of divisions to reach that precision.

To evaluate optimal precision, efficiency, and number of solutions for a single truth particle, has been evaluated in the number of division ranges $n^{div}_{\varphi_0} \in [2, 14]$ and $n^{div}_{q/p_t} \in [2, 14]$. These values translate into precision in φ_0 from roughly 2.7117×10^{-4} rad to 1.1107 rad, and precision in q/p_t from $8.6317 \times 10^{-5}\,\mathrm{GeV}^{-1}$ to $0.353\,55\,\mathrm{GeV}^{-1}$. The picture 5.3 presents the efficiency (left) and number of solutions for a single truth particle (right).

It can be noted that for almost the entire range of n^{div} the efficiency is 100% but for a very high number of divisions in both coordinates. It illustrates the argument against too fine granulation described before in that section. The histogram for the number of solutions per truth particle indicates that the total number of solutions varies significantly, by over 3 orders of magnitude. Also, this histogram suggests the optimal set of n^{div} parameters displayed in, what can be called the valley of optimal values. Then, the number of spurious solutions is minimized. Based on this histogram the optimal number of divisions in φ_0 has been set to 13 which translates into precision equal $\xi_{\varphi_0} = 0.001$ rad, and number of divisions in q/p_t equals to 8 which gives precision $\xi_{q/p_t} = 0.01\,\mathrm{GeV}^{-1}$.

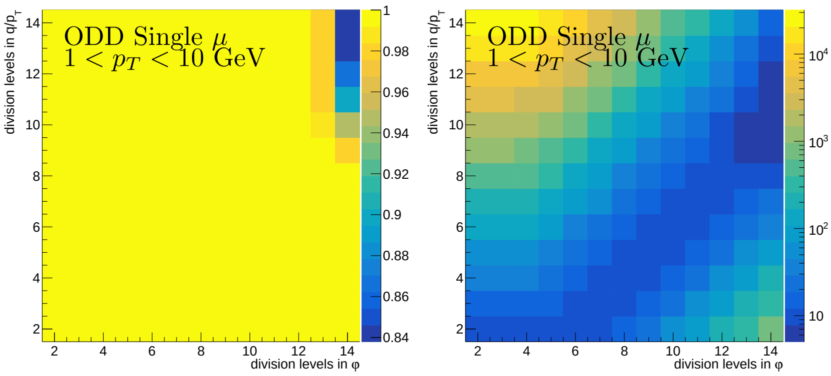
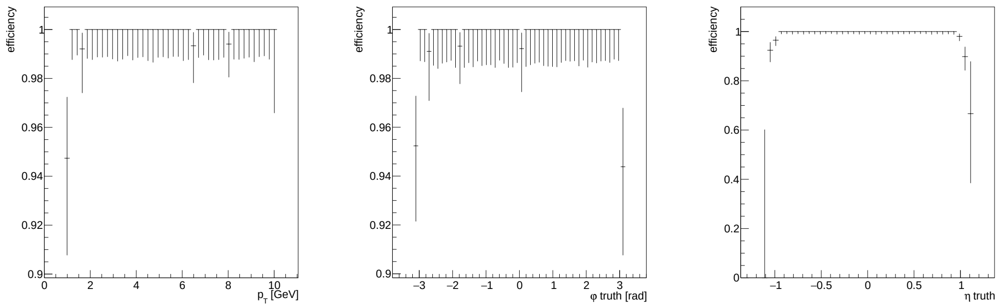


Figure 5.3: (left) Efficiently and (right) number of Hough solutions per a single truth particle, for the AHT for single μ events as a function of the number of divisions in each coordinate

5.2 Efficiency

Efficiency for a single particle event is defined as the ratio between the number of events with at least one Hough solution and the total number of events, that pass the requirement in the chapter 4, rejecting events with space-points in the barrel and disc regions. These requirements reduce the range of η for single μ events from $[-1.5, 1.5]$ to $[-1.08, 1.10]$.

The global efficiency equals 99.37%. Histograms of the efficiency as a function of transverse momentum, emission angle, and pseudorapidity are presented in figure 5.4. On efficiency expressed as a function of one of the parameters, additional cuts on values of other parameters have been applied to only account for variability caused by one of the parameters.



(a) Efficiently for p_t
$\varphi_0 \in [-3.0, 3.0]$,
$\eta \in [-1.0, 1.0]$

(b) Efficiently for φ_0
$p_t \in [1.5\,\mathrm{GeV}, 10\,\mathrm{GeV}]$,
$\eta \in [-1.0, 1.0]$

(c) Efficiently for η
$p_t \in [1.5\,\mathrm{GeV}, 10\,\mathrm{GeV}]$,
$\varphi_0 \in [-3.0, 3.0]$

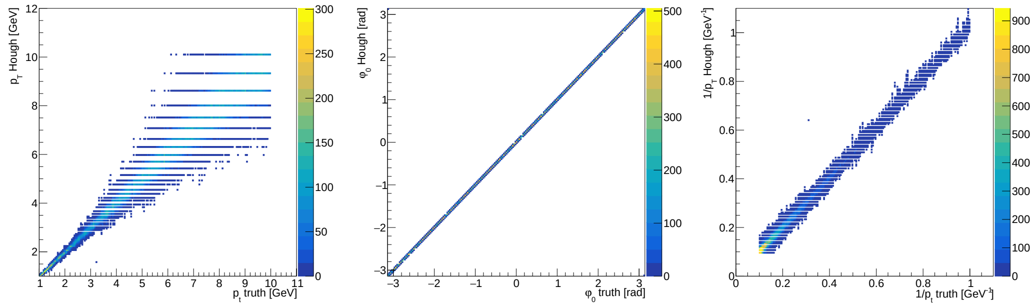
Figure 5.4: Efficiency for single μ events.

Efficiency is equal to 100% for the entire range in each parameter but for the value at the end of the range. A drop of the efficiency for $|\varphi_0| > 3.0\,\mathrm{rad}$ is caused by the periodic nature of that angle. This fact has been counted in the algorithm in the optimization steps. A decrease in the efficiency as a function of η is caused by preselection of the space–points requiring their location within the approximately $[-1.08, 1.10]$ range.

5.3 Scatter plot

A scatter plot is a useful tool for comparing parameters of truth particles and Hough solutions within one event. It enables verification of whether systematic deviation from the truth value does not occur for a certain range of values. Figure 5.5 presents scatter plot of p_t, φ_0 and $1/p_t$. Calculated values of φ_0 are closely correlated with the truth parameter. It is not however the case for p_t. A significant deviation is observed for larger transverse momenta. It stems from lower precision for high p_t values. The size of the solution candidate section after 8 divisions in q/p_t is equal to $7.81 \times 10^{-3}\,\mathrm{GeV}^{-1}$. For the section in the low p_t region of the accumulator (1 GeV) it translates to a width equal to $7.87 \times 10^{-3}\,\mathrm{GeV}$, for the high p_t region (10 GeV) width grows to 0.847 GeV. Thus precision as the function of q/p_t is constant in the entire accumulator region, and precision as a function of p_t decreases for higher transverse momenta. It can be observed in the figure 5.8a as a widened 'cloud' of the solution pair for higher p_t. This effect however is not visible for the scatter plot of $1/p_t$ – figure 5.8c. Increased thickness for that plot compared to 5.8b results from much finer accumulator granulation in φ_0 coordinate, precision in q/p_t equals 0.01, in φ_0 – 0.001.

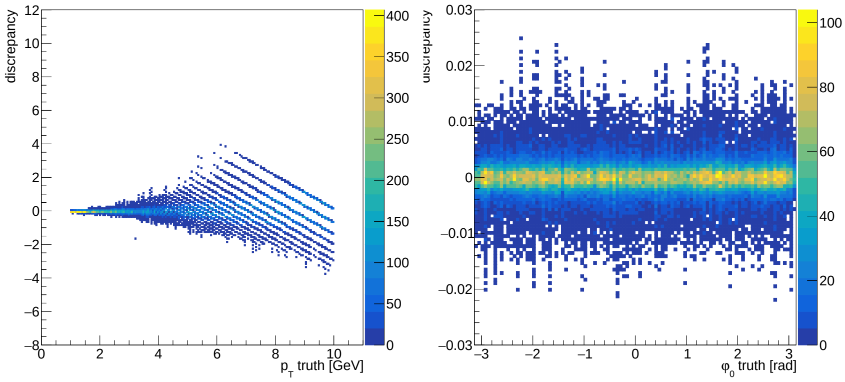
Further analysis of the precision of the particle tracking provides figure 5.6. It shows a discrepancy of a truth–Hough solution pair defined as $q/{p_t}^{Hough} - q/{p_t}^{truth}$ and as $\varphi_0^{Hough} - \varphi_0^{truth}$. These plots again show a very precise φ_0 angle estimation for the entire range, and a good fit in terms of p_t in the low transverse momentum



(a) Scatter plot for p_t. (b) Scatter plot for φ_0 (c) Scatter plot for η.

Figure 5.5: Parameters scatter plots for single μ events. Color scale indicates denisity of points.

region, with increasing discrepancy as momentum increases.



(a) Discrepancy between q/p_t^{Hough} and q/p_t^{truth}. (b) Discrepancy between φ_0^{Hough} and φ_0^{truth}

Figure 5.6: Discrepancies (relative difference) for accumulator variables for simulated single μ events. Color scale indicates denisity of points.

5.4 Number of solutions

The other parameter that quantifies the AHT applicability is the number of found solutions per single truth particle. We aim to obtain only a single solution for each truth particle. However, on average we get 9.92 Hough solutions. The distribution of the number of solutions presents figure 5.7. Further analysis of the number of solutions as a function of each of the parameters does not indicate that any range, in any parameter, yields, on average more solutions – figure 5.8.

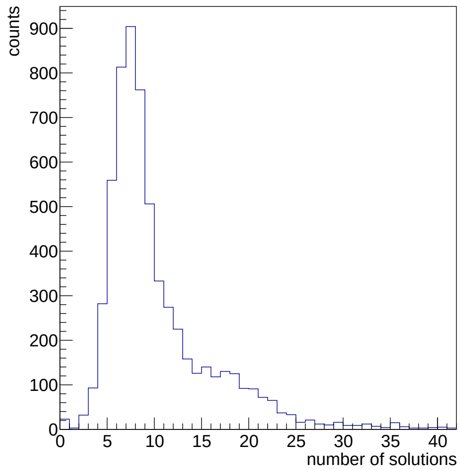
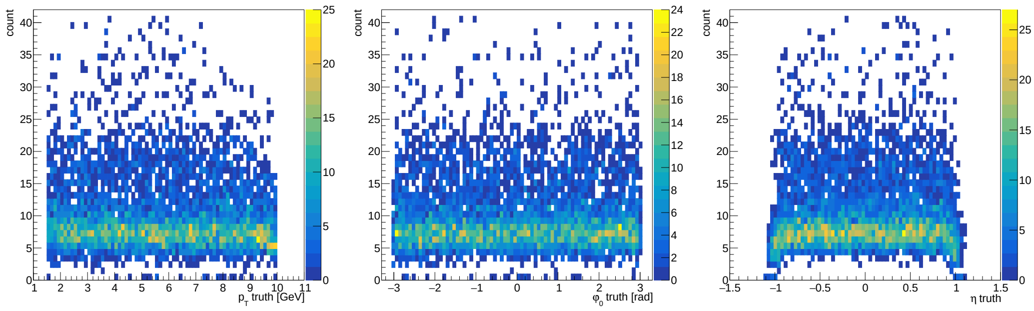


Figure 5.7: Distribution of the number of solutions per event for single μ events.



(a) Scatter plot of the number of solutions by p_t

(b) Scatter plot of the number of solutions by φ_0

(c) Scatter plot of the number of solutions by η

Figure 5.8: Scatter plots of the number of solutions for single μ events. The color scale indicate density of points.

Chapter 6

Efficiency and number of solutions optimization

When applied to the single μ events, the Hough transform reproduced the truth particle parameters with very high efficiency and precision. This indicates that the transform and the AHT algorithm can be effectively implemented for particle tracking in single–particle events. However, the most significant issue observed has been a significant fraction of spurious solutions. On average 9.92 Hough solutions were obtained for a single event. This problem is presented in the figure 6.1. The two figures depict the Hough solution for the same truth particle. When zoomed in, we can see that the solution candidates located closest to the truth particle have many line intersections within the section. Further cells include almost parallel lines, they do not cross. The purpose of the optimization algorithms has been to reduce the number of spurious Hough solutions, maintaining at the same time near 100% efficiency.

This chapter aims to describe and prove the fitness of the applied additional filtering algorithms for the AHT. These extra steps included:

- checking the number of line intersections within the cell by counting the number of line order changes on the boundaries of the section,
- peak finding calculated based on mean and standard deviation for line–crossing points
- data partitioning into wedges in φ_0 and η.

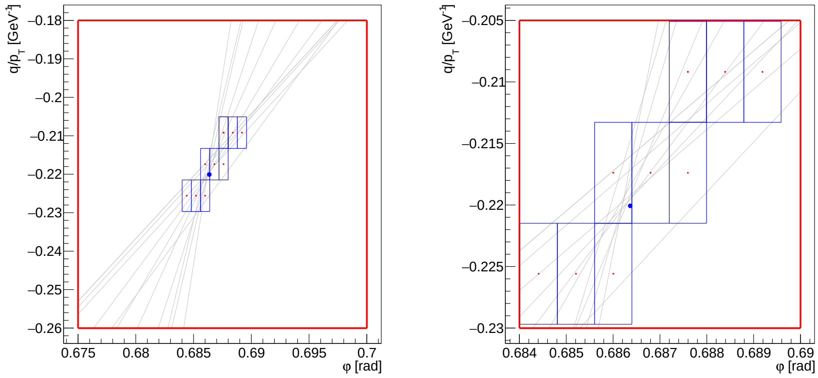
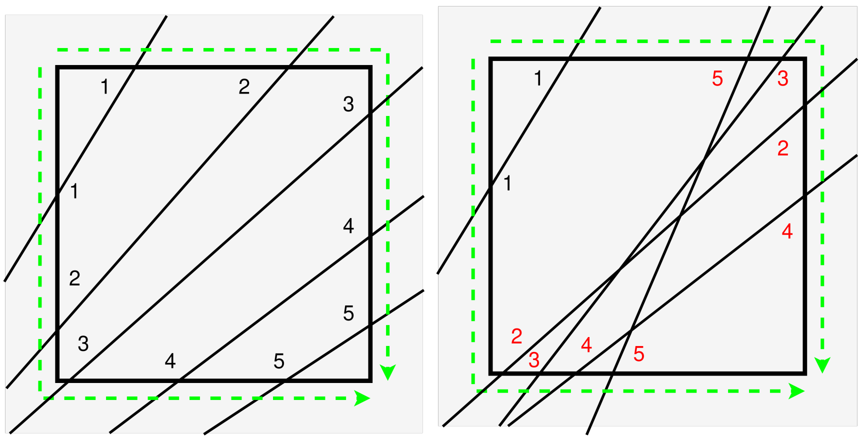


Figure 6.1: Section of the accumulator for simulated single μ event, $p_t = 4.54\,\mathrm{GeV}$, $\varphi_0 = 0.69\,\mathrm{rad}$, $q = -1$, $\eta = -0.27$.

6.1 Line order change

In the vicinity of the point corresponding to the truth particle parameters, multiple lines intersect which leads to several spurious solutions. As a result, with each truth particle, some duplicates are associated. Counting the number of lines order change imposes additional requirements on the solution candidates effectively reducing the number of solutions.



(a) Section rejected for further consideration.

(b) Section accepted for further divisions.

Figure 6.2: Graphical explanation of the line order changes filtering algorithm.

Figure 6.2 collates a section that will be rejected based on the line order change requirement (left) and a section that will be further divided (right). Ensuring that enough lines intersect within the cell could be executed by evaluating the point where each pair of lines crosses and making sure that the point lies within the section. However, the complexity of such an algorithm grows approximately proportionally to the square of the number of lines. This tool is not implemented at the final division step, just before adding the solution candidate to the solutions buffer, but approximately

3 division levels before that. It means that a significant number of lines will be present in the cell (especially for pile-up) which translates into unacceptably high computation time. Instead, an approach based on counting the number of line order changes has been adopted.

For every single line within the section, the position of the point where the line crosses the section boundary has been evaluated – $\mathfrak{w}$. The coordinates of these points (two for each line) are calculated along the cell boundary in the clockwise and counterclockwise direction starting from the upper left corner (`xBegin`, `yBegin + ySize`). Then for each direction, two arrays are generated, one with the calculated distances and the other one with the line ID, which in this case was the value of the `counter` – parameter which counts the number of lines that lie within a section. All the $\mathfrak{w}$ distances are normalized to the range $[0, 2]$, $\mathfrak{w} = 0$ if line passes through the point (`xBegin, yBegin + ySize`), $\mathfrak{w} = 2$ if the point (`xBegin + xSize, yBegin`) lies on the line.

Subsequently, the ID array has been sorted according to the $\mathfrak{w}$ distance in ascending order. If no lines cross within the section boundaries, the indices array will be identical. Any changes are due to the line crossing. In the last step, the number of non-identical IDs for the same index is counted. If fewer changes than the chosen threshold (equal to 3 for single μ events and pile-up) have been identified, the section is rejected. This algorithm targets mostly duplicate solutions which are defined as Hough solutions located near the truth particle parameters.

6.2 Peak finding

The peak finding algorithm relies on calculating the mean and standard deviation for φ_0 and q/p_t coordinates of each pair of intersection lines. Because in this filtering algorithm, we calculate intersection points for each pair of lines, this procedure is performed at the very last stage of the AHT, only for the solution candidate. This ensures a low number of lines present within the cell which makes the calculations sufficiently fast.

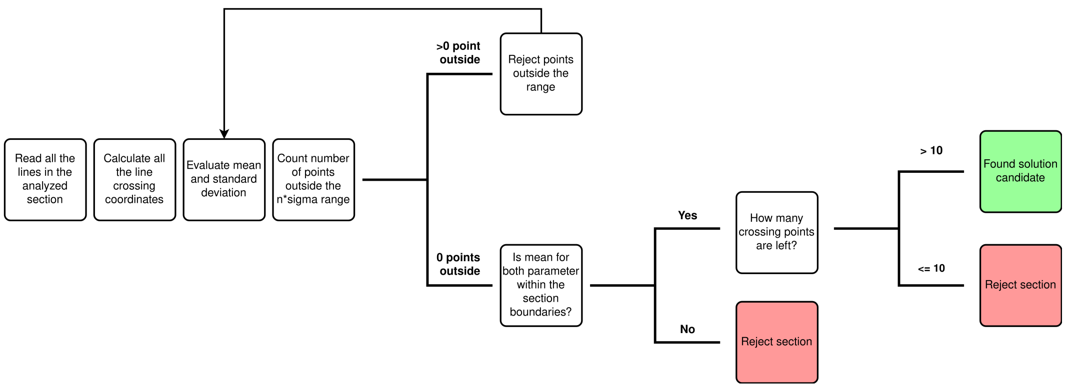


Figure 6.3: Block diagram of the peak finding optimization.

The procedure on which this optimization procedure was based, is presented in the block diagram in the figure 6.3. The steps incorporated in this procedure are as follows:

1. for the analyzed solution candidate access all the lines that cross the section,
2. calculates the intersection point for each pair of the lines, which is defined as $(\varphi_0^{\aleph}, q/p_t{}^{\aleph})$,
3. reject the intersection points $(\varphi_0^{\aleph}, q/p_t{}^{\aleph})$ which are located outside the accumulator, this step prevents including the intersection point of two nearly parallel lines which is located far from the analyzed section and strongly distorts the mean,
4. separately for $\varphi_0^{\aleph}$ and $q/p_t{}^{\aleph}$ evaluate mean $(\mu^{\aleph})$ and standard deviation $(\sigma^{\aleph})$ for the entire set of points,
5. make sure all the points used to calculate the basic statistics lie within the $\mu^{\aleph}_{\varphi_0} \pm n_{\aleph} \cdot \sigma^{\aleph}_{\varphi_0}$ and $\mu^{\aleph}_{q/p_t} \pm n_{\aleph} \cdot \sigma^{\aleph}_{q/p_t}$ ranges, where $n_{\aleph} = 2.0$ for single μ and pile-up events,
6. all the identified points outside the range remove from the initial $\varphi_0^{\aleph}$ and $q/p_t^{\aleph}$ arrays,
7. if more than zero points were rejected, repeat procedure described in the points $\mathtt{2} - \mathtt{7}$,
8. once all the points are within the defined range we can proceed to the proper solution candidate classification, if either mean is outside the appropriate range defined as $(\mathtt{xBegin} - \text{ע} \cdot \sigma^{\aleph}_{\varphi_0}, \mathtt{xBegin} + \mathtt{xSize} + \text{ע} \cdot \sigma^{\aleph}_{\varphi_0})$ and $(\mathtt{yBegin} - \text{ע} \cdot \sigma^{\aleph}_{q/p_t}, \mathtt{yBegin} + \mathtt{ySize} + \text{ע} \cdot \sigma^{\aleph}_{q/p_t})$ a solution candidate is rejected. These ranges are section boundaries extended in x coordinate by $\pm\text{ע} \cdot \sigma^{\aleph}_{\varphi_0}$ and $\pm\text{ע} \cdot \sigma^{\aleph}_{q/p_t}$ in y coordinate. Widened acceptance range allowed to retain very high efficiency, ע parameter equals 0.5 for both single μ and pile-up events,
9. ensure that at least 10 points comprise the final intersection point array, if fewer lines are left reject the solution candidate, this requirement applies only for pile-up events,
10. if the section passed both criteria (for mean location and number of points) add the cell to the solutions buffer.

The algorithm designed in the following way is capable of accounting for lines in the section that do not intersect with the remaining lines. Such a line, especially in the pile-up, even though contributes to meeting the criterion of a minimal number of lines in the section, should be discarded since it is not in any way correlated with the same particle trajectory as the lines corresponding to the remaining space-points. In pile-up events, this algorithm addresses the issue of the 'lost' lines, which are Hough transformed hits coordinates into lines that originate from another trajectory. In single μ events, it further reduces the number of solutions adjacent to the truth particle by imposing stricter requirements on the solution candidate.

An example of the approach implementation is graphically presented in the figure 6.4. Left plot picture lines present in the single solution candidate. The five green lines possibly generate a Hough solution, however, the red line most likely is a 'lost' line originating from another trajectory. So that the cell is eventually verified, the peak finding algorithm has been performed which is depicted in the schematic right plot. Green dots derive from the intersection points of the green lines, and red from the additional line. The latter ones were rejected since they are located outside the inner square. Because the remaining dots are closely clustered, likely no further points will be discarded.

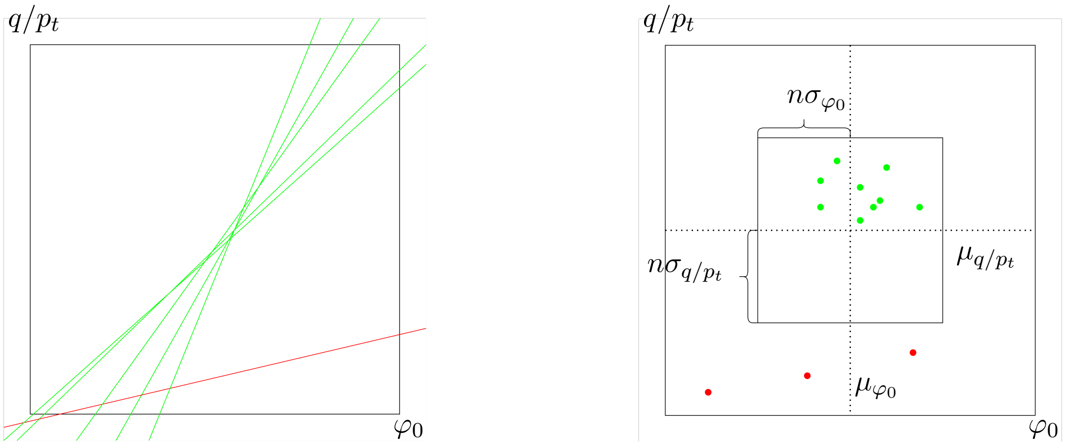


Figure 6.4: An illustration for the peak finding algorithm. In both drawing the accumulator space is shown, on the left the measurements lines and on the right their intersection points.

6.3 Division in φ_0 and η

The last implemented tool for improved particle tracking was splitting all the space-points into regions (wedges) defined by φ_0 angle and η. Each region is defined by the center of the wedge in both coordinates (η^{wedge} and φ_0^{wedge}) as well as the width of the wedge – $\Delta\eta^{wedge}$ and $\Delta\varphi_0^{wedge}$. Width is defined as the distance from the center of the wedge to either boundary. A single wedge is comprised of the space-points that meet the following inequalities: $|\eta^{SP} - \eta^{wedge}| < \Delta\eta^{wedge}$, $|\varphi^{SP} - \varphi_0^{wedge}| < \Delta\varphi_0^{wedge}$, where η^{SP} and φ^{SP} are values calculated for space-point. These constraints on the analyzed set of space-points return geometry presented in the figure 6.5.

Parameters of such a wedge are defined by range of φ_0 and η ($\Delta\varphi_0 = \varphi_0^{max} - \varphi_0^{min}$, $\Delta\eta = \eta^{max} - \eta^{min}$) and number of wedges in both coordinates – $N_{\varphi_0}^{div}$, N_{η}^{div}. Then width of each region is identical and equal to

$$\Delta\varphi_0^{wedge} = \frac{\Delta\varphi_0}{2 \cdot N_{\varphi_0}^{div}}, \tag{6.1a}$$

$$\Delta\eta^{wedge} = \frac{\Delta\eta}{2 \cdot N_{\eta}^{div}}. \tag{6.1b}$$

The center of the wedge (φ_0^{wedge}, η^{wedge}) equals

$$\varphi_0^{wedge} = \varphi_0^{min} + \Delta\varphi_0^{wedge} + 2 \cdot \Delta\varphi_0^{wedge} \cdot i, \quad \text{for} \quad i = 0, 1, 2, \ldots, N_{\varphi_0}^{div} - 1, \tag{6.2a}$$

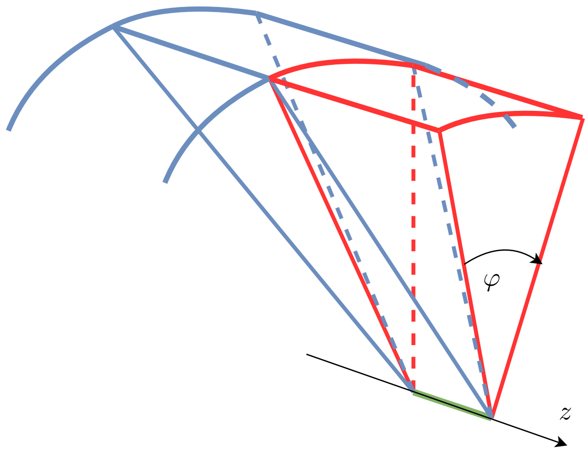


Figure 6.5: Schematic picture of the detector volume division used for the data partitioning.

$$\eta^{wedge} = \eta^{min} + \Delta\eta^{wedge} + 2 \cdot \Delta\eta^{wedge} \cdot j, \quad \text{for} \quad j = 0, 1, 2, \ldots, N_{\eta}^{div} - 1. \tag{6.2b}$$

The idea of the data partitioning is to separate a smaller subsection of the detector volume and to analyze a narrower set of space-point at once. That way, especially in the pile-up events we are less susceptible to the 'lost' lines which disproportionately inflate the total number of solutions. Furthermore, splitting the initial data into sets of hits allows us to simultaneously process the input data and search for the solution. Thus, data partitioning is a powerful tool for the parallelization of the computation process which was incorporated into CPU–based calculations.

To guarantee this approach's effectiveness, for each truth particle, there must be at least one wedge that encompasses all the hits associated with that particle trajectory. And the underlying assumption of this procedure is that when analyzing only a single wedge we get access to all the space-points providing information about the trajectory. However, this is not the case for such a simple definition of the regions. A new parameter has to be introduced – excess width in both coordinates ($\beth_{\varphi_0}$, $\beth_{\eta}$). It is defined as an overlap coefficient that governs what fraction of the total pool of the space-points will be analyzed in two adjacent regions. Effectively this parameter simply increases the width. Hence the definition of the wedge width in the equations 6.1a and 6.1b is altered into the following form

$$\Delta\varphi_0^{wedge} = \frac{\Delta\varphi_0}{2 \cdot N_{\varphi_0}^{div}} + \beth_{\varphi_0}, \tag{6.3a}$$

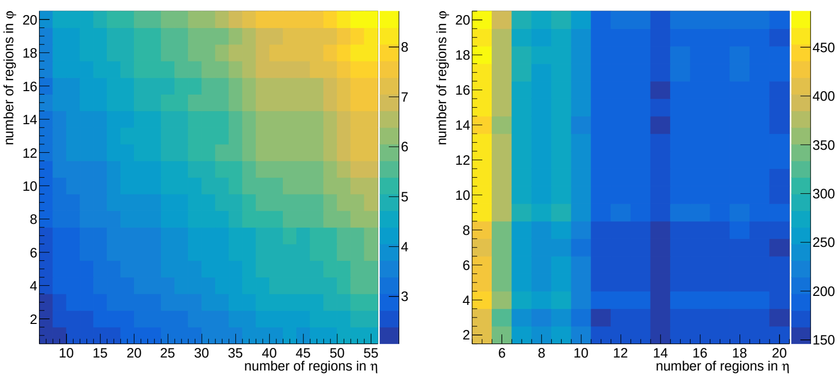
$$\Delta\eta^{wedge} = \frac{\Delta\eta}{2 \cdot N_{\eta}^{div}} + \beth_{\eta}. \tag{6.3b}$$

The necessity to adopt the excess width stems mostly from the fact that trajectories in the xy plane projection form a curve that cannot be easily constrained to a single wedge.

This optimization technique turned out to be crucial for the tracking of pile-up events, it greatly reduced computation time as well as the number of solutions. This approach was however redundant for single μ events. Its implementation led to an increased number of solutions.

The efficiency of particle tracking when data partitioning is involved, is a function of the number of divisions and the excess wedge width. First, the optimal number of divisions has been determined, and for this configuration, the excess width parameter ($\beth_{\varphi_0}$ and $\beth_{\eta}$) minimizing total number of solutions has been tailored.

To decide on the most suitable number of divisions number of Hough solutions has been calculated for a wide range of configurable division parameters. The results are presented in the figure 6.6. The two histograms present how the average number of solutions for a single truth particle changes with an altered number of divisions.



(a) Single μ events.

(b) Pile-up events.

Figure 6.6: Scatter plot of the number of solutions as a function of the number of divisions in φ_0 and η, each rectangle corresponds to one set of initial parameters.

For a single μ event we do not see a very strong dependence of the average number of solutions on the number of divisions. Progressively and roughly linearly, the number of solutions increases with a growing number of divisions. Relationship for pileup events is more complex and interesting. First of all the number of solutions marginally depends on the $N^{div.}_{\varphi_0}$, parameter of much greater importance is $N^{div.}_{\eta}$. The number of solutions is very high for a few divisions in η – approximately 450 and rapidly decreases as $N^{div.}_{\eta}$ grows, at last stabilizing at below 200. Based on these results the division parameters were equal to $N^{div.}_{\varphi_0} = 8$ and $N^{div.}_{\varphi_0} = 15$.

Such a choice of parameters might seem unjustified since we do not get a reduction in the number of solutions when divisions in φ_0 are introduced or when more than 11 divisions in η are performed. However, such a choice arises also from the constraints allowing to parallelize the calculations. That is the data for all measurements in a given region need to fit in a limited group (in `SYCL` terminology) space.

For the chosen $N^{div.}_{\varphi_0}$ and $N^{div.}_{\varphi_0}$, optimal value of the excess width parameter – $\beth_{\varphi_0}$ and $\beth_{\eta}$ has been evaluated. It was set to the minimal value which ensures sufficiently high efficiency. That parameter for both dimensions is the same and equal to 0.12.

6.4 Performance of the optimization algorithms

The ultimate goal of these optimization procedures has been to reduce the average number of Hough solutions for a single truth particle and to retain high efficiency. The efficiency in the single μ events calculated for the entire dataset of $10,000$ particles, when no additional solution filtering methods were applied equals 99.34%, for simple line order changes requirement – 99.30% and for peak finding as well – 99.16%. So we observe a very low decrease in the tracking efficiency and the number of solutions for a single muon particle dropped from 9.9 when no filtering algorithms were applied to merely 1.8 when performing peak finding and counting the number of line swaps – figure 6.7. Such a significant improvement in both of these parameters evaluating tracking quality indicates that the optimization algorithms have been properly defined and used parameters likely are the most optimal values.

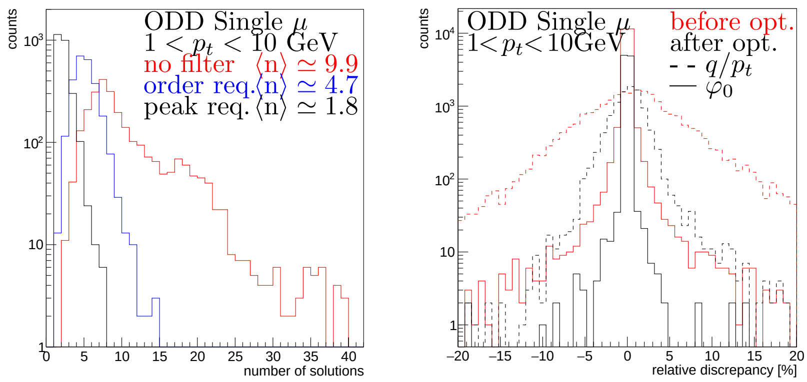


Figure 6.7: Optimization assessment of single μ events, (left) histogram of the number of solutions for different sets of filtering tools, (right) relative discrepancy of q/p_t and φ_0 defined as the difference between Hough solution and truth parameter.

Furthermore, from figure 6.7 we can conclude that optimization steps decreased discrepancy defined as the difference between the Hough solution and truth parameters. So that we at the same time succeeded in improving the precision of the particle parameter estimate.

Chapter 7

Performance for pile–up

The accumulator view depicted in figure 5.1 for single μ events gets noticeably more complicated for events with pile-up, a single event is comprised of up to $30,000$ lines instead of less than 10 for a single μ. Such a high density of lines resulted in populating the solutions stack with almost every single possible section. For precision in φ_0 equal 0.001 rad, and precision in q/p_t equal 0.01 GeV^{-1}, there are almost $2,100,000$ sections which can be regarded as solution candidates out of which about $1,800,000$ qualify as solutions when no filtering methods were employed. So almost every possible set of φ_0 and q/p_t has been labeled as a solution which is not particularly useful for the pattern recognition stage of the tracking.

Also, since the pile-up sample reassembles closely the experimental reality, truth particles are not uniformly distributed in the p_t. Most detected particles are located in the low transverse momentum region of the accumulator. It further complicates calculations since the Hough solutions for $q/p_t \approx \pm 1$ are more likely to overlap.

To address this issue the data partitioning into wedges has been implemented. The entire range of $\eta - [-1.5, 1.5]$, and $\varphi_0 - [-\pi, \pi]$ has been divided into respectively 15 and 8 regions. To skip the first few division steps of the AHT the initial range of the accumulator in φ_0 has been narrowed from $[-\pi, \pi]$ to $[\varphi_0^{wedge} - \Delta\varphi_0^{wedge}, \varphi_0^{wedge} + \Delta\varphi_0^{wedge}]$. This way areas of the accumulator that cannot produce viable solution candidates are not considered from the start.

An example of the accumulator filled with lines corresponding to space-points located in a single wedge is presented in the figure 7.1. These space-points were classified to the wedge with center and width parameters as follows: $\varphi_0^{wedge} = 1.178$ rad, $\Delta\varphi_0^{wedge} = 0.513$ rad, $\eta^{wedge} = -1.00$, $\Delta\eta^{wedge} = 0.22$. The inner rectangle sets the new boundaries for the accumulator, the initial section is identical to that rectangle. Figure 4.3 shows the same set of lines but only the 'active' area of the accumulator is displayed.

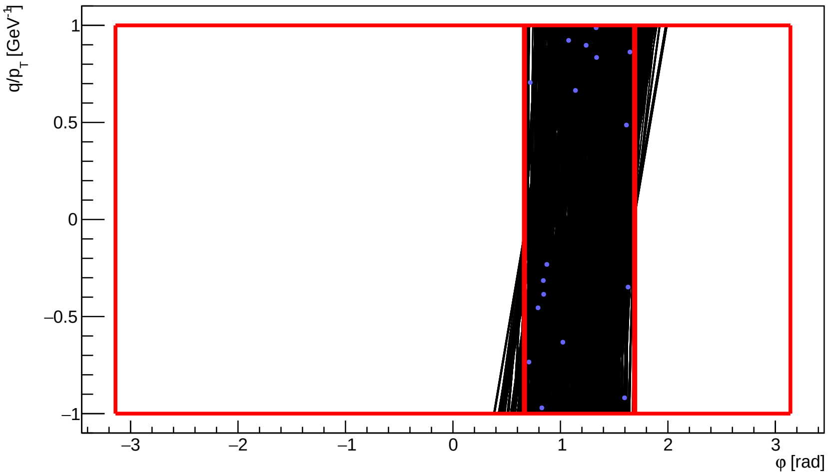


Figure 7.1: Accumulator filled with lines corresponding to a single wedge, inner red rectangle denotes the size of the accumulator, blue dots represent truth particle parameters located in that wedge.

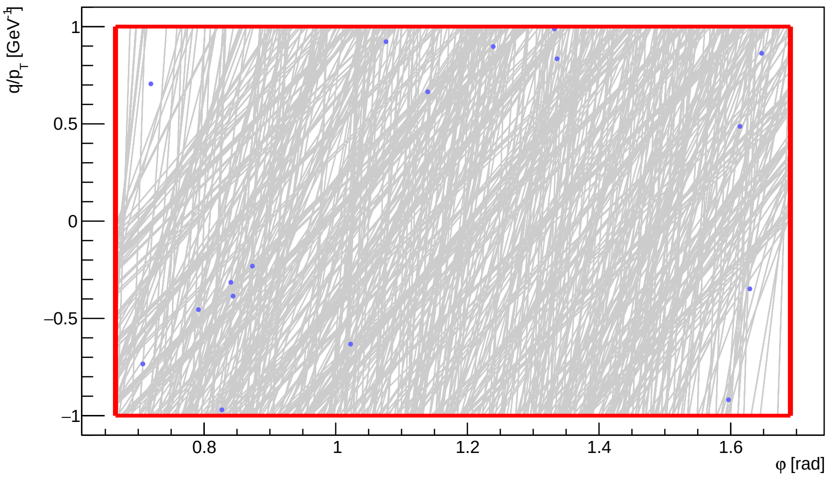


Figure 7.2: Accumulator filled with lines corresponding to a single wedge. The blue dots indicate truth solutions.

The same part of the accumulator has also been plotted but with obtained Hough solutions – figure 7.3. Red dots correspond to these solutions. The horizontal strip located alongside the line $q/p_t = 0$ where no Hough solutions are present stems from the fact that if the p_t for the solution candidate is greater than 10.5 GeV, the section is not added to the solution buffer. This requirement prevents regarding as a solution a particle with the transverse momentum outside of the $[1\,\mathrm{GeV}, 10\,\mathrm{GeV}]$ range. And because we aim at parameterizing particles in this momentum range, this requirement is justified.

Here again, on a much larger scale, we can observe the issue of the spurious

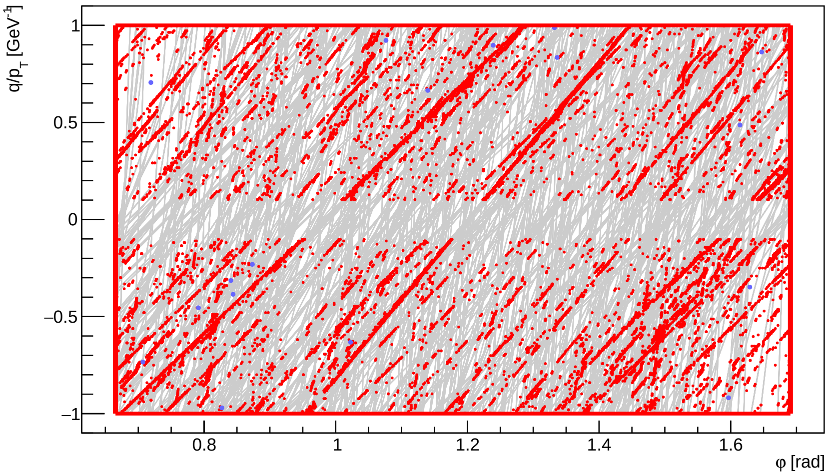


Figure 7.3: Accumulator filled with (gray) lines corresponding to a single wedge, blue dots represent truth particle located in that wedge and red dots Hough solutions.

events like had been depicted in the figure 6.1. This time, however, the scale of this effect is by a magnitude stronger. Around 200 Hough solutions were found for each truth particle. In the case of single μ events all spurious solutions can be named as duplicates, they are located near the truth parameters. It is not the case for pile-up, most solutions can be labeled as fakes, which means that they do not occur near the truth particle parameters due to higher line density close to the true solutions. Most of the Hough solutions are present because of the high number of 'lost' lines.

7.1 Efficiency

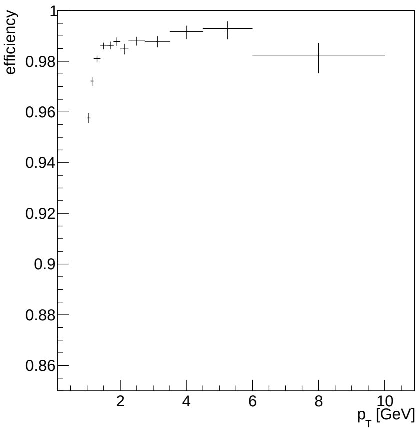
Truth particle for single events is deemed as reconstructed when at least one solution was found for the given event. For pile-up, particle (η^{truth}, φ_0^{truth}, $q/p_t{}^{truth}$) is considered found if at least one Hough solution (η^{Hough}, φ_0^{Hough}, $q/p_t{}^{Hough}$) satisfies all of the inequalities

- $|\eta^{truth} - \eta^{Hough}| < 1.2 \cdot \Delta\eta^{wedge}$,
- $|\varphi_0^{truth} - \varphi_0^{Hough}| < 0.05$,
- $|q/p_t{}^{truth} - q/p_t{}^{Hough}| < 0.1$.

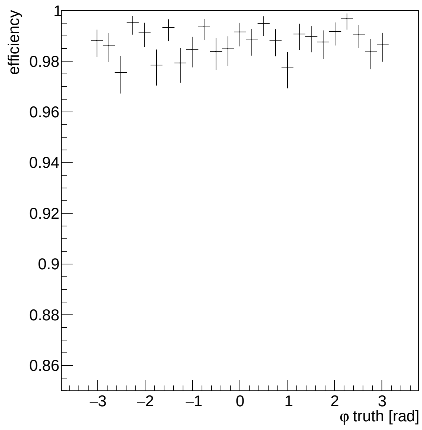
All the results presented in this chapter are for results acquired from the divided space-points into wedges. For results obtained without additional filtering (counting the number of line order changes and peak finding), the efficiency equals 99.1%. Percentage of fake solutions was 62.2%, of duplicates – 37.6%.

To investigate efficiency more thoroughly, efficiency has been presented as the function of η, φ_0, and p_t – figure 7.4.

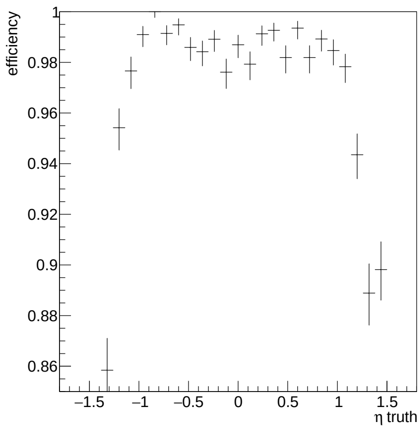
For pileup events, efficiency expressed as a function of φ_0 is very high for the entire range. We do not observe here, contrary to single μ events, wedge effect



(a) Efficiency by p_t



(b) Efficiency by φ_0



(c) Efficiency by η

Figure 7.4: Efficiency as a function of the truth particle parameters. The range of the parameters which is not on the abscissa has been limited to the range $\varphi_0 \in [-\pi, \pi]$, $p_t \in [2\,\text{GeV}, 7\,\text{GeV}]$, $\eta \in [-1.1, 1.1]$ so that variability in efficiency caused by another parameter does not influence the efficiency plot.

which dragged down efficiency for $\varphi_0 \approx \pm\pi$. Efficiency for η is noticeably reduced for pseudorapidity outside the range $[-1.0, 1.0]$. This effect is partially caused by rejecting space-points outside the $[-1.5, 1.5]$ which is the range of variety we are interested in. In addition, drop for η around $[-1.5, -1.0]$ and $[1.0, 1.5]$ is a feature of the detector and is observed for other tracking algorithms as well. Efficiency beyond $|\eta| = 1.5$ should recover once the signals from all detectors are used.

An interesting property of the efficiency is displaced on the plot of the reconstruction efficiency against transverse momentum. High performance is observed only for the range roughly from 2 GeV to 7 GeV and drop occurs below and above this range. A low p_t decrease in efficiency might be caused by too narrow wedges in φ_0 angle. The curvature of particle trajectory increases for lower transverse momenta, thus the excess width parameter ($\beth_{\varphi_0}$) might be insufficient to cover all the space-points.

Decrease for high p_t should be further investigated due to very low statistics.

7.2 Number of solutions

Introducing filtering algorithms in pile-up events reduced the tracking efficiency by around 6 percentage points, which will be acceptable if we managed to considerably decrease the average number of Hough solutions for truth particles. A comparison of the number of solutions distribution for various sets of filtering tools applied is presented in figure 7.5.

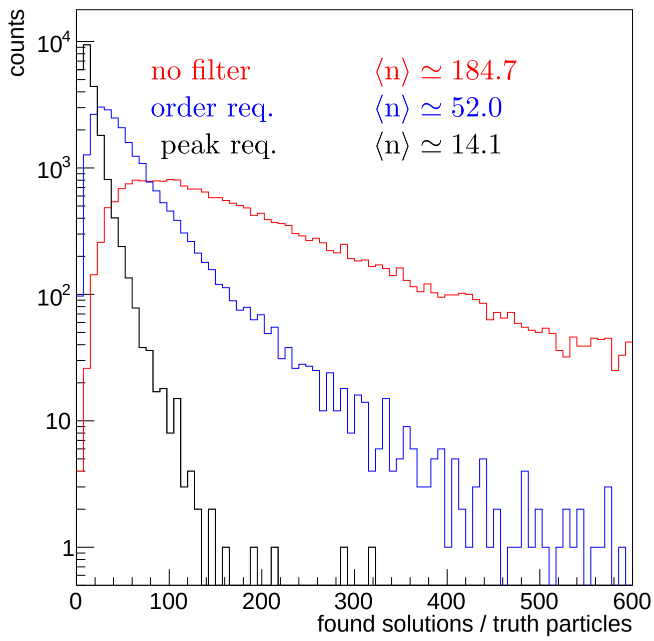


Figure 7.5: Comparison of the number of solutions

This optimization of the AHT led to a noticeable decrease in the average number of obtained solutions for a single truth particle. From an initial 184.7 to 14.1 when all the filtering algorithms were implemented.

Chapter 8

Results comparison to the reference calculations

The last stage in the analysis of the Hough transform in general, as well as AHT, its overall performance, efficiency, and number of solutions produced in comparison to the benchmarks calculations. The results for the particle tracking in High Luminosity LHC of the Inner Tracker Upgrade for Phase II have been used as the benchmark calculations [14].

8.1 Single μ events

The reference tracking reconstruction as the function of η for the single μ events is shown in figure 8.1. For this calculation, efficiency is equal to 100% just like for the AHT – 5.4. The obtained results are in full agreement with the benchmark results.

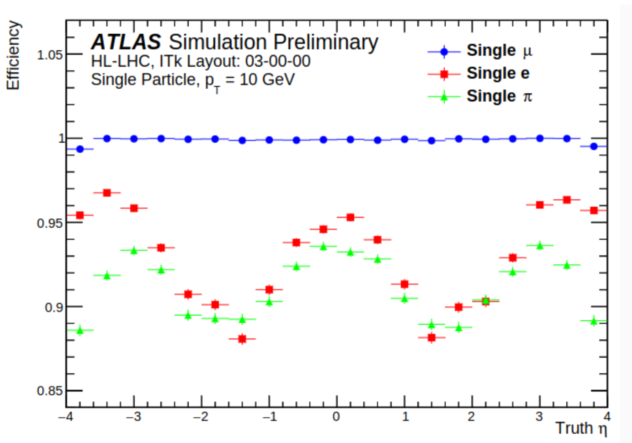


Figure 8.1: The reference tracking efficiency for single μ events [14].

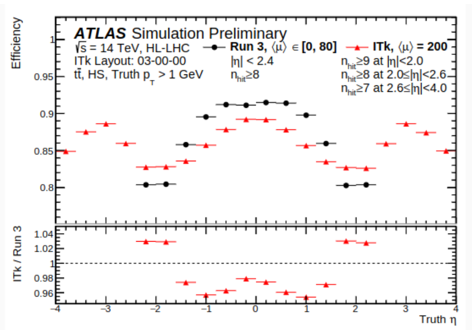
8.2 Pile-up events

The efficiency results as well as the number of solutions by the number of interactions for pile-up are presented in the figure 8.2. Efficiency follows similar changes like in

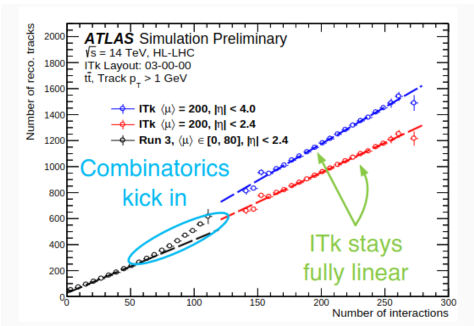
AHT, high efficiency near $\eta = 0$, with a gradual drop for higher absolute values of pseudorapidity. The AHT pile-up efficiency for particles produced in the xy plane is above 98%, for the reference calculations, it is less than 90%. The benchmark efficiency for $|\eta| = 1.5$ is around 87%, and for the Hough transform is around 84%.

A certain discrepancy is observed also for for the total number of solutions – 8.2b. In the pile-up region which is the scope of that analysis – the number of interactions in each event ≈ 200, for the reference calculations approximately 900 track were reconstructed (for particles with transverse momentum $> 1\,\text{GeV}$). In the tracking implementing the Hough transform for each event approximately $6,500$ tracks were parametrized.

From the comparison of the reference and AHT results, it can be concluded that the Hough transform efficiency is higher for $\eta \approx 0$ and lower than the benchmark value for higher absolute values of pseudorapidity. The drop can be explained by narrowing the η range which caused rejecting some of the space-points from the consideration. The increased efficiency is offset by a significantly higher number of the reconstructed tracks. However, it should be added, that the reference to which the comparison is made is a complete tracking chain with fitting and ambiguity resolution and the only relevant parameter in this case is the efficiency.



(a) Reference efficiency for pile-up events [14].



(b) Reference number of solutions for pile-up events [14].

Figure 8.2: Reference parameters of tracking for pile-up events.

Chapter 9

Summary

The scope of the thesis was to present the application and performance of the adaptive Hough transform, as a pattern-finding stage of particle tracking

Efficiency as well as the number of solutions has been evaluated for single μ and pile-up events. Pile-up dataset aimed at modeling the detector response for high data volume that is expected to be gathered by the ATLAS experiment at the High Luminosity Large Hadron Collider. Each event contained data generated when on average 200 proton–proton interaction occur at each bunch crossing. Single μ events were used to optimize the parameters of the AHT.

Solutions obtained from the single μ events indicated very high tracing efficiency – 99.34%, for precision in φ_0 and q/p_t equal respectively 0.001 rad and 0.01 GeV^{-1}. A high number of spurious Hough solutions was found for the naive HT implementation. For each truth particle around 9.9 Hough solutions have been identified.

To address this issue additional steps of solution filtering have been implemented. These include counting the number of lines crossing, peak location for lines intersection points, and splitting the detector data into wedges. These tools, when configured with the optimal parameters significantly reduced the number of solutions without affecting efficiency significantly. When including requirements for line crossing and peak finding, the number of solutions for a single μ particle dropped to 1.8 with efficiency equals 99.16%.

When the optimization steps described in the chapter 6 were implemented, efficiency dropped to 93.2%. It also altered the percentage of fake and duplicate spurious solutions, now this fraction equals respectively 38.2% and 64.8%. So we can conclude that the implemented filtering algorithms are particularly effective in reducing the number of fake solutions. Additional measures could be introduced to focus more on duplicates.

Once the algorithm was proven effective for single μ events pile-up dataset was analyzed. To reduce the number of Hough solutions and introduce parallel computation space-points were divided into wedges. The entire detector volume has been divided into 8 regions in φ_0 and 15 regions in η. The number of solutions per true particle in the $1\,\mathrm{GeV} < p_t < 10\,\mathrm{GeV}$ range was 185 with 99.1% efficiency. Introducing solutions filtering reduces the number of solutions per true particle to 14.1 and efficiency to 93.2%.

After analysis of all the described calculations, it can be concluded that AHT is an effective method for pattern pattern-finding stage of particle tracking. In a single particle event, it returned trajectories parameterized with nearly 100% efficiency and merely 1.8 solutions for a single truth particle. Also, very high efficiency was obtained for pile-up, in the range $[-1.0, 1.0]$ it exceeded the reference value. All three optimization procedures are very effective in reducing the number of events and at the same time retaining high efficiency.

Still, the spurious Hough solutions should further be reduced which will be an objective of this project in the following stages. Also, effort will be made to evaluate the performance of the algorithm using GPU and CPU (with SYCL), extend the accumulator into the impact parameter – d_0, apply and tune the AHT to ITk simulation data, and at last, integrate the algorithm with `Atena` or `ACTS`.

Bibliography

[1] Lyndon Evans and Philip Bryant. LHC Machine. *Journal of Instrumentation*, 3(08):S08001, aug 2008. https://dx.doi.org/10.1088/1748-0221/3/08/S08001.

[2] ATLAS Collaboration. ATLAS HL-LHC Computing Conceptual Design Report. *CERN Document Server: CERN-LHCC-2020-015, LHCC-G-178*, 2020. https://cds.cern.ch/record/2729668.

[3] ATLAS Collaboration. Studies of the performance of the atlas detector using cosmic-ray muons. *CERN Document Server: CERN-PH-EP-2010-070*, 71:1593, 2011. https://cds.cern.ch/record/1311111.

[4] S Baird. Accelerators for pedestrians; rev. version. *CERN Document Server: AB-Note-2007-014, CERN-AB-Note-2007-014, PS-OP-Note-95-17-Rev-2, CERN-PS-OP-Note-95-17-Rev-2*, 2007. https://cds.cern.ch/record/1017689.

[5] CERN. CERN's accelerator complex. https://home.cern/science/accelerators/accelerator-complex.

[6] ATLAS Colaboration. ATLAS Fact Sheet. http://cds.cern.ch/record/1457044, 2010.

[7] M. Beckingham, M. Duehrssen, E. Schmidt, M. Shapiro, M. Venturi, J. Virzi, I. Vivarelli, M. Werner, S. Yamamoto, T. Yamanaka on on behalf of ATLAS Collaboration. The simulation principle and performance of the ATLAS fast calorimeter simulation FastCaloSim. *CERN Document Server: ATL-PHYS-PUB-2010-013*, 2010. https://cds.cern.ch/record/1300517.

[8] ATLAS Collaboration. Atlas inner detector: Technical design report, 1. 1997. https://cds.cern.ch/record/331063.

[9] ATLAS Collaboration. Atlas pixel detector readies to tackle run 3. 2022. https://atlas.cern/Updates/Briefing/ATLAS-Pixel-Detector-Run3.

[10] Pernegger, Heinz on behalf of ATLAS Collaboration. The Pixel Detector of the ATLAS Experiment for LHC Run-2. *CERN Document Server: ATL-INDET-PROC-2015-001*, (06), 2015. https://cds.cern.ch/record/1985432.

[11] Heggelund, Andreas Lokken on behalf of ATLAS Collaboration. Overview of the ATLAS ITk Pixel Detector. *CERN Document Server: ATL-ITK-PROC-2022-014*, (02), 2023. http://cds.cern.ch/record/2834491.

[12] ATLAS Collaboration. Operation and performance of the atlas semiconductor tracker. *Journal of Instrumentation*, 9(08):P08009, aug 2014. https://dx.doi.org/10.1088/1748-0221/9/08/P08009.

[13] Sebastien Roy-Garand. ATLAS Inner TracKer Upgrade, 2023. https://cds.cern.ch/record/2869708.

[14] ATLAS Collaboration. Expected tracking and related performance with the updated ATLAS Inner Tracker layout at the High-Luminosity LHC. *CERN Document Server: ATL-PHYS-PUB-2021-024*, 2021. https://cds.cern.ch/record/2776651.

[15] T G Cornelissen. Track Fitting in the ATLAS Experiment, 2006. Presented on 12 Dec 2006, https://cds.cern.ch/record/1005181.

[16] R. Fisher, S. Perkins, A. Walker and E. Wolfart. Hough transform, 2003. https://homepages.inf.ed.ac.uk/rbf/HIPR2/hough.htm.

[17] Oxford Instruments. Electron backscatter diffraction (ebsd) pattern formation. https://www.ebsd.com/ebsd-explained/pattern-formation.

[18] P. V. C. Hough. Machine Analysis of Bubble Chamber Pictures. *Conf. Proc. C*, 590914:554–558, 1959. https://inspirehep.net/literature/919922.

[19] Richard O. Duda and Peter E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Commun. ACM*, 15(1):11–15, jan 1972. https://doi.org/10.1145/361237.361242.

[20] Zbigniew Kąkol, Jan Żukrowski. Ruch naładowanych cząstek w polu magnetycznym. https://home.agh.edu.pl/~kakol/efizyka/w22/main22c.html.

[21] Piotr Libucha. Development of Hough transform implementation for helix finding on graphic cards processors, 2023.

[22] Corentin Allaire et al. OpenDataDetector. apr 2022. https://doi.org/10.5281/zenodo.6445359.

[23] Xiaocong Ai et al. A Common Tracking Software Project. *Comput. Softw. Big Sci.*, 6(1):8, 2022. https://inspirehep.net/files/bf090262a22f124c10cde91dc5c0f5ad.