AGH UNIVERSITY OF SCIENCE AND TECHNOLOGY
Faculty of Physics and Applied Computer Science

Department of Particle Interactions and Detection Techniques

Bachelor of Science Thesis

Extention of the ROOT data structure describing proton-proton collisions at STAR experiment with secondary vertices

Rozszerzenie struktury danych opisujacych zderzenia proton-proton w eksperymencie STAR o informacje o wierzchołkach wtórnych

Author: Przemysław Ryś
Major: technical physics
Supervisor: dr inż. Leszek Adamczyk

Krakow, January 2024

Abstract

The following work is focused on the analysis of the topology of the decay of short-lived strange particles and on the development of a program written in the ROOT framework, the purpose of which was to add information relevant for measurement of neutral kaons (K^0_S), Λ's and $\bar{\Lambda}$'s to the data samples commonly used for physics analysis. Moreover, histograms of the simulation-generated distributions with an appropriate selections of topological information for the tracks of the analyzed particle pairs were made. For that purpose dedicated class, which was written by other STAR members, was used. They were also presented using the method proposed by R. Armenteros and J. Podolanski, which is an interesting alternative for identification of V^0-particles using only information about the momenta of the daughters without any assumptions about masses of the V^0-particle and its daughters.

Contents

1 Introduction **1**

1.1 STAR experiment . . . 2
1.2 Quark-gluon plasma . . . 5
1.3 Ultra-peripheral collisions . . . 7
1.4 Inelastic proton-proton scattering . . . 8
1.5 Production and decay of strange particles . . . 10
1.6 Concep of the Armenteros-Podolanski plot . . . 12

2 Description of the code **13**

3 Analysis **16**

3.1 DCA of the V^0 to the beam line . . . 18
3.2 DCA between V^0 daughters . . . 20
3.3 Cosine of pointing angle . . . 22
3.4 V^0 invariant mass . . . 24
3.5 V^0 decay length . . . 26
3.6 Armenteros-Podolanski plot . . . 28
3.7 Cosine of theta angle in CMS . . . 29

4 Summary **31**

A Code merging the trees **32**

A.1 Libraries . . . 32
A.2 Class header file . . . 33
A.3 Class source file . . . 34
A.4 Main function . . . 36
A.5 Building script . . . 38

B TTree as a ROOT data structure **39**

C Pseudorapidity **41**

List of Figures **42**

Chapter 1

Introduction

Research phenomenology

In elementary particle physics, we engage with objects and phenomena that, despite forming the fundamental building blocks of our reality, often appear beyond our comprehension. Despite the intricacies of natural laws, humanity has progressively mastered their manipulation through tools developed over the years, fostering technological advancements based on new concepts. This is driven by our relentless pursuit to uncover the deeper mysteries of nature. High energy physics is at the forefront of this endeavor, exploring matter at the smallest scales. It pioneers new discoveries in the realm of particle physics, revealing the essential constituents of everything, even though most of these particles are exotic and not part of our everyday experience. The mere knowledge of their existence expands our understanding and fuels the passion of those dedicated to advancing this field. Using sophisticated tools and methodologies, we are able to gather crucial information to decipher the subatomic world. This, in turn, facilitates the refinement of existing theories, exploration of novel phenomena, and further technological innovation.

1.1 STAR experiment

Solenoidal Tracker at RHIC, also known as STAR, is one of two currently active experiments at the Relativistic Heavy Ion Collider, which is known as RHIC. The experiment is led in Brookhaven National Laboratory (BNL). . RHIC is the main object at the BNL and the most sizeable particle accelerator. There is only two heavy-ion colliders where aforementioned is the only one ever built which can collide beams previously spin-polarized protons, and RHIC is one of them. It means that all protons in a given beam are ”spinning” in the same direction, which will make it possible to learn the reason why quarks constitute only 30% of this particle spin [1]. A view of the entire complex with appropriately marked specific places for the experiment and the lines connecting the vacuum tubes through which the accelerated protons or ions move is shown in the figure 1.1

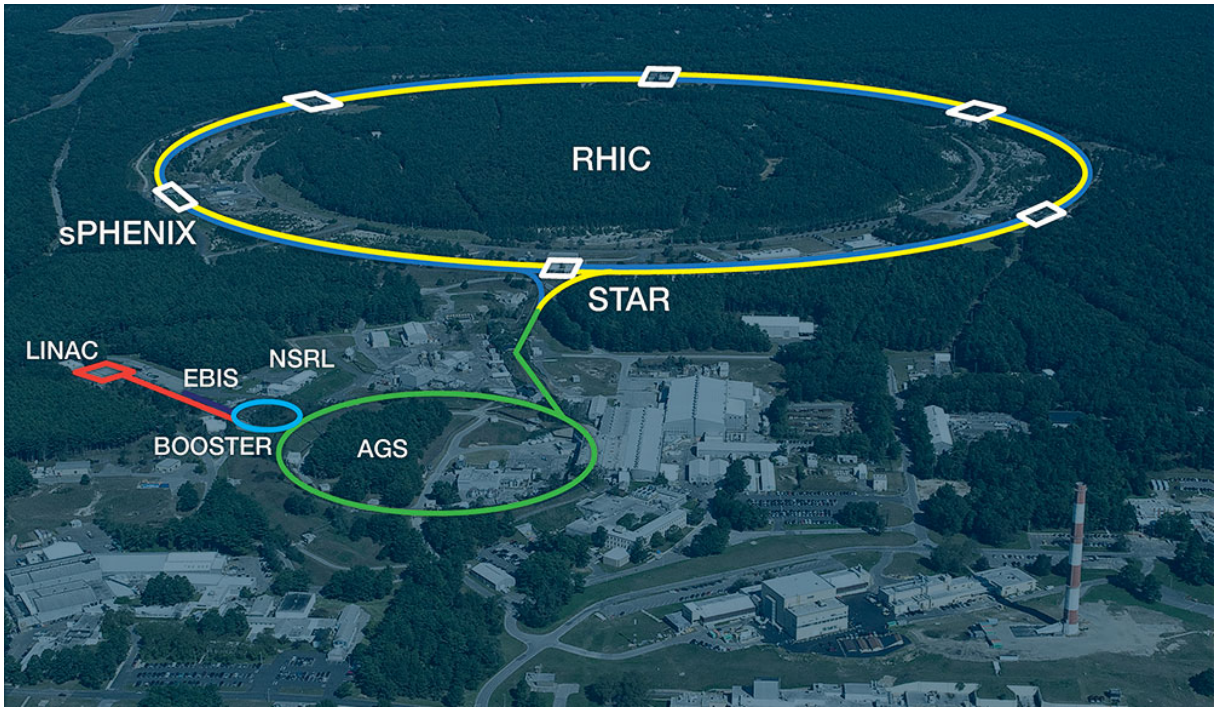

Fig. 1.1: View of the complex of RHIC [2] with the specific experiments and special objects marked.

The Ions can achieve energies up to 100 GeV [3] per the nucleon but in case of protons, energies are up to 250 GeV. However, since 2010, higher energies in the case of heavy ion collisions have been achieved by another research center located in Geneva, CERN. Therefore, it has the potential to discover phenomena involving previously unavailable energies and this is not the end, ultimately they are to be even larger.

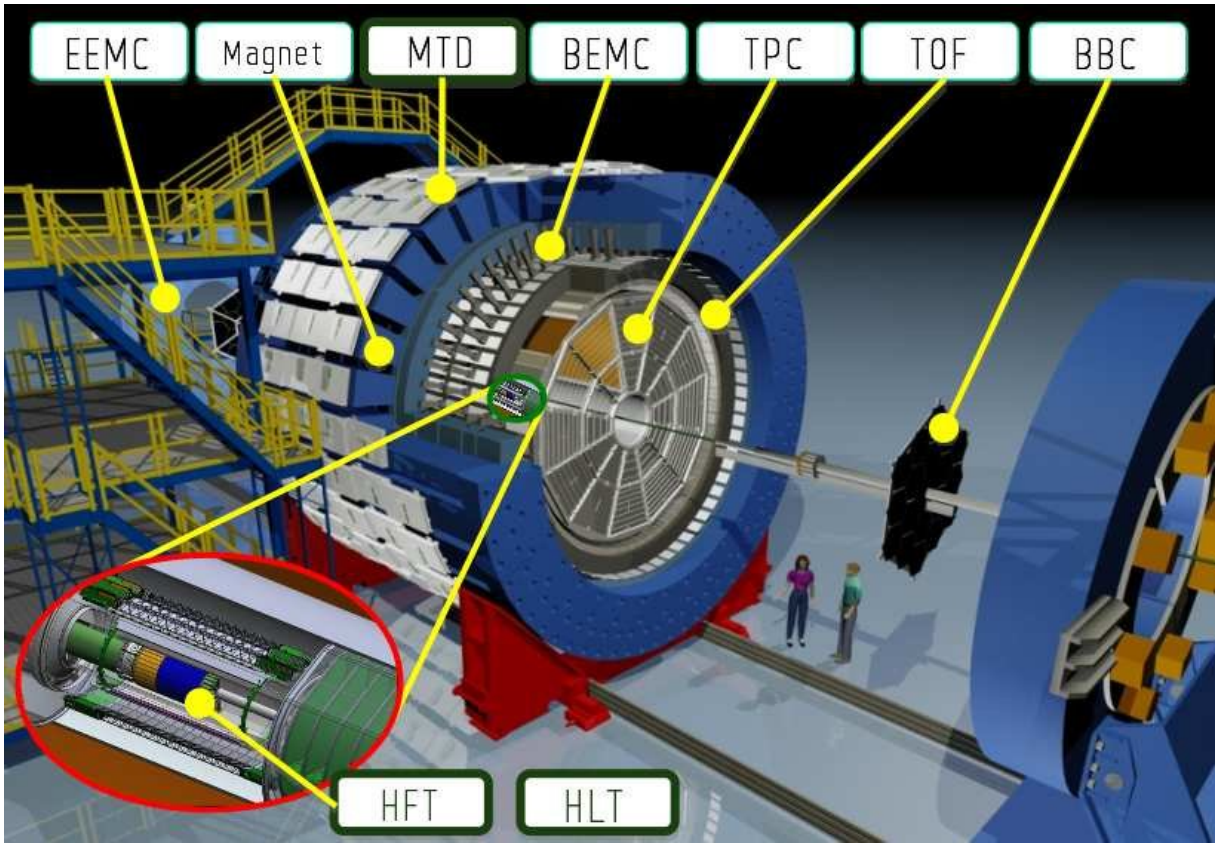

Fig. 1.2: Illustration showing the main components of the STAR detector [4].

The main elements of the STAR experiment are listed below [5], their location is shown in the illustration 1.2.

- ★ Time Projection Chamber (TPC) is the heart of the detector, which allows to reconstruct a trajectory of the particles in the gas volume, analyze energy losses of charged particles by measuring the amount of energy they lose (dE/dx) as they pass through a detector,
- ★ Time-Of-Flight Detector (TOF) allows particles to be identified based on their time of flight. It also allows to select interesting events in the trigger system and to eliminate those TPC tracks that are not synchronized in time with the main event,
- ★ The magnet is solenoid, it ensures a uniform magnetic field essential for the curvature of charged particles. This curvature is used to determine the momentum of the particles,
- ★ Muon Telescope Detector (MTD) provides muon identification,
- ★ Barrel Electromagnetic Calorimeter (BEMC) in the STAR experiment is used to measure the energy of electrons and photons in high-energy collisions. It is a barrel calorimeter, which means that it surrounds the collision region in a cylindrical arrangement, covering a large fixed angle around the collision point,

- ★ The Endcap Electromagnetic Calorimeter (EEMC) complements the operation of the BEMC. It is a detector designed to measure the energy of particles emerging at small angles to the direction of the beam. This allows the analysis of particles that are not fully recorded by BEMC due to their direction of flight,
- ★ Beam-Beam Counters (BBC) are positioned at each end of the detector. They are used to trigger the entire detection system and to determine the vertex position of the collisions. This allows for the identification of the location within the detector where particle collisions have occurred. The BBC is essential for examining the centrality of heavy ions collisions,
- ★ Heavy Flavor Tracker (HFT) is advanced detector allowing for precision tracking measurements of heavy quarks at low momentum,
- ★ High-Level Trigger (HLT) is a system processes data in real-time to select the most interesting events for recording and detailed analysis,
- ★ Roman Pot (RP) detectors are used for precise measurement of protons that are scattered at very small angles as a result of collisions and reconstruct their momenta with high efficiency and precision.

All of the above technologies provide us a complete picture of a physical event. The physics program in the STAR experiment at RHIC focuses on three main research areas:

1. First of all, the experiment focused on the study of high-density quantum chromodynamics (QCD) and the search for evidence of the existence of quark-gluon plasma through heavy ion collisions (QGP), which was achieved shortly thereafter, it is described in section 1.2.
2. The analysis of the interactions of photons which are the result of intense electromagnetic fields of colliding ions and strong interactions, using ultra-peripheral collisions that are described in section 1.3.
3. Making measurements of the spin structure of the proton in order to determine the preferred orientation of gluon spins and their contribution to the total spin of which is the focus of the analysis of inelastic proton-proton processes which are described in section 1.4. Part of this research deals with interactions described by Pomeron exchanges[1] which are of particular interest because they provide insights into the gluon structure of hadrons.

[1]Pomeron exchange is the exchange of nuclear (strong) forces where the Pomeron itself represents the absorption part of the nuclear potential and is often treated as a colorless particle with vacuum quantum numbers. Pomeron can be thought of as a cluster of gluons, and this exchange results in a type of interaction where the colliding particles survive the process, and additional particles are produced [6].

1.2 Quark-gluon plasma

The BNL is famous in science world not only by numerous Nobel Prizes, but also gained prominence for providing in June 2000 experimental evidence of the existence of quark-gluon plasma (QGP). This is a state of matter in which the elementary building blocks described by a powerful mathematical apparatus that make up matter, i.e. quarks and gluons, which are usually trapped in hadrons (we live in lower energies every day), become free. This state dominated in the early stages of the universe because it is defined by high temperature and density. Interestingly, research conducted in this direction has shown that this plasma, despite being a different state of matter, behaves almost like an ideal low-viscosity fluid. Although before the observations, theorists expected a rather weakly interacting gas consisting of those components present inside.

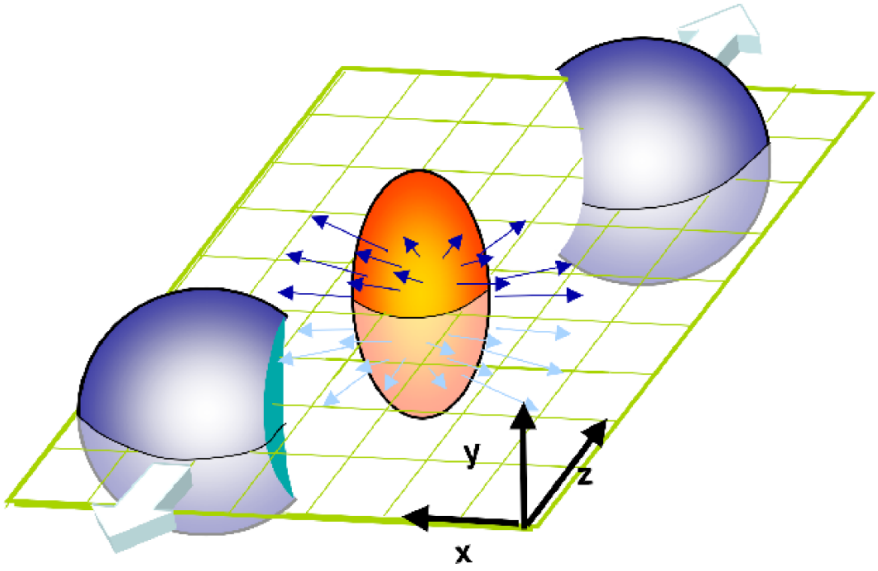

Fig. 1.3: Illustration presenting an off-center collision of heavy ions, the interaction zone, is often called a ”fireball” [7].

The image 1.3 depicts the QGP after a non-central heavy ion collision, which is where the ”fireball” or area of interaction is created. This region is a colored ellipse on it symbolizing the flow nature of this state.

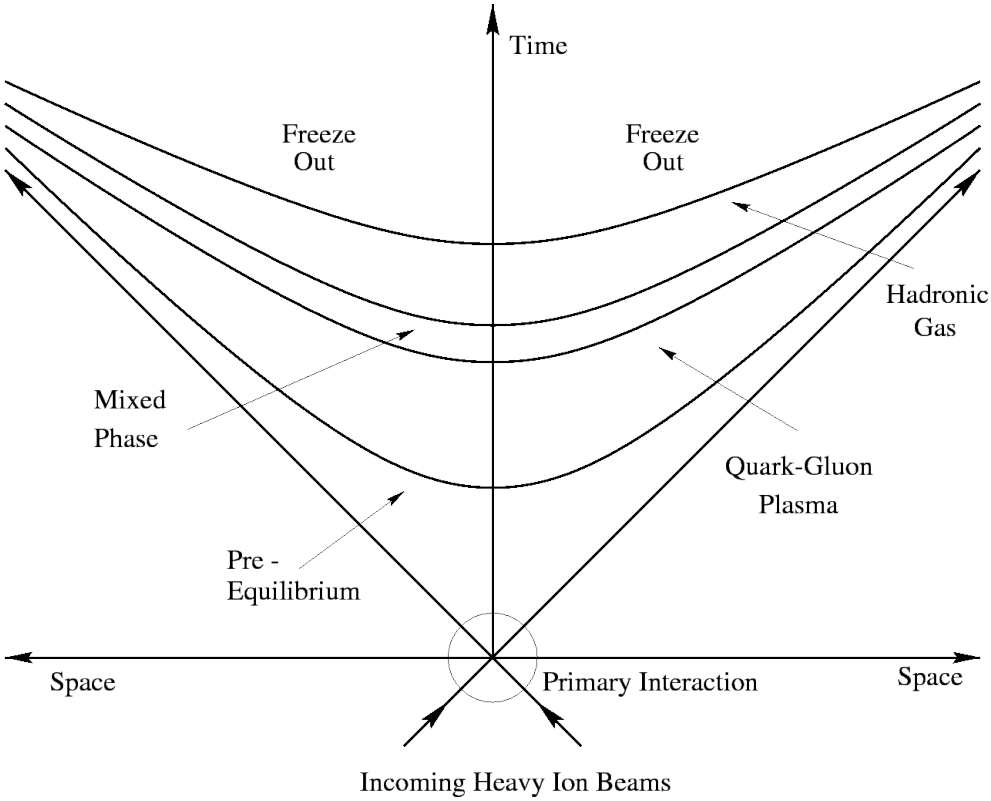

Fig. 1.4: The of evolution transforming to a QGP in a space-time diagram [8].

These which is created in the collision secondary partons interact with each other, as a result the parton density increases, leads to easier formation of $q\bar{q}$ pairs (Debye screening process[2]). After equilibrium of matter it forms to quark-gluon plasma [9]. After a short time state expands and cools, it hadronizates again, i.e. the quarks are again held together by the color force, because they cannot exist in a free state for any length of time.

The energy required for this purpose must be greater than the rest mass of the predicted particle, because in addition to its creation, it is required to overcome potential barriers and transform to kinetic energy. Since all quarks are fermions, this means that they obey the Pauli rule[3]. As light quarks are produced, there are enough of them to fill all available low energy levels. Therefore, processes of producing heavier quarks such as the strange quark $s(\bar{s})$ and, consequently, hadrons: the K_S^0 meson and $\Lambda(\bar{\Lambda})$ baryons become possible.

[2]Phenomena of attenuation of the electric field inside the plasma due to the presence of free charges (electrons and ions). In a high-temperature plasma, such as a quark-gluon plasma, free charges shielding and reducing the effective range interactions between particles.

[3]No two identical fermions, can occupy the same quantum state simultaneously within the same quantum system [10].

1.3 Ultra-peripheral collisions

Ultra-peripheral collisions (UPC) are a unique type of collision studied, among others, at BNL. During these collisions, ultra-relativistic heavy ions (e.g. gold or silver) do not collide directly with each other, but fly at relatively close distances. Consequently, for such large collision parameters ($b > R_1 + R_2$), there is no strong nuclear interaction between them, only electromagnetic interaction, so they effectively behave like a beam of photons. Thus, ions interact through photon-ion and photon-photon collisions with very high energies. Possible processes are summarized in the figure 1.10 which presents Feynman diagrams for this event.

The high energy of the process also leads to Lorentz contraction, this effect is illustrated in the figure 1.11. In the rest frame of the atomic nucleus, the flux of equivalent photons described by the equation 1.1 is stronger in the case of heavy ion collisions (e.g. Au-Au collisions) compared to proton-proton collisions. It results from the proportionality of the flux to the square of the atomic number Z. For example, in the case of gold, the amplification factor is $79^2 = 6241$.

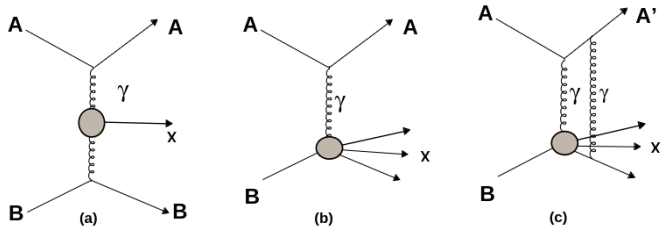

Fig. 1.10: An illustration of (a) an electromagnetic process where photons radiated by ions engage with each other, (b) a photon and nucleus interaction involving a photon from one ion interacting with a different nucleus, and (c) a reaction involving photons and nuclei that results in the disintegration of the nucleus due to photon transfer [12].

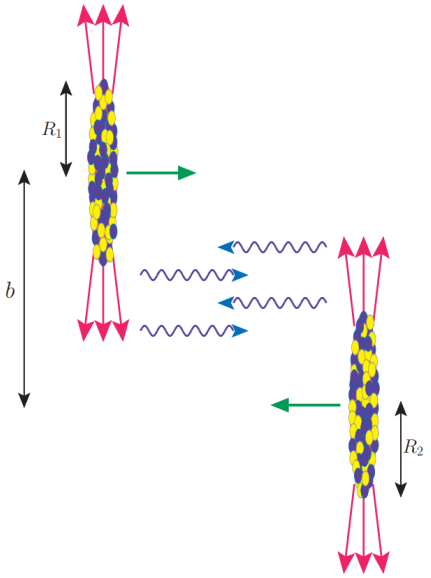

Fig. 1.11: Drawing showing the interaction of heavy ions in ultra-peripheral collisions [11]. The apparent flattening in the direction of flight in the laboratory-related system is due to Lorentz contraction.

The electromagnetic field around an ultra-relativistic nucleus can be described as a flux of almost-real photons. In this so-called equivalent photon approximation, the number of photons at a distance b from nucleus per unit energy and per unit area is given by the equation 1.1 [12].

$$\frac{d^3N_\gamma}{dkd^2b} = \frac{Z^2\alpha x^2}{\pi^2 kb^2}\left[K_1^2(x) + \frac{K_0^2(x)}{\gamma^2}\right], \text{ where } \quad x = \frac{kb}{\gamma\hbar c}, \tag{1.1}$$

α is the electromagnetic fine structure constant[4], Z is the atomic number, b is the collision parameter, γ is the Lorentz factor, k is the momentum of the interacting photon and $K_0(x), K_1(x)$ are modified Bessel functions.

[4] A dimensionless physical constant approximately equal to $\frac{1}{137}$, it is a mathematical parameter describing the strength of the electromagnetic interaction, therefore it is of key importance in quantum electrodynamics [13].

1.4 Inelastic proton-proton scattering

Proton structure

Proton collisions are interesting because their structure is not a collection of nucleons, because they themselves constitute this single component, which was considered elementary until 1964, when physicists Murray Gell-Mann and George Zweig independently postulated the existence of quarks. Quarks are fundamental particles that form other particles, while gluons are carriers of the strong force that bind quarks together. However, at that time they were still considered mathematical concepts, not real particles.

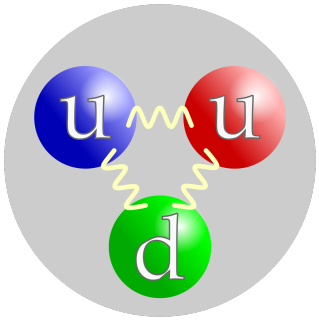

Fig. 1.12: Structure of proton.

Parton model

The parton model, is a concept in particle physics that assumes that hadrons are composed of smaller, point-like particles called partons. Partons are the general name for quarks and gluons. In experiments such as deep inelastic scattering, partons are considered to be the active components of hadrons with which other particles interact. Scattering experiments conducted in the early 1970s confirmed the existence of partons. The results of these experiments demonstrated the existence of point particles in hadrons, their charges were measured, and found that about half of the proton's momentum is carried by gluons, which are electrically neutral. These results were consistent with the predictions of the quark model developed by Murray Gell-Mann and others. Feynman published the parton model in 1969. It is worth noting that he was already familiar with the concept of quarks and gluons by Gell-Mann and Zweig, especially since he published a paper on them together with them in 1964. In this work on partons, Feynman also divided collision experiments into exclusive and inclusive. Exclusive processes are those in which the energies and momenta of all particles produced in a reaction are measured. This allows for a through analysis of the collision process. In contrast, inclusive experiments focus on measuring only some of the reaction products, which provides information about the general properties of the process but not about the details of the collision.

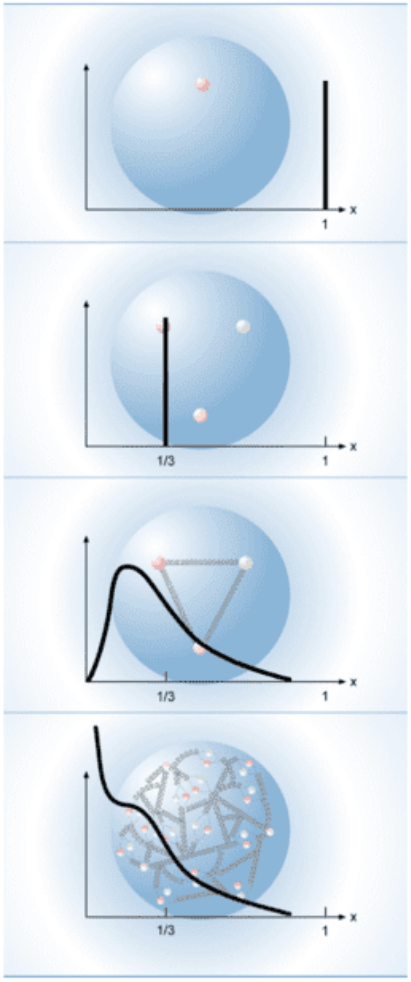

Fig. 1.13: PDF[5], particle has no internal structure (1), consists of three quarks (2), there is gluon interaction (3), including QCD effects (4) in the momentum fraction argument [14].

[5]Parton Distribution Function is a probability that a parton carries a fraction of particle momentum.

Particle scattering

The total cross section of proton-proton collisions is splited into non-diffractive and diffractive interactions, where the latter are divided into elastic and inelastic processes. We define diffraction interactions as particle interactions that exchange only vacuum quantum numbers. The non-diffractive [15] process proceeds according to the scheme $a + b \to X$. While, diffractive processes are divided into:

- ★ Elastic Scattering: $a + b \to a' + b'$;
- ★ Single Diffraction (SD): $a + b \to a' + X$, where particle b dissociates into state X;
- ★ Double Diffraction (DD): $a + b \to X + Y$, where both particles a i b dissociate into states X and Y;
- ★ Central Diffraction (CD): $a+b \to a'+X+b'$, where additional state X is produced.

The $X(Y)$ represents either a dissociated proton or a centrally-produced hadronic system. These inelastic pp processes can be illustrated using the Pomeron concept, which describes the interactions as a series of fusion processes. These include proton-proton (ND), Pomeron-proton (SD and DD), and Pomeron-Pomeron (CD) interactions. This conceptual framework helps in understanding the complex dynamics of inelastic scattering. The described processes are illustrated in figure 1.14.

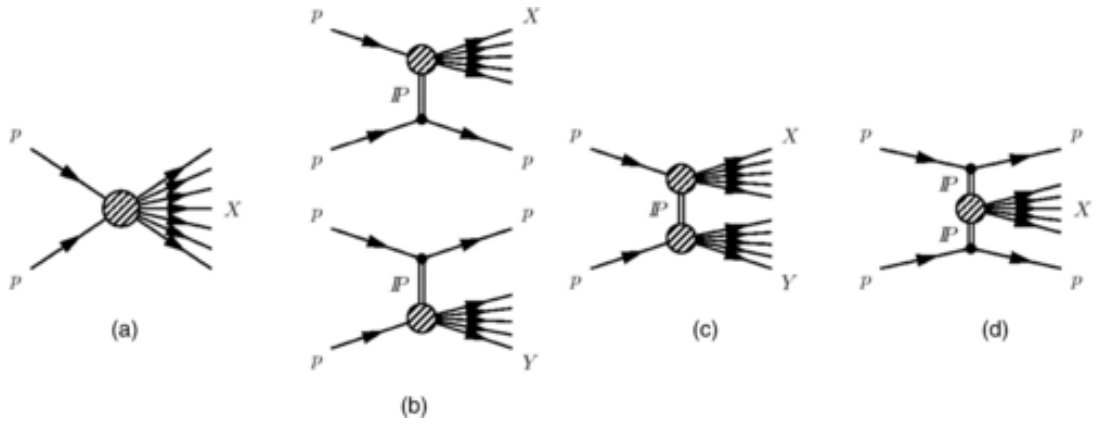

Fig. 1.14: Schematic representation of processes in proton-proton collisions: (a) nondiffractive process $pp \to X$, and diffractive processes: (b) single dissociation $pp \to Xp$ or $pp \to pY$, (c) double dissociation $pp \to XY$, (d) central diffraction scattering $pp \to pXp$, where X and Y represents the dissociated proton or centrally produced hadronic system [16].

1.5 Production and decay of strange particles

The strangeness

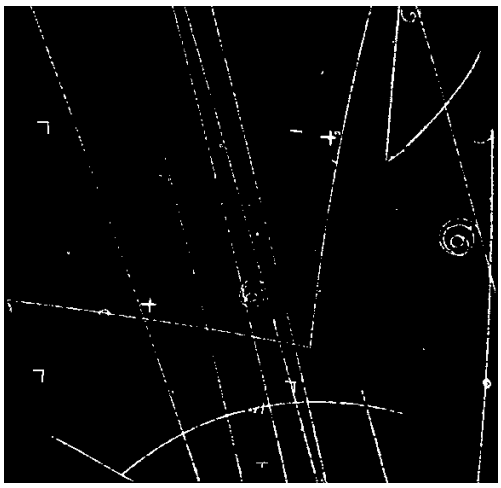

Fig. 1.15: Photo of the decay of strange particles into charged particles in a liquid hydrogen bubble chamber [17].

The characteristic lifetime of strange particles is of the order of $10^{-8} - 10^{-9}$s. The first particles discovered, even before the quark structure was postulated, were kaons, their lifetime was longer than expected, hence the name "strange". Kaons are interesting because they can only split into pions. The rest decays either directly onto protons and pions, or it happens sequentially. Strange particles, similarly to the phenomenon of pair production (particle-antiparticle), are always created in a pair, but due to the preservation of strangeness instead of charge. Strangeness is not just a term, but a strictly defined quantum number, which was introduced by M.Gell-Mann and Nishijima in 1953. Particles are called strange if they contain at least one strange quark (antiquark). These are, for example: K_S^0, Λ, $\overline{\Lambda}$, Ξ^-, $\overline{\Xi}^+$, Ω^-, $\overline{\Omega}^+$. The strangeness of a particle is the sum of the strangeness of its components (quarks). The strangeness is not always preserved in the weak interaction, it can change by -1 or 1. However, it is preserved during the production of these particles under the influence of the strong nuclear force. An example process that does not preserve this number is decay of charged kaons:

$$K^- \rightarrow \mu^- + \bar{\nu_\mu}. \tag{1.2}$$

The strangeness of the relevant particles is $S_K = -1$, $S_\mu = 0$, $S_{\bar{\nu}} = 0$, as a result of the decay, the strangeness has changed by 1. The strangeness of the muon and muon antineutrino is 0 because they are not built by a strange quark.

The neutral strange particles decaying into two charged particles are often called V^0-particles due to the characteristic shape of this letter created during the weak decays they undergo, which can be observed in the cloud chamber as it is shown in figure 1.15. In the case of a neutral meson K_S^0, decay to $\pi^+\pi^-$ accounts for 69.2%, while decay: $K_S^0 \rightarrow 2\pi^0$ accounts for 30,7%. The particle structure K_S^0 is ds. The quark structure of Λ is uds, and it decays into $p\pi^-$, while the $\bar{\Lambda}$ quark structure is $\bar{u}\bar{d}\bar{s}$, and it decays into $\bar{p}\pi^+$.

V^0 decay topology

These particles can exist in different states due to the different combination possibilities of quarks. They are unstable and undergo weak decay. It means that they can travel some distance from the point of their creation in a collision before they decay , but they will not reach the TPC detector, which is used to measure decay products. Although uncharged strange particles are not directly detected,

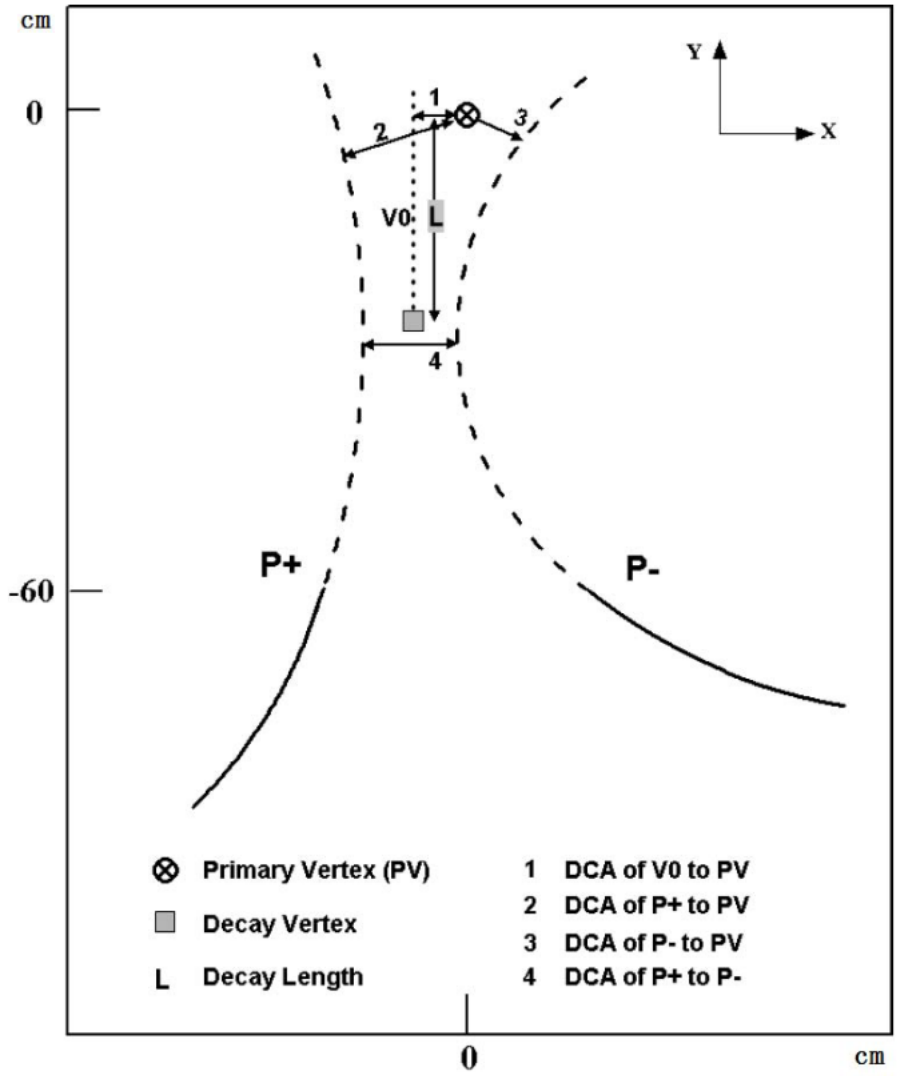

Fig. 1.16: The strange V^0-particle decay topology. P+(-) refers to a positively(negatively) charged daughters. The solid line is detectable by the TPC anode sector 60 cm away from the beam line. The dashed line is extrapolated from the solid line or reconstructed V^0-candidate. The topological cuts are shown in the plot which is from Hai Jiang's thesis [18].

they can be reconstructed based on their charged decay products such as pions, kaons, and protons interacting with detector material. These products can be identified through the measurement of energy loss dE/dx in the TPC detector. General scheme is presented on the figure 1.16. In addition to outlining the decay topology, the drawing also defines the appropriate quantities marked with numbers from 1 to 4.

These quantities are DCA (Distance of Closest Approach), which is a measure of the shortest distance between two objects in space. It is important in particle identification because it allows us to assess whether the particle was produced in a primary collision or was created by the decay of another particle. Only DCA between daughters is independent from the Primary Vertex (PV).

1.6 Concep of the Armenteros-Podolanski plot

In 1954, R. Armenteros and J. Podolanski [19] proposed a method for analyzing the dynamics of V^0- particles decay. This method allowed to distinguish the decays of $K^0_S \to \pi^+\pi^-$ from the decays of $\Lambda^0 \to p\pi^-$ without assuming the masses of the final decay products but only with the transverse momentum (with respect to direction of V^0-particle) of the positivelly charged daughter as a function of parameter α , which is defined by the equation 1.3.

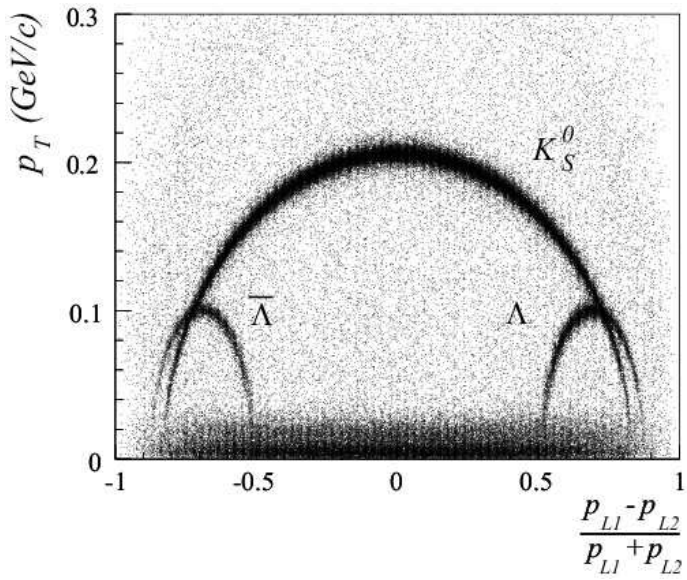

Fig. 1.17: The Armenteros-Podolanski Plot showing the dependence of the transverse momentum on the α parameter with clear traces corresponding to the meson K^0_S and baryons: Λ and $\bar{\Lambda}$ [20].

$$\alpha = \frac{p_{L1} - p_{L2}}{p_{L1} + p_{L2}}, \qquad (1.3)$$

This parameter is defined as the ratio of the difference between the longitudinal momentum (with respect to direction of V^0-particle) of daughters to their sum. The points are arranged in the shapes of ellipse arcs. Each arc correspond to different V^0-particle. The obtained distribution can be explained by the fact that the decay products of $K^0_S \to \pi^+\pi^-$ possess identical masses, leading to an average symmetric distribution of their momenta. The proton (antiproton) tends to carry a larger portion of the momentum on average, leading to an asymmetric distribution. The important thing in understanding this plot, presented as a two-dimensional histogram 1.17, is to note that the momenta are analized in laboratory frame, with the z-axis pointing along the momentum of the V^0-particle. The two particles have the same transverse momentum due to momentum balance. Using ultra-relativistic approximation ($\beta \to 1$), we obtain the functional form of decay described by the ellipse equation [21] where all parameters depend on the masses of the particles involved in the decay:

$$\frac{(\alpha-\alpha_0)^2}{r_\alpha^2} + \frac{p_T^2}{p^{*2}} = 1, \text{where} \quad \alpha_0 = \frac{m_1^2 - m_2^2}{M^2}, \quad r_\alpha = \frac{2p^*}{M}, \qquad (1.4)$$

$$p^{*2} = \frac{1}{4M^2}\left(M^4 + m_1^4 + m_2^4 - 2M^2(m_1^2+m_2^2) - 2m_1^2m_2^2\right). \qquad (1.5)$$

Chapter 2

Description of the code

In the STAR experiment the raw detector data are reconstructed and physical information (like TPC track momenta) are stored in the data storage tag (DST) in the form of the so called *microDST* ROOT tree format. *microDST* contains full physics and limited detector level information and usually are not directly used in the analysis due to the limited storage space and time needed for processing. Therefore, compact data structures (the so called *picoDST*) are created with limited physics information and used later in the analysis. For some specific purposes even smaller data structure are created (*femtoDST*). One example of the *femtoDST* is *upcDST* structure used mainly for analysis of ultra-peripheral collision data but also proton-proton collision data used for diffractive and exclusive analysis. *upcDST* contains information from RP subsystem (not included in *picoDST*) and only small part of the TPC tracks which are associated with primary vertexes. Primary vertexes are defined as vertexes located close (less than 3 cm) to the beam-line. This condition is a serious limitation for analysis of short-lived strange particles especially Λ and $\bar{\Lambda}$ baryon. Reproduction of *upcDST* samples directly from archived *microDST* is very time consuming process. Therefore much faster possibility was investigated to use *picoDST* samples to extract missing information and merge them event by event with existing *upcDST*. Two scenarios were considered. In both scenarios the algorithm to reconstruct V^0-particles is run with *picoDST* as input and then special temporary ROOT tree is created with V^0-particle candidates and their daughters or alternatively with only daughters from which original V^0-particle can be reconstructed later in the physics analysis. As a final step the temporary ROOT tree will be merged event by event with existing *upcDST* to form extended *upcDST* structure. In this diploma work the first scenario is described. The algorithm for reconstruction of V^0-particle was already coded in the software used at STAR experiment. The author task was to write a program for merging temporary ROOT tree with exiting *upcDST*.

The first step to using the program is to compile it with the prepared script. The script compiles the source code using the g++ compiler with the appropriate libraries and flags, including those from the ROOT environment and these defined by STAR software environment. The compilation script is included in the appendix A.5.

When executing the program, provide the following arguments in the command line:

1. The name of the file containing original *upcDST* tree,
2. The name of the file containing temporary ROOT tree with V^0-candidate information (in this case K^0_S and Λ candidates),

where the files that these arguments are must have a $.root$ extention. For proper merging, the initial code fragment checks whether the files and trees loaded from them have been loaded correctly, this is standard error handling. Then the line: $UPCTree->SetBranchAddress("mUPCEvent", \&upcEvt);$ links mUPCEvent branch of *upcDST* tree with a certain object describing content of a single event stored in *upcDST* tree. Since the first file is opened in update mode, one can create new branches that will store information from temporary ROOT file about K^0_S and Λ candidates. These branches store data vectors whose meaning is presented below.

- ★ **eventId:** Unique event ID;
- ★ **leadPt/subleadPt:** Transverse momentum of the leading[1]/subleading daughter;
- ★ **leadPhi/subleadPhi:** Azimuthal angle of the leading/subleading daughter;
- ★ **leadEta/subleadEta:** Pseudorapidity[2] of the leading/subleading daughter;
- ★ **p1Pt/p2Pt:** Transverse momentum of the first/second daughter;
- ★ **p1Phi/p2Phi:** Azimuthal angle of the first/second daughter;
- ★ **p1Eta/p2Eta:** Pseudorapidity of the first/second daughter;
- ★ **p1Ch:** Electric charge of the first daughter from the V^0-candidate;
- ★ **p1HasTOFInfo/p2HasTOFInfo:** Information about whether the first/second daughter from the V^0-candidate has time of flight (TOF) data;
- ★ **pairCharge:** Electric charge of a V^0-candidate;
- ★ **pairPhi:** Azimuthal angle of the V^0-candidate;
- ★ **pairEta:** Pseudorapidity of the V^0-candidate;
- ★ **pairPt:** Transverse momentum of the V^0-candidate;
- ★ **pairMass:** Mass of the V^0-candidate;

An entry in the *upcDST* tree corresponds to the single collision of the two bunches of protons or ions identified by a unique event identification number (*eventID*) in the given run. An entry in the tree with V^0-candidates corresponds to the single V^0-candidate. Entries in both trees are ordered with respect to the *eventID* therefore following procedure was implemented in order to add one, many or no V^0-candidates to the single entry in the *upcDST* tree. Loop is performed through all entries of the *upcDST* tree. In

[1] Leading daughter is the one with greater transverse momentum.

[2] Quantity explained in the **appendix C**.

this (external) loop, internal loop is performed through few entries of the tree with V^0 candidates. If the *eventID* stored in both entries from two different trees are equal the V^0 candidate is added to the new branch of *upcDST* tree and the next entry with V^0 candidate is read in the of the *upcDST* tree. In the first case, the internal loop is broken and the external loop is continued as long as internal *eventID* is equal or smaller compared to external *eventID*. In the second case, the internal loop continues as long as internal *eventID* is equal or larger compared to external *eventID*. The second case can happen since *upcDST* tree contains filtered events and not all *eventID*'s are present. It is worth noting that the number of V^0 candidates may be greater than one. For this reason, the preferred structure of the new branch is a variable sized vector container, which also ensures proper memory management. To optimize the code and not needlessly loop through data that has already been analised, the internal loop starts at the entry where it stopped in the previous step, although the benefits of this approach only become apparent when dealing with large amount of entries. This solution is possible due to the fact that the entries in both trees are ordered, so there is no possibility of any candidate being missed. In case of unordered trees it is possible to change the code by extending the algorithm with an approach involving the use of a data structure such as $std :: map$. In order to use it, the *eventID* could be treated as the appropriate key. This solution required however additional full loop over tree with V^0-candidates to initialised the map. To overcame this ROOT also enables declarations of friendship between trees. This allows two or more data trees to be related, allowing information to be shared between them without physically connecting the data. This is an interesting way to bypass the merging process. A preview of the tree structure of the relevant files is presented in figure 2.1.

(a) File containing a *upcDST* tree, its structure is more nested.

(b) A file containing a tree with K^0_S meson and Λ barion.

Fig. 2.1: Tree structure of merged ROOT files.

Chapter 3

Analysis

The following describes the analysis of the simulated data. The samples contain generated K^0_S, Λ and $\bar{\Lambda}$ particles. V^0- particles were decayed and passed through STAR detector using GEANT3 package to simulate detector response. Simulated detector responce was embedded with zero-bias collider data to account for in-time and off-time background and reconstructed by STAR software in the same way as the real data are reconstructed. The analysis was performed to optimize selection of V^0-cadidates from the daughter particles measured in TPC.

The distributions of topological quantities: DCA to the beam line, DCA between daughters, hypothetical decay length and cosine of pointing angle as well as invariant mass of K^0_S, Λ and $\bar{\Lambda}$ candidates are analized. The general picture of the decay topology is presented in Figure 1.16. Below is a description of these variables:

- **DCA between V^0 daughters** is the parameter of the decay of particle into the "daughters". It is the distance of closest approach between the trajectories of these daughter particles. The middle point of this distance defines also the V^0 Decay Vertex.
- **DCA of V^0 to the beam line** is the quantity that, refers to the shortest distance between the trajectory of the V^0 candidate and the accelerator beam line. The trajectory of the V^0 is a straight line parallel to the V^0 momentum (sum of the daughters momenta) passing through the Decay Vertex. The point on the beam line closest to the trajectory of V^0 candidate defines also V^0 Production Vertex.
- V^0 **decay length** is the distance between Production Vertex and Decay Vertex.
- **Cosine of pointing angle** is the cosine of the angle between the vector connecting the Production Vertex with the Decay Vertex, and the momentum vector of the V^0 candidate. In practice, this angle helps to determine whether the daughter particles actually come from the decay of a parent particle that was created in the Production Vertex.
- V^0 **invariant mass** is a measure of mass in relativistic physics that remains constant regardless of the frame of reference. It can be used to identify whether a predicted or yet unknown particle was produced in a collision; The invariant mass is calculated from momenta of the daughters assuming particular invariant masses of the daughters characteristic for a given V^0 candidate.

The selection criteria are always a crucial point in an analysis chain. For this purpose, an object of the StUPCV0 class is created, describing the V^0 candidate, from the pair of TPC tracks corresponding to the daughters. In order to provide a fast event selection and rejection of nearly empty events the algorithm starts with the following criteria:

The analysis only uses tracks that meet the following conditions:

- ★ number of hits in TPC which form the track is grater than 15;
- ★ $p_t > 0.15$ GeV/c;
- ★ $|\eta| < 1.1$;
- ★ Tracks must be of opposite charge;
- ★ Both tracks are matched with a hit in ToF detector;
- ★ hypothesis that the track is associated with a specific particle with a probability equivalent to 2.5 σ of the normal distribution;

The analysis applies cuts on the V^0-candidate formed by pair of selected tracks:

- ★ DCA beteen V^0 daughters < 1.5 cm;
- ★ DCA of the V^0 to the beam line < 1.5 cm;
- ★ Cosine of pointing angle > 0.925;
- ★ Invariant mass of K^0_S candidate 0.495 ± 0.035 GeV$/c^2$;
- ★ Invariant mass of Λ candidate 1.115 ± 0.035 GeV$/c^2$;
- ★ Invariant mass of $\bar{\Lambda}$ candidate 1.115 ± 0.035 GeV$/c^2$;

These conditions can be understood as a loose preselection used in order to speed up the selection algorithm by reducing the amount of data stored for further processing. The file containing the K^0_S candidate tree contains 126608 events, while the file containing the Λ's candidate tree contains 124047 events. Each event contains two K^0_S or one Λ and one $\bar{\Lambda}$. Both files are very similar in size, so statistical precision for K^0_S if much better compared to Λ and $\bar{\Lambda}$. In the following subsections all the selection cuts described above are applied except the one on the plotted variable. An exception will also be the decay length histogram, which will have a reverse angle cut, which is, of course, properly argued. An Armenteros-Podolanski plot was also prepared, it confirms that the data describe the resonances considered here. The set of histograms summarizing the analysis for each candidate describes the distribution of the polar angle, θ^*, of the positively charged daughters measured in the rest frame of V^0-particle.

3.1 DCA of the V^0 to the beam line

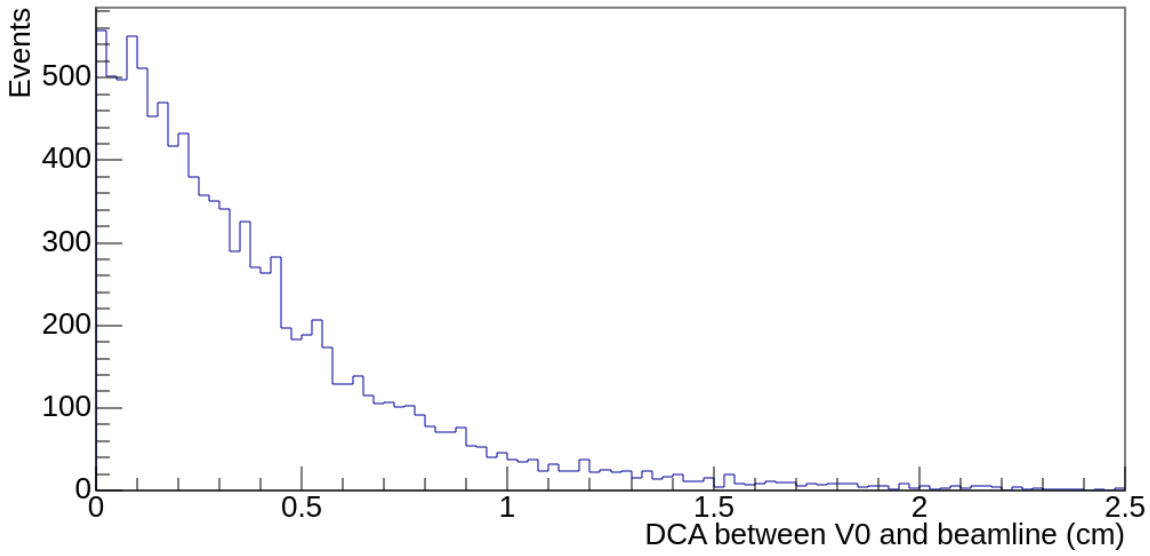

Fig. 3.1: Histogram showing Distance of Closest Approach of the V^0 to the beam line for K^0_S candidates.

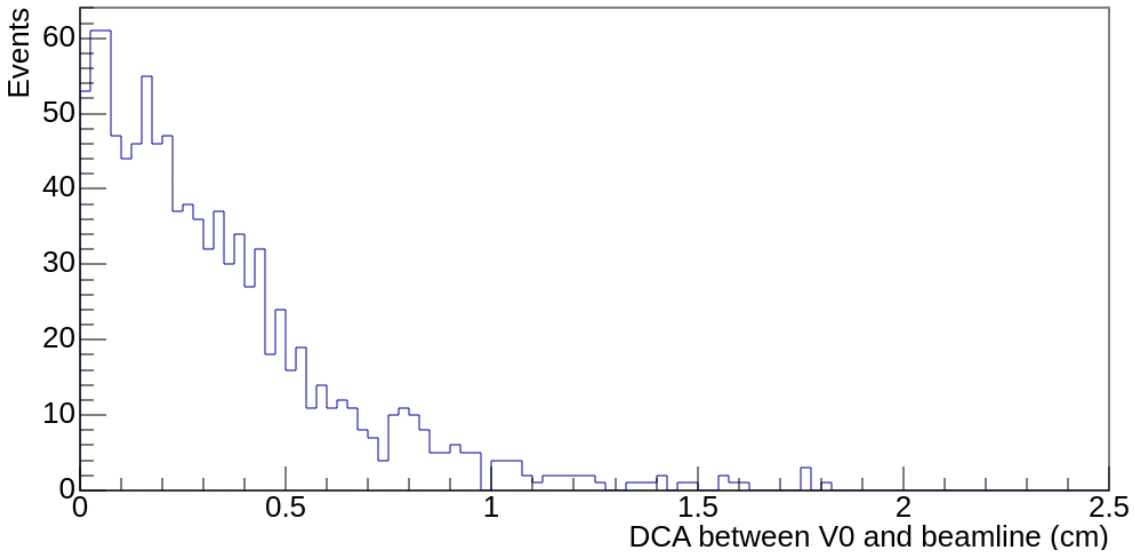

Fig. 3.2: Histogram showing Distance of Closest Approach of the V^0 to the beam line for Λ candidates.

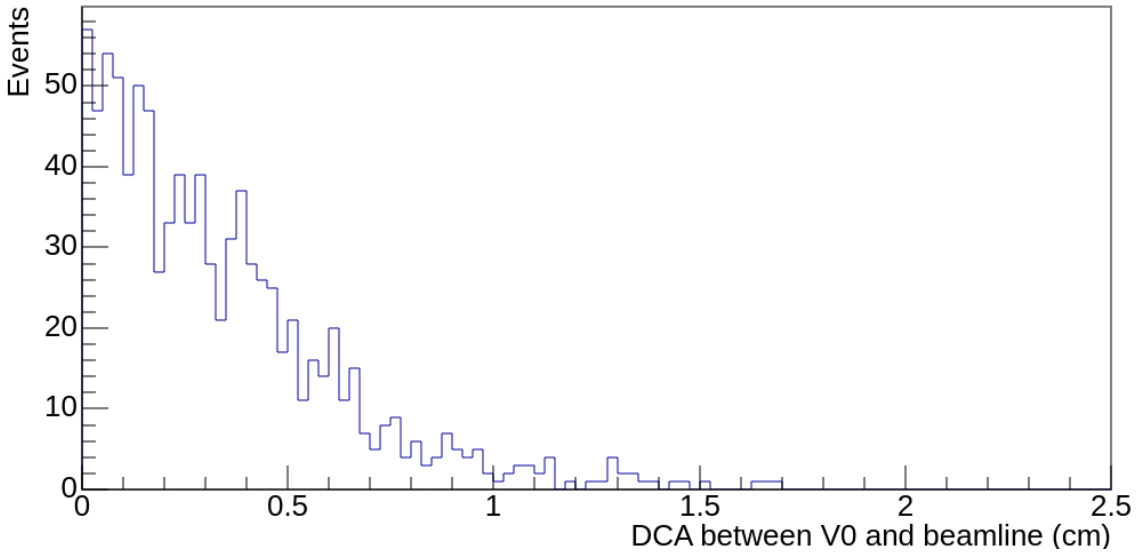

Fig. 3.3: Histogram showing Distance of Closest Approach of the V^0 to the beam line for $\bar{\Lambda}$ candidates.

On the plots showing the DCA of the V^0 to the beam line for K^0_S (figure 3.1), Λ (figure 3.2) and $\bar{\Lambda}$ (figure 3.3) particles, one can notice a decrease in the number of counts for increasing distances from the beam line. This means that most of the reconstructed V^0 candidates are produced close to the beam axis. One should also noticed that distributions are very similar for all three types of V^0 particles. The decay length decreases exponentially according to the law of spontaneous particle decay. True V^0 is produced on the beam line with pointing angle equal to 0. Small deviation form these values are allowed due to the detector resolution. But large values indicate that particle is not real V^0-particle.

3.2 DCA between V^0 daughters

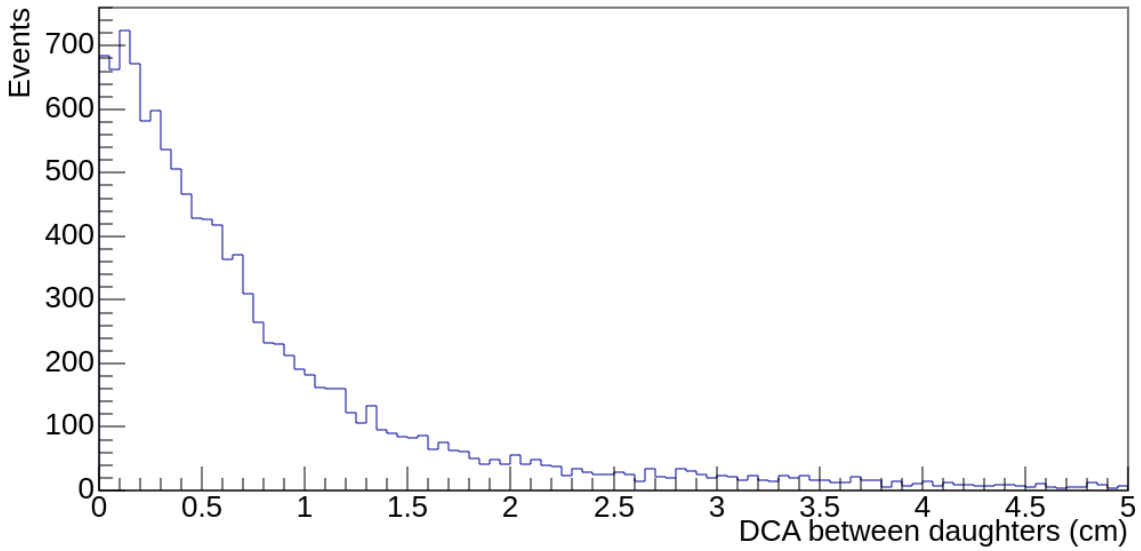

Fig. 3.4: Histogram showing Distance of Closest Approach between V^0 daughters for K^0_S candidates.

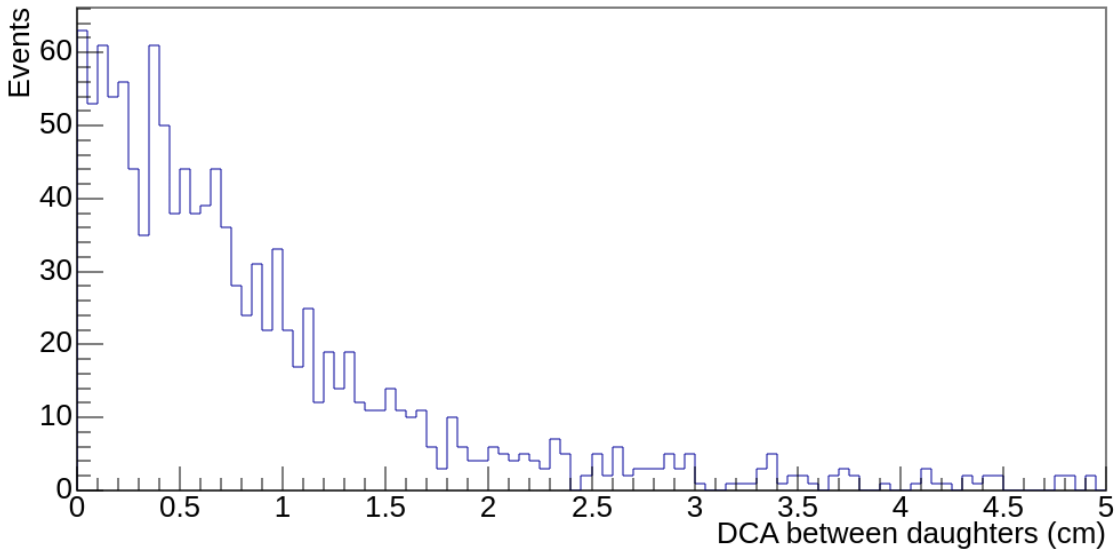

Fig. 3.5: Histogram showing Distance of Closest Approach between V^0 daughters for Λ candidates.

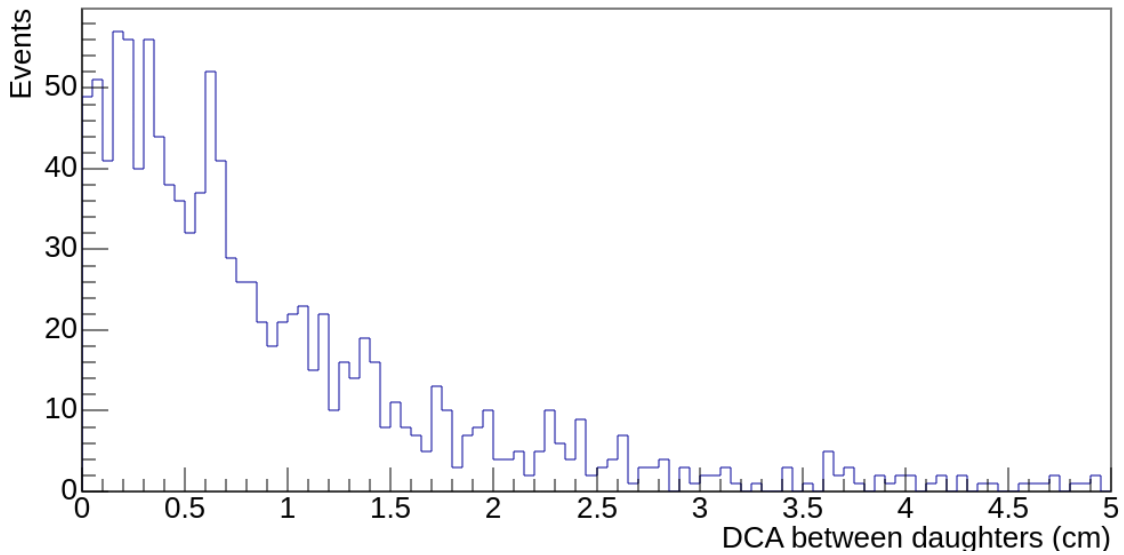

Fig. 3.6: Histogram showing Distance of Closest Approach between V^0 daughters for $\bar{\Lambda}$ candidates.

On the plots showing the DCA between V^0 daughters for K^0_S (figure 3.4), Λ (figure 3.5) and $\bar{\Lambda}$ (figure 3.6) particles, one can notice a decrease in the number of counts for increasing distances between daughters. For each of the three histograms, the number of particles decreases similarly to that for DCA of the V^0 from the beam line. This behaviour is expected since both distributions are related to the STAR detector resolution. In practice, this means that most of the daughter particles remain at close distances to each other. The daughters created in the decay of a neutral primary particle have opposite charges, therefore the magnetic field in the detector curves their tracks. For this reason, after the decay, their distance will only become greater. Therefore, point of DCA between daughters, defined as the shortest distance between daughters extended tracks, defines Production Vertex.

3.3 Cosine of pointing angle

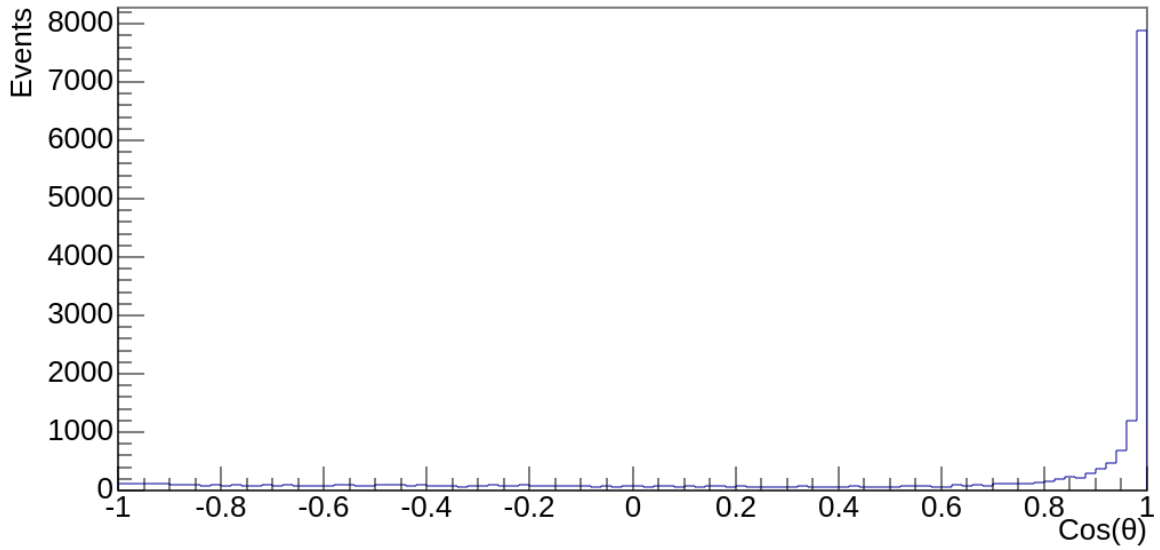

Fig. 3.7: Histogram showing the distribution of the cosine of pointing angle for K^0_S candidates.

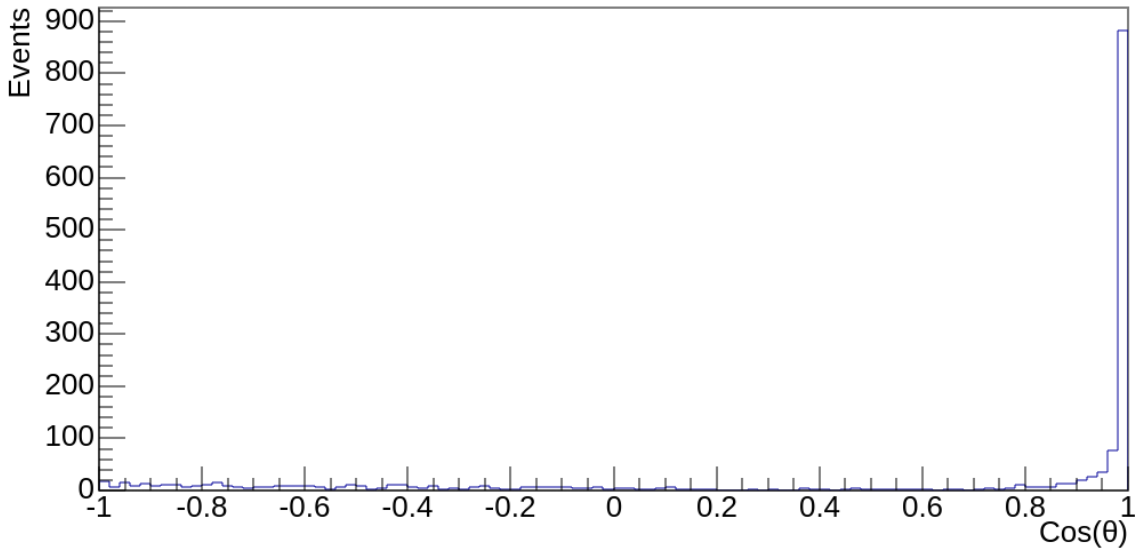

Fig. 3.8: Histogram showing the distribution of the cosine of pointing angle for Λ candidates.

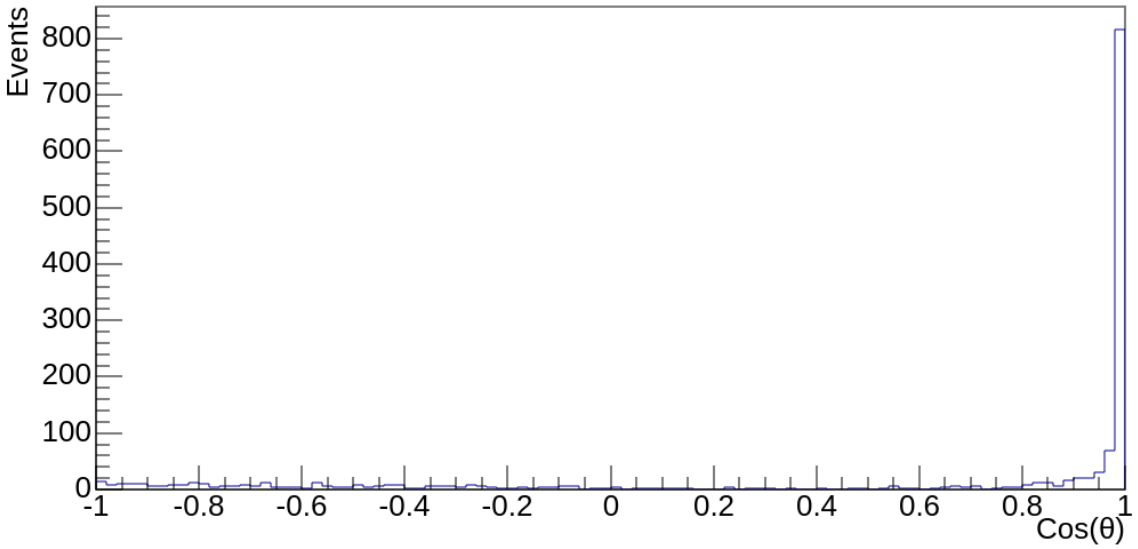

Fig. 3.9: Histogram showing the distribution of the cosine of pointing angle for $\bar{\Lambda}$ candidates.

From plots showing the cosine of pointing angle for K^0_S (figure 3.7), Λ (figure 3.8), and $\bar{\Lambda}$ (figure 3.9), it is clear that most events are close to unity for all three histograms. A high cosine of the pointing angle usually indicates that the V^0 momentum are closely aligned with the direction from the Production Vertex to the Decay Vertex. The long tail extending from -1, especially noticeable for negative cosine values, is associated with decays for which the decay length is small, it means less than 3 cm. They are the result of the detector's resolution, because the pointing angle is calculated between momentum of V^0-particle at decay vertex and the vector pointing from production vertex to the decay vertex. As the distance between these points increases, the fluctuations in the angle determination decreases.

3.4 V^0 invariant mass

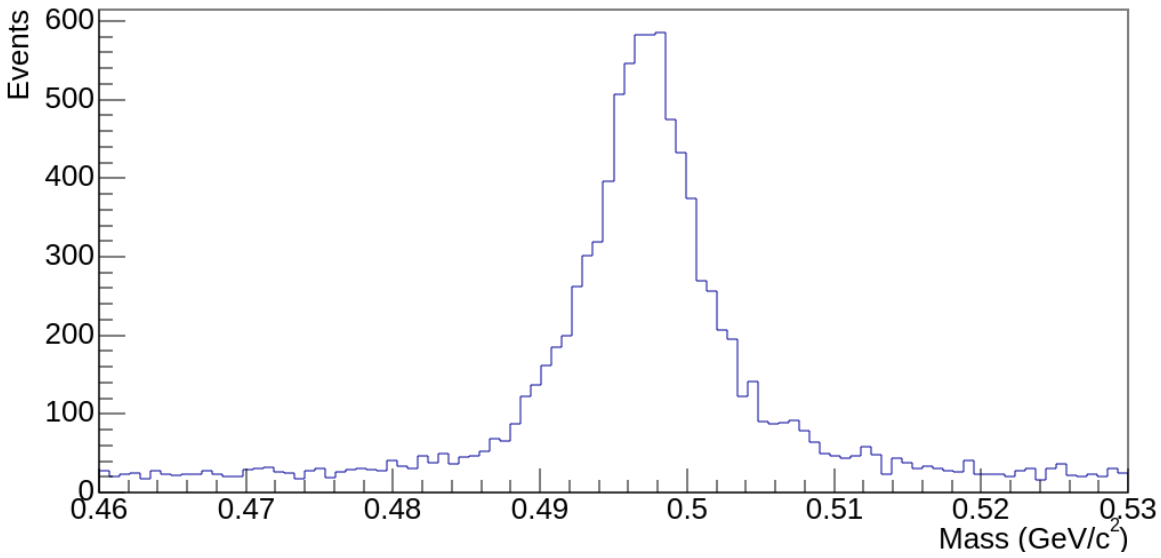

Fig. 3.10: Histogram showing the invariant mass distribution of the K^0_S candidates.

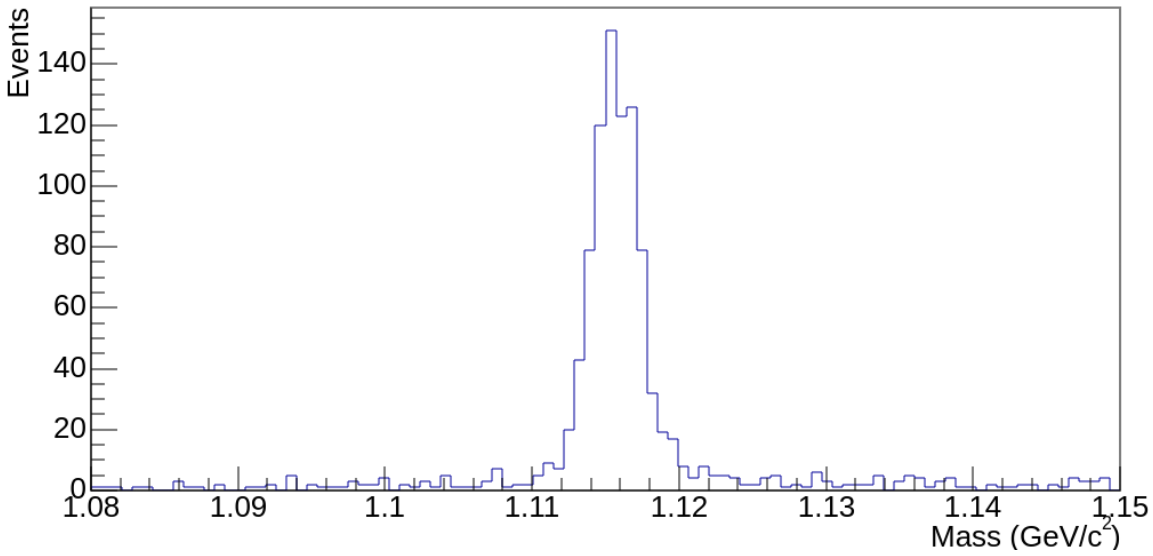

Fig. 3.11: Histogram showing the invariant mass distribution of the Λ candidates.

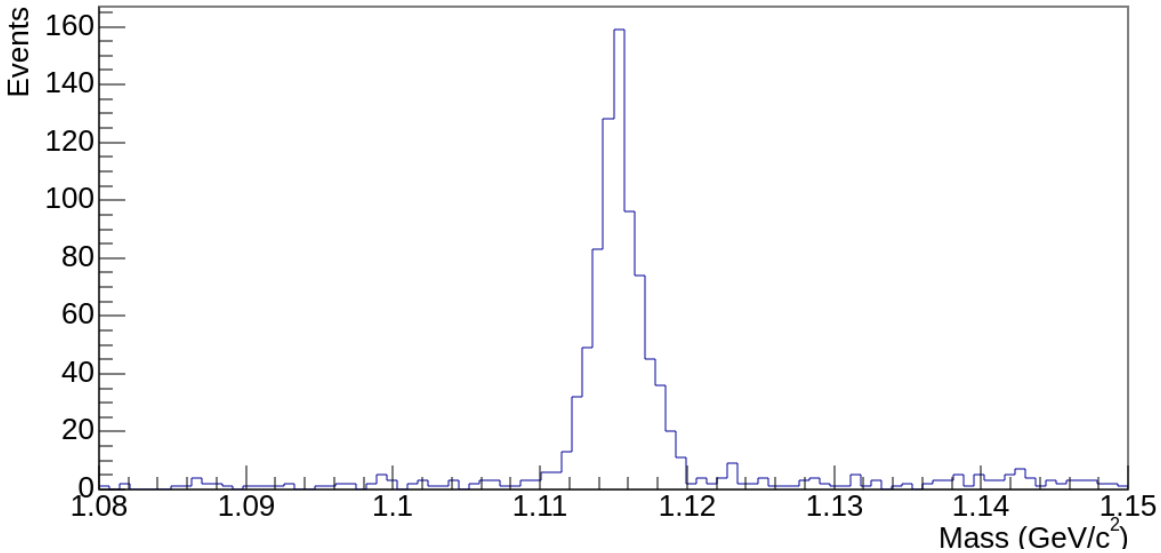

Fig. 3.12: Histogram showing the invariant mass distribution of the $\bar{\Lambda}$ candidates.

On the plots showing the invariant mass for K^0_S (figure 3.10), Λ (figure 3.11) and $\bar{\Lambda}$ (figure 3.12) peaks are observed that correspond to the rest masses of particles that are produced in collisions. The height of the histograms are related to the number of V^0 candidates close to the hypothetical rest mass of the particle. The width of the peak, in turn, is related to detector resolution. The natural widths of both the K^0_S and Λ are much smaller than those shown in the histograms. This is due to the reconstruction of the mother mass from momenta of the decay products which are measured with final detector resolution. The distribution for Λ and $\bar{\Lambda}$ are narrower than distribution for K^0_S, the reason for this is that the decay products of the K^0_S have larger momenta compared to the decay products of the Λ and therefore larger smearing is observed.

3.5 V^0 decay length

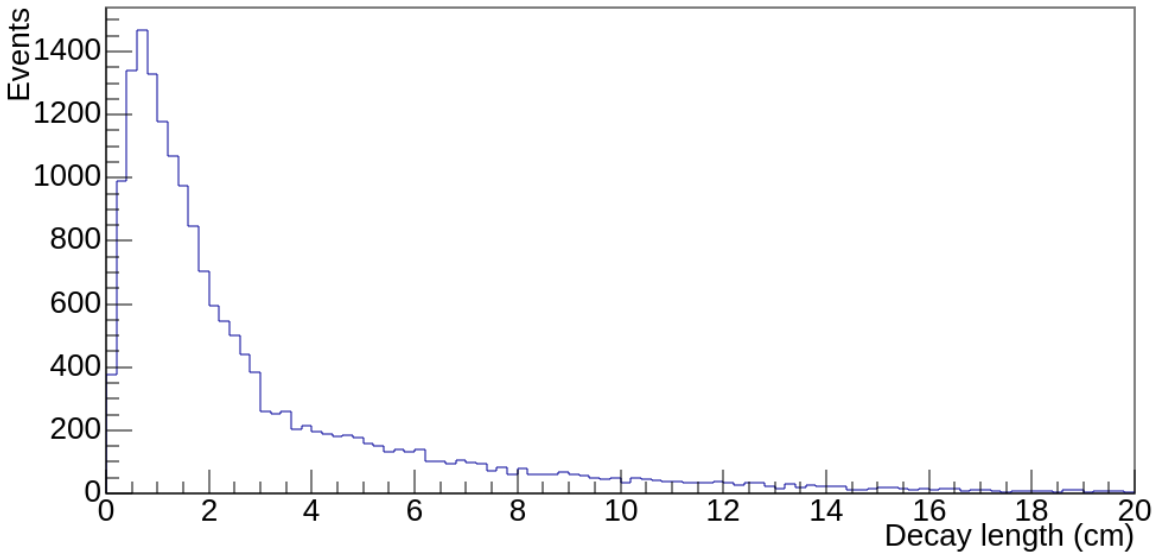

Fig. 3.13: Histogram showing reconstructed decay length distribution of the K^0_S candidates.

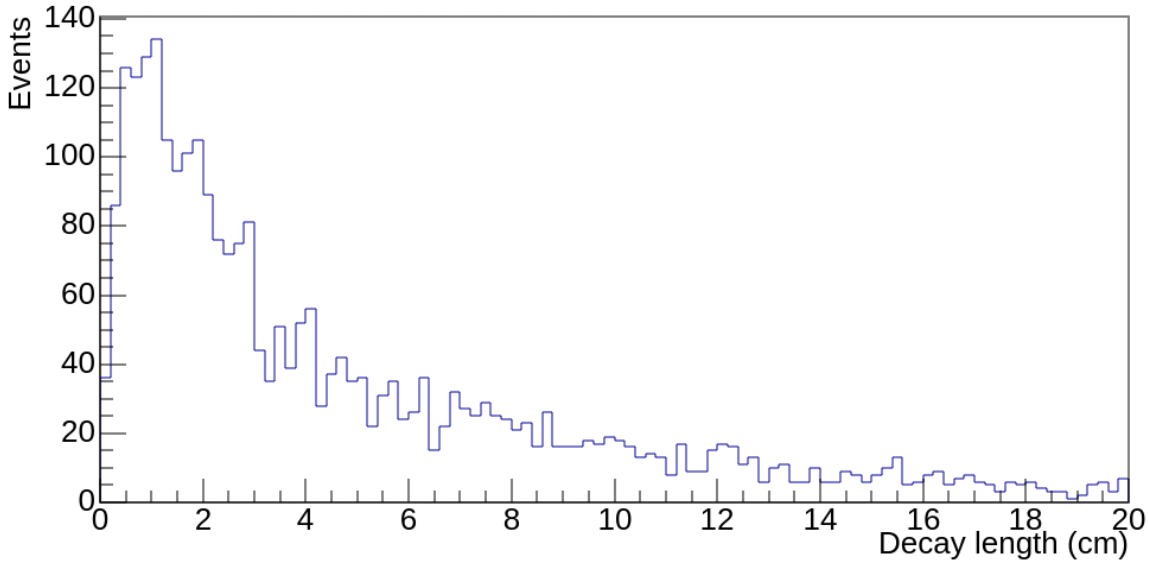

Fig. 3.14: Histogram showing reconstructed decay length distribution of the Λ candidates.

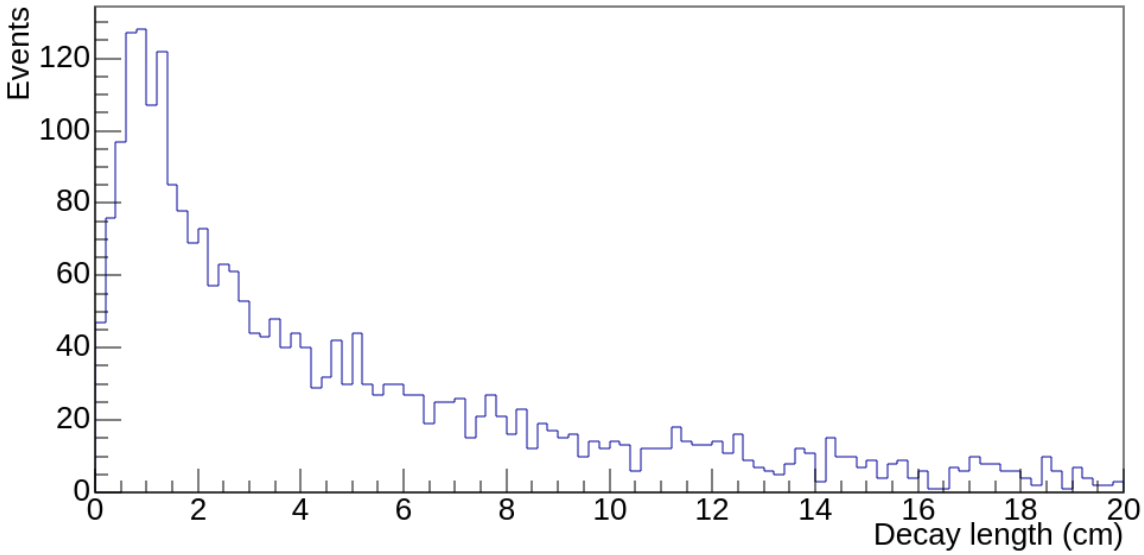

Fig. 3.15: Histogram showing reconstructed decay length distribution of the $\bar{\Lambda}$ candidates.

On the plots showing the reconstructed decay length for K^0_S (figure 3.13), Λ (figure 3.14) and $\bar{\Lambda}$ (figure 3.15) an approximate exponential decrease is observed at values larger than 1 cm. Lengths close to zero are unlikely due to the detector resolution. These plots provides direct information about the lifetime of the V^0-particles. Data for K^0 decreasing quickly, which is related to its short lifetime. At a distance of about 3 cm, begins an unusual change in the distribution, which may be associated with the secondary particle production in the beam-pipe wall of the 3.5 cm radius. In turn, the width of the Λ and $\bar{\Lambda}$ distributions are much longer, their tail reaching up to 20 cm, which is reasonable due to the fact that Λ's live is longer than K^0_S.

3.6 Armenteros-Podolanski plot

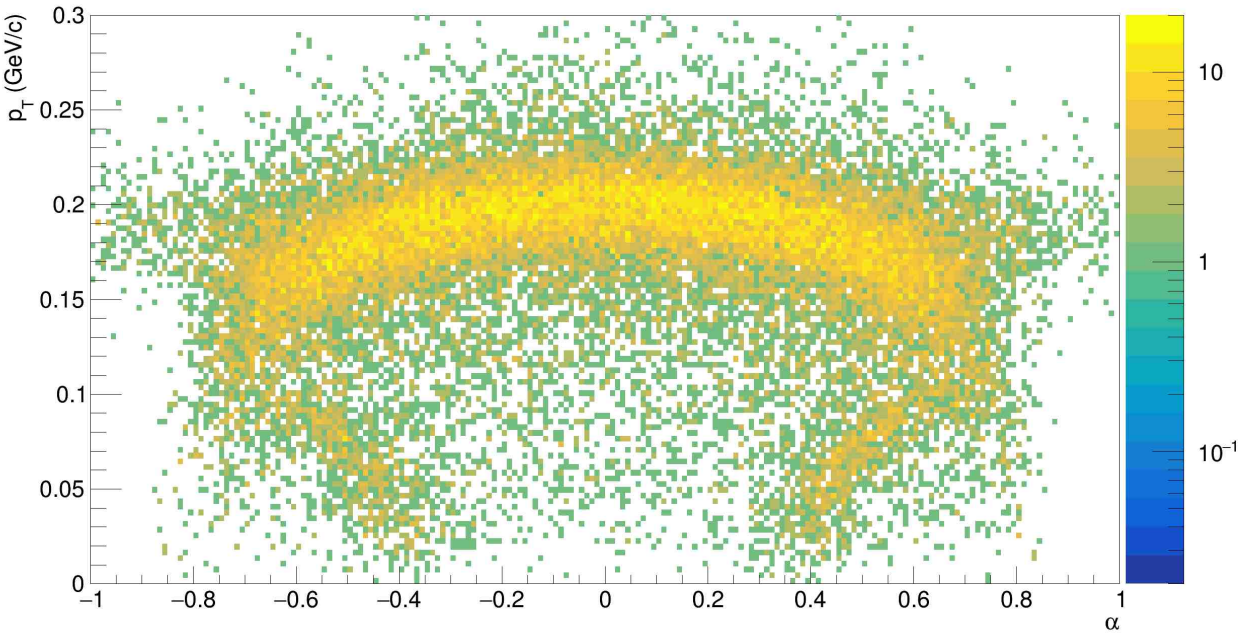

Fig. 3.16: The Armenteros-Podolanski plot.

The Figure 3.16 shows the Armenteros-Podolanski method of analysis of the V^0 decays. The plotted variables are explained in the 1.6 section. In the regions, where the ellipses representing K^0 and Λ ($\bar{\Lambda}$) overlaps, it is not possible to separate each other. We do not observe second halves for ellipses originating from Λ ($\bar{\Lambda}$) due to TPC geometrical acceptance.

The reconstructed distributions are smeared around the theoretical curves, this is associated with the uncertainty in determining the momentum of particles created in the decay. Which directly translates into the values of the V^0-candidate momentum. The α parameter, as described earlier, is a measure of the asymmetry of the longitudinal momenta of the decay products in the system where z axis is directed along the V^0-particle momentum. To do this, daughters momenta must be rotated appropriately.

3.7 Cosine of theta angle in CMS

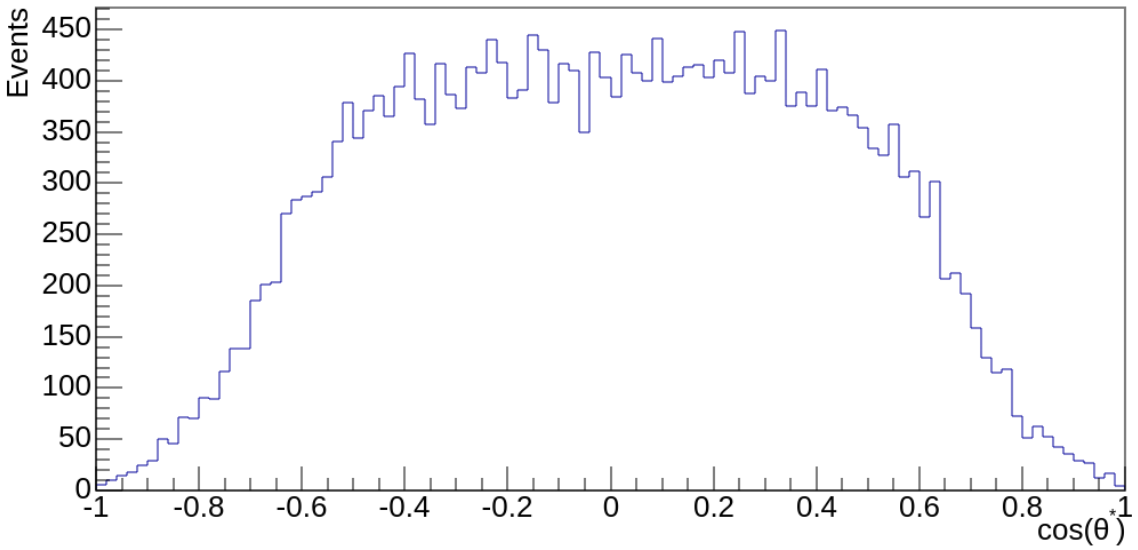

Fig. 3.17: Histogram of cosine of theta angle in the center of mass system (CMS) for K^0 candidates.

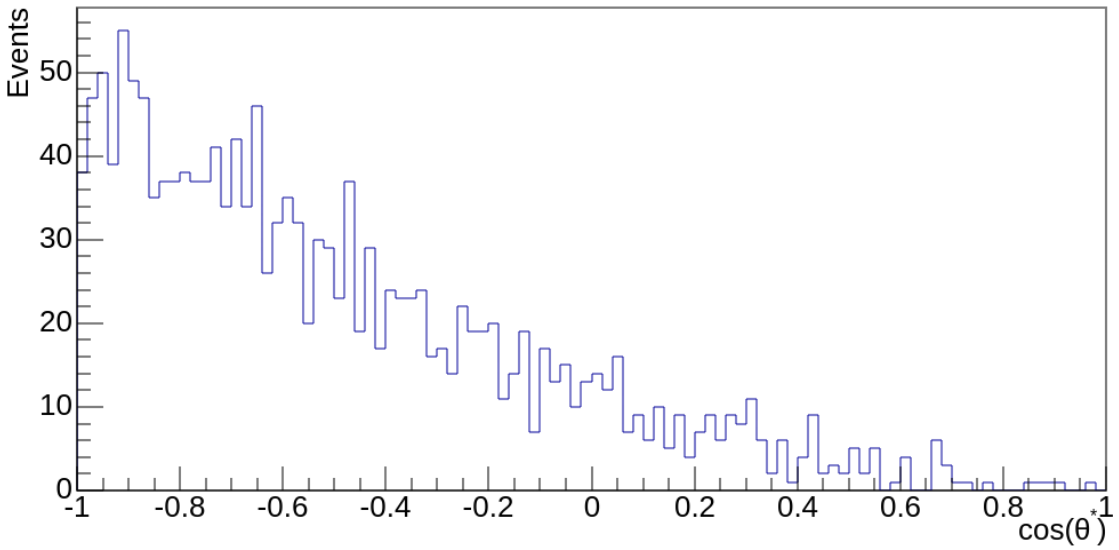

Fig. 3.18: Histogram of cosine of theta angle in the center of mass system (CMS) for Λ candidates.

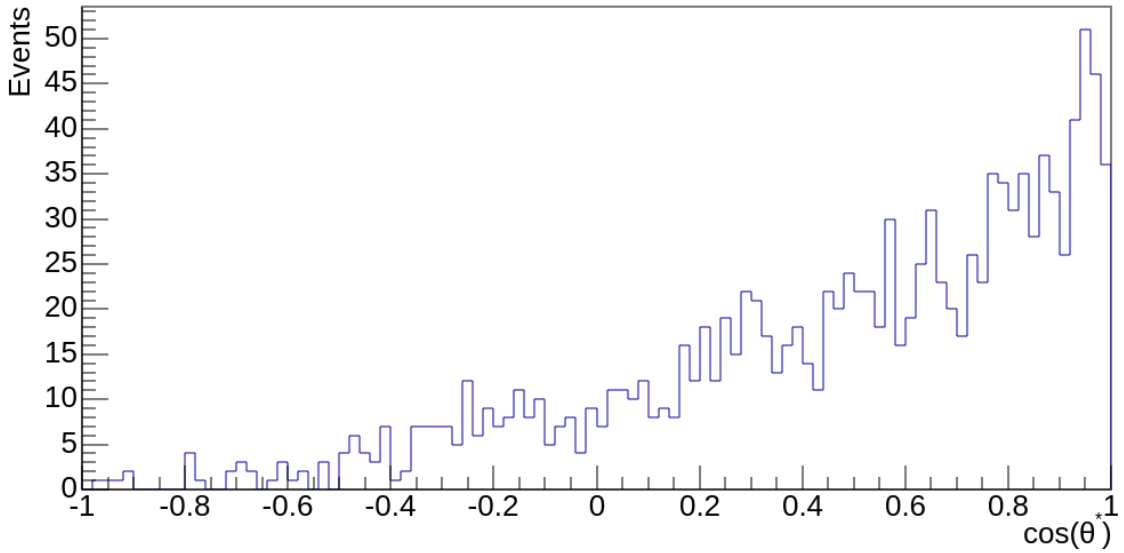

Fig. 3.19: Histogram of cosine of theta angle in the center of mass system (CMS) for $\bar{\Lambda}$ candidates.

The plots show cosine of theta angle in the center of mass system between the direction of the positive daughter momentum and the V^0-particle momentum for K^0_S (figure 3.17), Λ (figure 3.18) and $\bar{\Lambda}$ (figure 3.19) candidates.

There is a significant difference between the plot for K^0 and for Λ ($\bar{\Lambda}$). This distribution for K^0 results from isotropic decay of a particle with spin 0 into two particles with spin 1/2. As a result of isotropic decay, we obtain the distribution of the θ^* angle proportional to the sin(θ^*), so the distribution of the cos(θ^*) is uniform. The lack of events close to the angles $-\frac{\pi}{2}$ and $\frac{\pi}{2}$, results from the limited acceptance of the detector.

In the case of spin 1/2 baryons, Λ and $\bar{\Lambda}$, the decay is not isotropic. There is a tendency that proton or anti-proton moves in the opposite direction compared to Λ and $\bar{\Lambda}$ momentum.

Chapter 4

Summary

Detecting resonances based on information about their decay products is an extremely complex process. Starting from collecting measurement data through complex read-out electronics, selecting interesting events while rejecting the background, through the analysis of this data, up to the complicated theory of particle physics, describing what is happening on a subatomic scale, in a state of unimaginably high pressures, energies and momenta. Despite these difficulties, this process is not impossible, and humanity has made enormous progress in this regard. I sincerely hope that my work will contribute to expanding this field of science.

The original goal of the work has been achieved, and the code correctly supplements the data files containing events involving candidates for strange particles. The C++ based root framework used for this purpose allows this task to be performed in a satisfactorily fast manner, which is necessary for large data, which is obviously what we deal with in the analysis of experimental data of elementary particles.

In the second part of the work, I analyzed the data stored in two separate ROOT trees, one of them contained collection of candidates for K^0_S, the other for Λ's. The data were generated by simulation, and the conclusions from their analysis proofed that the work was done correctly. DCA of the V^0 to the beam line, DCA between daughters and cosine of pointing angle plots provide information about the expected distributions under the hypothesis that pairs of charged particle were produced at the same point by a decay of the particle pointing to the beam line. In turn, the distribution of the hypothetical decay length provides information about particle life-time, and the plot showing the invariant mass allows to identify the resonance based on data about its rest mass.

The histogram of the cosine of the polar angle in CMS should theoretically be a uniform distribution for spinless particles such as K^0_S, but detector effects lead to the lack of events at the edges of the distribution. In the case of spin 1/2 particles the decay is not isotropic. Proton and anti-protons tend to fly in the opposite direction to the Λ and $\bar{\Lambda}$ momentum, respectively.

Last but not least, the Armenteros-Podolanski plot shows distribution characteristic for decay of short-lived neutral particles.

Appendix A

Code merging the trees

A.1 Libraries

The content of the file **Includes.h**, which includes the necessary libraries for the correct compilation of the program.

```
# ifndef INCLUDES_H
# define INCLUDES_H

#include <iostream>
#include <string>
#include <utility>
#include <sstream>
#include <algorithm>
#include <stdio.h>
#include <stdlib.h>
#include <vector>
#include <fstream>
#include <cmath>
#include <cstdlib>
#include <sys/stat.h>
#include <iterator>
#include <ostream>
#include <iomanip>
#include <stdexcept>
#include <limits>
#include "TROOT.h"
#include "TSystem.h"
#include "TParticle.h"
#include "TParticlePDG.h"
#include "TThread.h"
#include "TFile.h"
#include "TTree.h"
#include "TChain.h"
#include "TH1D.h"
#include "TProfile.h"
#include <TH2.h>
#include <TF1.h>
#include <TF2.h>
#include <THStack.h>
#include <TStyle.h>
#include <TGraph.h>
#include <TGraph2D.h>
#include <TGraphErrors.h>
#include <TCanvas.h>
#include <TLegend.h>
#include <TGaxis.h>
#include <TString.h>
#include <TColor.h>
#include <TLine.h>
#include <TExec.h>
#include <TFitResultPtr.h>
#include <TFitResult.h>
#include <TLatex.h>
#include <TMath.h>
#include <TLorentzVector.h>
#include <TTreeReader.h>
#include <StRPEvent.h>
#include <StUPCRpsTrack.h>
#include <StUPCRpsTrackPoint.h>
#include <StUPCEvent.h>
#include <StUPCTrack.h>
#include <StUPCBemcCluster.h>
#include <StUPCVertex.h>
#include <StUPCTofHit.h>
#include "StUPCV0.h"
#include "ReadPicoLambdaK0.h"
#include <map>

# endif // INCLUDES_H
```

A.2 Class header file

Contents of the file **ReadPicoLambdaK0.h:**

```
#ifndef READ_PICO_LAMBDA_K0_H
#define READ_PICO_LAMBDA_K0_H

#include <Includes.h>

class ReadPicoLambdaK0 {
public:
    ReadPicoLambdaK0(TTree* tree_2);
    void ProcessData(Long64_t i, StUPCEvent* upcEvt, TTree* tree_1, TTree* tree_2);

    std::vector<Int_t> eventIdVectors;
    std::vector<Float_t> leadPtVectors;
    std::vector<Float_t> leadPhiVectors;
    std::vector<Float_t> leadEtaVectors;
    std::vector<Float_t> subleadPtVectors;
    std::vector<Float_t> subleadPhiVectors;
    std::vector<Float_t> subleadEtaVectors;
    std::vector<Float_t> p1PtVectors;
    std::vector<Float_t> p1PhiVectors;
    std::vector<Float_t> p1EtaVectors;
    std::vector<Int_t> p1ChVectors;
    std::vector<Int_t> p1HasTOFInfoVectors;
    std::vector<Float_t> p2PtVectors;
    std::vector<Float_t> p2PhiVectors;
    std::vector<Float_t> p2EtaVectors;
    std::vector<Int_t> p2HasTOFInfoVectors;
    std::vector<Int_t> pairChargeVectors;
    std::vector<Float_t> pairPhiVectors;
    std::vector<Float_t> pairEtaVectors;
    std::vector<Float_t> pairPtVectors;
    std::vector<Float_t> pairMassVectors;

private:
    Int_t eventId, p1_hasTOFinfo, p2_hasTOFinfo, pair_charge, p1_ch;
    Float_t lead_pt, lead_phi, lead_eta,
            sublead_pt, sublead_phi, sublead_eta,
            p1_pt, p1_phi, p1_eta,
            p2_pt, p2_phi, p2_eta,
            pair_phi, pair_eta, pair_pt, pair_mass;
    int j=0;
};

#endif  // READ_PICO_LAMBDA_K0_H
```

A.3 Class source file

Contents of the file **ReadPicoLambdaK0.cxx:**

```
#include "ReadPicoLambdaK0.h"

ReadPicoLambdaK0::ReadPicoLambdaK0(TTree* tree_2) {

    tree_2->SetBranchAddress("eventId", &eventId);
    tree_2->SetBranchAddress("lead_pt", &lead_pt);
    tree_2->SetBranchAddress("lead_phi", &lead_phi);
    tree_2->SetBranchAddress("lead_eta", &lead_eta);
    tree_2->SetBranchAddress("sublead_pt", &sublead_pt);
    tree_2->SetBranchAddress("sublead_phi", &sublead_phi);
    tree_2->SetBranchAddress("sublead_eta", &sublead_eta);
    tree_2->SetBranchAddress("p1_pt", &p1_pt);
    tree_2->SetBranchAddress("p1_phi", &p1_phi);
    tree_2->SetBranchAddress("p1_eta", &p1_eta);
    tree_2->SetBranchAddress("p1_ch", &p1_ch);
    tree_2->SetBranchAddress("p1_hasTOFinfo", &p1_hasTOFinfo);
    tree_2->SetBranchAddress("p2_pt", &p2_pt);
    tree_2->SetBranchAddress("p2_phi", &p2_phi);
    tree_2->SetBranchAddress("p2_eta", &p2_eta);
    tree_2->SetBranchAddress("p2_hasTOFinfo", &p2_hasTOFinfo);
    tree_2->SetBranchAddress("pair_charge", &pair_charge);
    tree_2->SetBranchAddress("pair_phi", &pair_phi);
    tree_2->SetBranchAddress("pair_eta", &pair_eta);
    tree_2->SetBranchAddress("pair_pt", &pair_pt);
    tree_2->SetBranchAddress("pair_mass", &pair_mass);
}

void ReadPicoLambdaK0::ProcessData(Long64_t i, StUPCEvent* upcEvt, TTree* tree_1, TTree*
    tree_2) {

    eventIdVectors.clear();
    leadPtVectors.clear();
    leadPhiVectors.clear();
    leadEtaVectors.clear();
    subleadPtVectors.clear();
    subleadPhiVectors.clear();
    subleadEtaVectors.clear();
    p1PtVectors.clear();
    p1PhiVectors.clear();
    p1EtaVectors.clear();
    p1ChVectors.clear();
    p1HasTOFInfoVectors.clear();
    p2PtVectors.clear();
    p2PhiVectors.clear();
    p2EtaVectors.clear();
    p2HasTOFInfoVectors.clear();
    pairChargeVectors.clear();
    pairPhiVectors.clear();
    pairEtaVectors.clear();
    pairPtVectors.clear();
    pairMassVectors.clear();

    tree_1->GetEntry(i);
    Long64_t Event1 = upcEvt->getEventNumber();

    for(int jj = j ; jj < tree_2->GetEntriesFast(); ++jj){
        tree_2->GetEntry(jj);
        Long64_t Event2 = eventId;
        if (Event1 == Event2){
```

```
59              ++j;
60              eventIdVectors.push_back(eventId);
61              leadPtVectors.push_back(lead_pt);
62              leadPhiVectors.push_back(lead_phi);
63              leadEtaVectors.push_back(lead_eta);
64              subleadPtVectors.push_back(sublead_pt);
65              subleadPhiVectors.push_back(sublead_phi);
66              subleadEtaVectors.push_back(sublead_eta);
67              p1PtVectors.push_back(p1_pt);
68              p1PhiVectors.push_back(p1_phi);
69              p1EtaVectors.push_back(p1_eta);
70              p1ChVectors.push_back(p1_ch);
71              p1HasTOFInfoVectors.push_back(p1_hasTOFinfo);
72              p2PtVectors.push_back(p2_pt);
73              p2PhiVectors.push_back(p2_phi);
74              p2EtaVectors.push_back(p2_eta);
75              p2HasTOFInfoVectors.push_back(p2_hasTOFinfo);
76              pairChargeVectors.push_back(pair_charge);
77              pairPhiVectors.push_back(pair_phi);
78              pairEtaVectors.push_back(pair_eta);
79              pairPtVectors.push_back(pair_pt);
80              pairMassVectors.push_back(pair_mass);
81          }
82          else if (Event1 < Event2){
83              break;
84          }
85          else if (Event1 > Event2){
86              continue;
87          }
88          else{
89              break;
90          }
91      }
92  }
```

A.4 Main function

Contents of the file **MergeK0withUPCDst.cxx:**

```
#include "Includes.h"

using namespace std;

int main(int argc, char** argv) {

    if (argc != 3) {
        cerr << "two input files required" << endl;
        return 1;
    }

    TFile file1(argv[1], "update");
    if (!file1.IsOpen()) {
        cerr << "Failed to open input file1." << endl;
        return 1;
    }

    TTree *UPCTree = static_cast<TTree*>(file1.Get("mUPCTree"));
    if (!UPCTree) {
        cerr << "Failed to retrieve mUPCTree tree." << endl;
        file1.Close(); return 1;
    }

    static StUPCEvent *upcEvt = 0x0;
    UPCTree->SetBranchAddress("mUPCEvent", &upcEvt);

    vector<Int_t> eventIdVectors;
    vector<Float_t> leadPtVectors, leadPhiVectors, leadEtaVectors;
    vector<Float_t> subleadPtVectors, subleadPhiVectors, subleadEtaVectors;
    vector<Float_t> p1PtVectors, p1PhiVectors, p1EtaVectors;
    vector<Int_t> p1ChVectors, p1HasTOFInfoVectors;
    vector<Float_t> p2PtVectors, p2PhiVectors, p2EtaVectors;
    vector<Int_t> p2HasTOFInfoVectors;
    vector<Int_t> pairChargeVectors;
    vector<Float_t> pairPhiVectors, pairEtaVectors, pairPtVectors, pairMassVectors;

    TBranch *beventIdVectors = UPCTree->Branch("eventIdVectors", &eventIdVectors);
    TBranch *bleadPtVectors = UPCTree->Branch("leadPtVectors", &leadPtVectors);
    TBranch *bleadPhiVectors = UPCTree->Branch("leadPhiVectors", &leadPhiVectors);
    TBranch *bleadEtaVectors = UPCTree->Branch("leadEtaVectors", &leadEtaVectors);
    TBranch *bsubleadPtVectors = UPCTree->Branch("subleadPtVectors", &subleadPtVectors);
    TBranch *bsubleadPhiVectors = UPCTree->Branch("subleadPhiVectors", &subleadPhiVectors);
    TBranch *bsubleadEtaVectors = UPCTree->Branch("subleadEtaVectors", &subleadEtaVectors);
    TBranch *bp1PtVectors = UPCTree->Branch("p1PtVectors", &p1PtVectors);
    TBranch *bp1PhiVectors = UPCTree->Branch("p1PhiVectors", &p1PhiVectors);
    TBranch *bp1EtaVectors = UPCTree->Branch("p1EtaVectors", &p1EtaVectors);
    TBranch *bp1ChVectors = UPCTree->Branch("p1ChVectors", &p1ChVectors);
    TBranch *bp1HasTOFInfoVectors = UPCTree->Branch("p1HasTOFInfoVectors",
                                                    &p1HasTOFInfoVectors);
    TBranch *bp2PtVectors = UPCTree->Branch("p2PtVectors", &p2PtVectors);
    TBranch *bp2PhiVectors = UPCTree->Branch("p2PhiVectors", &p2PhiVectors);
    TBranch *bp2EtaVectors = UPCTree->Branch("p2EtaVectors", &p2EtaVectors);
    TBranch *bp2HasTOFInfoVectors = UPCTree->Branch("p2HasTOFInfoVectors",
                                                    &p2HasTOFInfoVectors);
    TBranch *bpairChargeVectors = UPCTree->Branch("pairChargeVectors", &pairChargeVectors);
    TBranch *bpairPhiVectors = UPCTree->Branch("pairPhiVectors", &pairPhiVectors);
    TBranch *bpairEtaVectors = UPCTree->Branch("pairEtaVectors", &pairEtaVectors);
    TBranch *bpairPtVectors = UPCTree->Branch("pairPtVectors", &pairPtVectors);
    TBranch *bpairMassVectors = UPCTree->Branch("pairMassVectors", &pairMassVectors);
```

```
60
61      const char* inputFile2 = argv[2];
62      TFile* file2 = TFile::Open(inputFile2);
63      if (!file2) {
64          cerr << "Failed to open second input file2." << endl;
65          file1.Close();
66          return 1;
67      }
68
69      TTree* tree_2 = static_cast<TTree*>(file2->Get("ntp_K0s"));
70
71      ReadPicoLambdaK0 Read_K0(tree_2);
72
73      for (Long64_t i = 0; i < UPCTree->GetEntries(); ++i) {
74
75          Read_K0.ProcessData(i, upcEvt, UPCTree, tree_2);
76
77          eventIdVectors = Read_K0.eventIdVectors;
78          leadPtVectors = Read_K0.leadPtVectors;
79          leadPhiVectors = Read_K0.leadPhiVectors;
80          leadEtaVectors = Read_K0.leadEtaVectors;
81          subleadPtVectors = Read_K0.subleadPtVectors;
82          subleadPhiVectors = Read_K0.subleadPhiVectors;
83          subleadEtaVectors = Read_K0.subleadEtaVectors;
84          p1PtVectors = Read_K0.p1PtVectors;
85          p1PhiVectors = Read_K0.p1PhiVectors;
86          p1EtaVectors = Read_K0.p1EtaVectors;
87          p1ChVectors = Read_K0.p1ChVectors;
88          p1HasTOFInfoVectors = Read_K0.p1HasTOFInfoVectors;
89          p2PtVectors = Read_K0.p2PtVectors;
90          p2PhiVectors = Read_K0.p2PhiVectors;
91          p2EtaVectors = Read_K0.p2EtaVectors;
92          p2HasTOFInfoVectors = Read_K0.p2HasTOFInfoVectors;
93          pairChargeVectors = Read_K0.pairChargeVectors;
94          pairPhiVectors = Read_K0.pairPhiVectors;
95          pairEtaVectors = Read_K0.pairEtaVectors;
96          pairPtVectors = Read_K0.pairPtVectors;
97          pairMassVectors = Read_K0.pairMassVectors;
98
99          beventIdVectors->Fill();
100         bleadPtVectors->Fill();
101         bleadPhiVectors->Fill();
102         bleadEtaVectors->Fill();
103         bsubleadPtVectors->Fill();
104         bsubleadPhiVectors->Fill();
105         bsubleadEtaVectors->Fill();
106         bp1PtVectors->Fill();
107         bp1PhiVectors->Fill();
108         bp1EtaVectors->Fill();
109         bp1ChVectors->Fill();
110         bp1HasTOFInfoVectors->Fill();
111         bp2PtVectors->Fill();
112         bp2PhiVectors->Fill();
113         bp2EtaVectors->Fill();
114         bp2HasTOFInfoVectors->Fill();
115         bpairChargeVectors->Fill();
116         bpairPhiVectors->Fill();
117         bpairEtaVectors->Fill();
118         bpairPtVectors->Fill();
119         bpairMassVectors->Fill();
120     }
121
122     file1.cd();
```

```
123     UPCTree->Write("", TObject::kOverwrite);
124     file1.Close();
125     file2->Close();
126 
127     return 0;
128 }
```

A.5 Building script

```
1     #!/usr/bin/bash
2 
3     DIR="$( cd "$( dirname "${BASH_SOURCE[0]}" )" && pwd )"
4 
5     rm -r build/MergeK0withUPCDst
6     cd build
7 
8     g++ -g $DIR/MergeK0withUPCDst.cxx $DIR/src/ReadPicoLambdaK0.cxx \
9     -o MergeK0withUPCDst -Wl,--copy-dt-needed-entries `root-config \
10    --cflags` `root-config --libs` -I $DIR/../star-upc/include/ -I \
11    $DIR/include -L $DIR/../star-upc/build/ -lstar-upc
```

Appendix B

TTree as a ROOT data structure

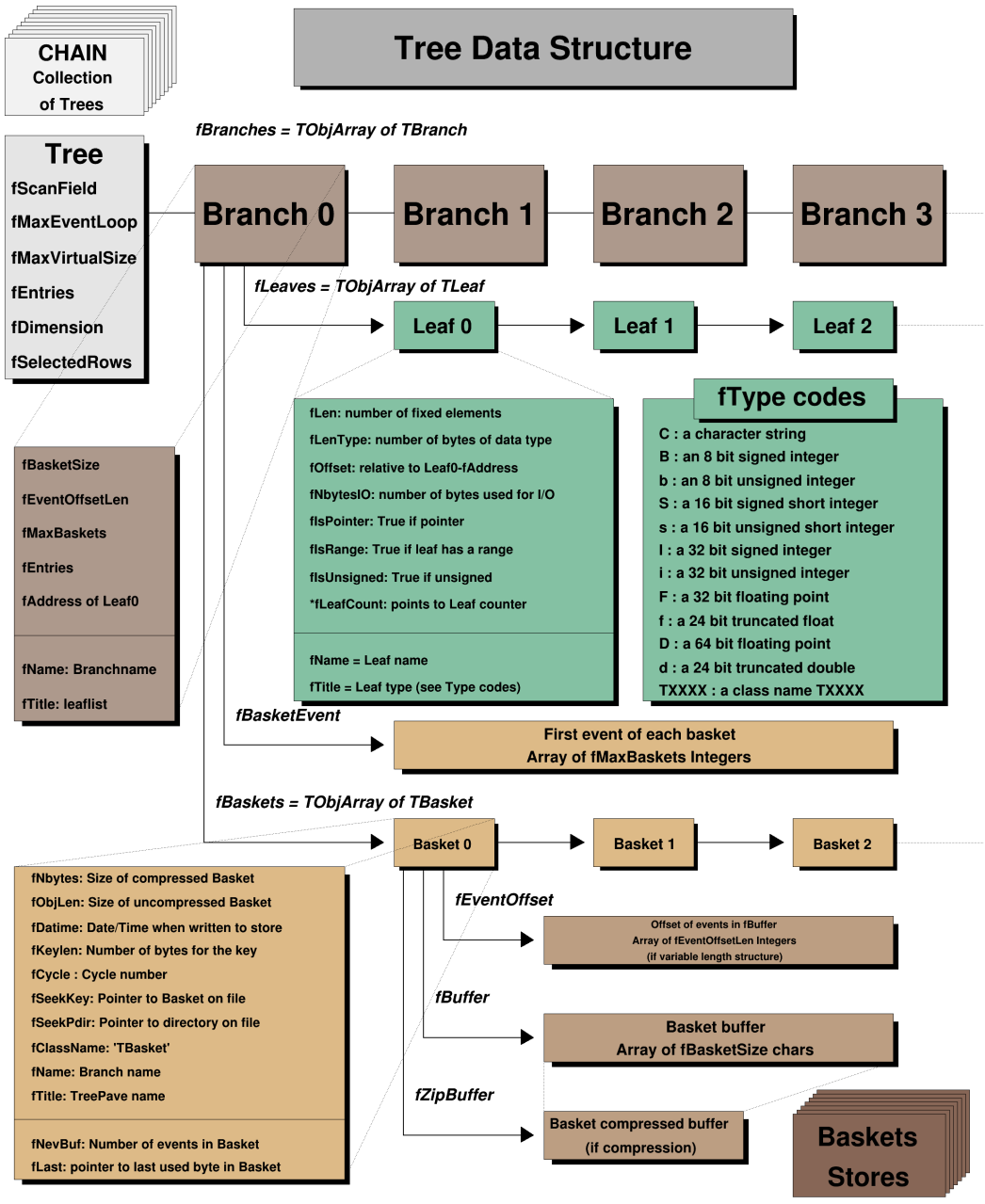

Fig. B.1: Diagram of the data structure TTree of the ROOT framework with connections [22].

The core components of the ROOT data structure are the TTree and TBranch. A TTree is a collection of TBranch objects. Each TBranch represents a ”column” of data, so to speak, which can be of various types and structures. Inside each TBranch, there are TLeaf objects. These leaves are the lowest level of the branch hierarchy, which are the variable, with a specific, predetermined data type. TBranches store their data in chunks called TBasket. These are serialized buffers that hold the compressed or uncompressed data of the branch. Baskets are used to optimize I/O operations by loading data in and out of memory efficiently. This is a kind of protection, because the tree is often too large to fit in memory, and sometimes (rarely) even a single TBranch is too large. This is what TBasket are for, which are ”batches” of data. TBasket is the smallest unit that can be read from TTree: if you want to read the first entry, you must read the first TBasket. The diagram also indicates variables such as fEntries, which is the number of entries (rows) in the TBranch, and fBasketSize, which is the size of the data buffer for each basket. the structure diagram of TFile and its component TTree is presented in the image B.2, while the TTree is presented in more detail in B.1.

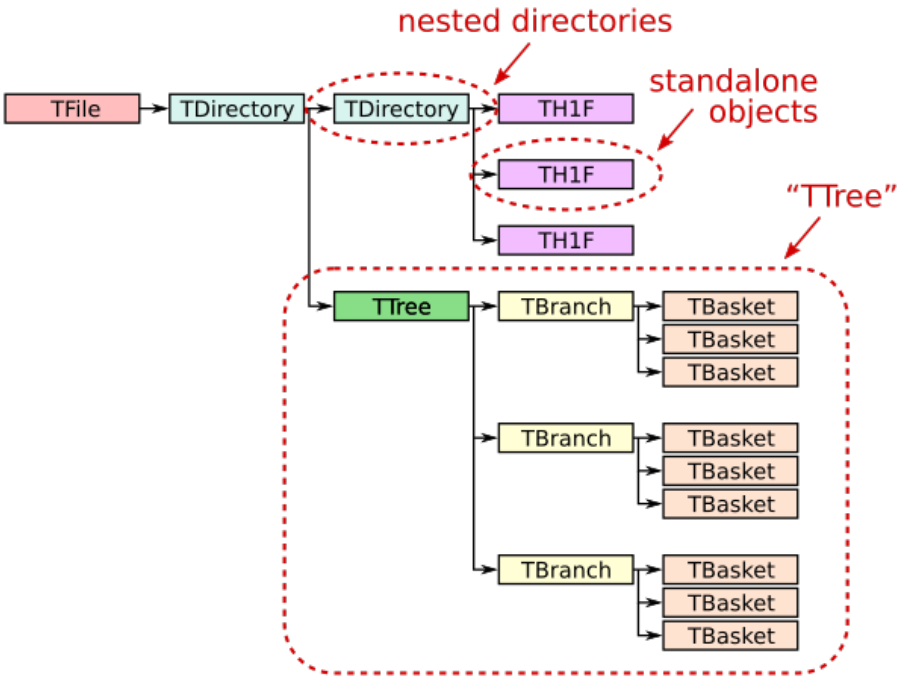

Fig. B.2: Data storage structure in ROOT: Hierarchy of files, trees and branches [23].

This structure allows ROOT to handle complex datasets efficiently, making it a powerful tool for data analysis in experiments.

If you come across specialized vocabulary whose meaning confuses you, visit the website CERN Glossary [24].

Appendix C

Pseudorapidity

In particle physics, measuring the traditional rapidity, which describes the relative speed of particles in collisions, encounters difficulties with relativistic particles. This requires accurate information about the particles' energy and momentum, which can be complicated. There is therefore a need to find a more practical solution, which resulted in the introduction of the concept of pseudorapidity. let's start with the definition of rapidity [25]:

$$y \equiv \frac{1}{2}\ln\left(\frac{E + p_z c}{E - p_z c}\right). \tag{C.1}$$

Following the relativistic equation

$$E^2 = (pc)^2 + (mc^2)^2, \tag{C.2}$$

and expand it using a Taylor series expansion, because for relativistic particles $pc >> mc^2$, we receive:

$$E = pc\left[1 + \left(\frac{mc^2}{pc}\right)^2\right]^{\frac{1}{2}} \approx pc + \frac{(mc^2)^2}{2pc} + \dots . \tag{C.3}$$

Putting this into the equation C.1 we receive:

$$y \approx \frac{1}{2}\ln\left(\frac{1 + \frac{p_z}{p} + \frac{(mc^2)^2}{2(pc)^2} + \dots}{1 - \frac{p_z}{p} + \frac{(mc^2)^2}{2(pc)^2} + \dots}\right). \tag{C.4}$$

The expression $\frac{p_z}{p} = \cos(\theta)$ describes the relationship between the component of momentum along the axis of collision (p_z) and the total momentum of the particle (p) which defines the cosine of the polar angle (θ) from the component axis to the direction of motion of the particle (it was assumed that this is the z axis). In the next step, I expand the cosine with the trigonometric identity $\cos(2\theta) = 2\cos^2(\frac{\theta}{2}) - 1 = 1 - 2\sin^2(\frac{\theta}{2})$, receiving:

$$y \approx \frac{1}{2}\ln\left(\frac{\cos^2\frac{\theta}{2}}{\sin^2\frac{\theta}{2}}\right) = -\ln\left(\tan\left(\frac{\theta}{2}\right)\right) \equiv \eta. \tag{C.5}$$

Pseudorapidity η is quicker and easier to estimate than rapidity, moreover simplifies analyses as it depends only on the polar angle, making it invaluable in experiments where direct measurement of energy and momentum is challenging.

List of Figures

1.1 View of the complex of RHIC [2] with the specific experiments and special objects marked. 2
1.2 Illustration showing the main components of the STAR detector [4]. 3
1.3 Illustration presenting an off-center collision of heavy ions, the interaction zone, is often called a "fireball" [7]. 5
1.4 The of evolution transforming to a QGP in a space-time diagram [8]. 6
1.11 Drawing showing the interaction of heavy ions in ultra-peripheral collisions [11]. The apparent flattening in the direction of flight in the laboratory-related system is due to Lorentz contraction. 7
1.10 An illustration of (a) an electromagnetic process where photons radiated by ions engage with each other, (b) a photon and nucleus interaction involving a photon from one ion interacting with a different nucleus, and (c) a reaction involving photons and nuclei that results in the disintegration of the nucleus due to photon transfer [12]. 7
1.12 Structure of proton. 8
1.13 PDF[5], particle has no internal structure (1), consists of three quarks (2), there is gluon interaction (3), including QCD effects (4) in the momentum fraction argument [14]. . . . 8
1.14 Schematic representation of processes in proton-proton collisions: (a) nondiffractive process $pp \to X$, and diffractive processes: (b) single dissociation $pp \to Xp$ or $pp \to pY$, (c) double dissociation $pp \to XY$, (d) central diffraction scattering $pp \to pXp$, where X and Y represents the dissociated proton or centrally produced hadronic system [16]. 9
1.15 Photo of the decay of strange particles into charged particles in a liquid hydrogen bubble chamber [17]. 10
1.16 The strange V^0-particle decay topology. P+(-) refers to a positively(negatively) charged daughters. The solid line is detectable by the TPC anode sector 60 cm away from the beam line. The dashed line is extrapolated from the solid line or reconstructed V^0-candidate. The topological cuts are shown in the plot which is from Hai Jiang's thesis [18]. 11
1.17 The Armenteros-Podolanski Plot showing the dependence of the transverse momentum on the α parameter with clear traces corresponding to the meson K^0_S and baryons: Λ and $\bar{\Lambda}$ [20]. 12

2.1 Tree structure of merged ROOT files. 15

3.1 Histogram showing Distance of Closest Approach of the V^0 to the beam line for K^0_S candidates. 18
3.2 Histogram showing Distance of Closest Approach of the V^0 to the beam line for Λ candidates. 18
3.3 Histogram showing Distance of Closest Approach of the V^0 to the beam line for $\bar{\Lambda}$ candidates. 19
3.4 Histogram showing Distance of Closest Approach between V^0 daughters for K^0_S candidates. 20
3.5 Histogram showing Distance of Closest Approach between V^0 daughters for Λ candidates. 20

3.6 Histogram showing Distance of Closest Approach between V^0 daughters for $\bar{\Lambda}$ candidates. 21
3.7 Histogram showing the distribution of the cosine of pointing angle for K^0_S candidates. . . 22
3.8 Histogram showing the distribution of the cosine of pointing angle for Λ candidates. . . . 22
3.9 Histogram showing the distribution of the cosine of pointing angle for $\bar{\Lambda}$ candidates. . . . 23
3.10 Histogram showing the invariant mass distribution of the K^0_S candidates. 24
3.11 Histogram showing the invariant mass distribution of the Λ candidates. 24
3.12 Histogram showing the invariant mass distribution of the $\bar{\Lambda}$ candidates. 25
3.13 Histogram showing reconstructed decay length distribution of the K^0_S candidates. 26
3.14 Histogram showing reconstructed decay length distribution of the Λ candidates. 26
3.15 Histogram showing reconstructed decay length distribution of the $\bar{\Lambda}$ candidates. 27
3.16 The Armenteros-Podolanski plot. 28
3.17 Histogram of cosine of theta angle in the center of mass system (CMS) for K^0 candidates. 29
3.18 Histogram of cosine of theta angle in the center of mass system (CMS) for Λ candidates. 29
3.19 Histogram of cosine of theta angle in the center of mass system (CMS) for $\bar{\Lambda}$ candidates. 30

B.1 Diagram of the data structure TTree of the ROOT framework with connections [22]. . . . 39
B.2 Data storage structure in ROOT: Hierarchy of files, trees and branches [23]. 40

Bibliography

[1] Brookhaven National Laboratory (BNL), "Spin Physics at the Relativistic Heavy Ion Collider," 2023. [Online; accessed 20-December-2023] https://www.bnl.gov/rhic/spin.php.

[2] Brookhaven National Laboratory (BNL), "Recalling Quark Matter '83 and the Birth of RHIC," 2021. [Online; accessed 20-December-2023] https://www.bnl.gov/newsroom/news.php?a=26204.

[3] U.S. Department of Energy, Office of Nuclear Physics, "RHIC Energies," 2023. [Online; accessed 20-December-2023] https://science.osti.gov/np/Facilities/User-Facilities/RHIC.

[4] D. Arkhipkin and J. Lauret, "STAR Online Meta-Data Collection Framework: Integration with the Pre-existing Controls Infrastructure," 2017.

[5] Brookhaven National Laboratory (BNL), "Introduction to STAR Physics," 2024. [Online; accessed 13-January-2024] https://www.star.bnl.gov/central/physics/.

[6] S. Klein, "Pomeron Physics," 2023. [Online; accessed 28-December-2023] https://www.star.bnl.gov/public/pec/periph_overview.html.

[7] U. Heinz, "The strongly coupled quark-gluon plasma created at RHIC," 2008.

[8] I. Gorbounov, "Pattern Recognition in the Inner Tracking System of HERA-B and Measurement of the V_0 Production Cross Section in pN Collisions," 2003. [Online; accessed 06-January-2024] https://www.star.bnl.gov/~gorbunov/main/.

[9] Brylinski, Wojciech (Warsaw U. of Tech.), "Calibration of the momentum scale of a particle physics detector using the Armenteros-Podolanski plot," 2024. [Online; accessed 29-December-2023] https://www.semanticscholar.org/reader/1666761ad3f4199dae3d3a6307c9edaddc9248db.

[10] I. G. Kaplan, "Pauli Exclusion Principle and its theoretical foundation," 2019.

[11] M. Kłusek-Gawenda, "Seminar on Central Collisions," 2023. [Online; accessed 29-December-2023] https://indico.koza.if.uj.edu.pl/event/10/contributions/806/attachments/237/320/MKG_UJ_6March.pdf.

[12] A. J. Baltz, et al., "The Physics of Ultraperipheral Collisions at the LHC," 2008.

[13] CODATA Task Group, "Fundamental Physical Constants," 2023. [Online; accessed 29-December-2023] https://physics.nist.gov/cuu/Constants/alpha.html.

[14] Forum, "How to Understand Proton Structure Function (PDF)," 2021. [Online; accessed 06-January-2024] https://www.physicsforums.com/threads/how-to-understand-proton-structure-function-pdf.1009775.

[15] L. Fulek, "Charged Particle Production in Diffractive Proton-Proton Scattering at the RHIC and LHC Energies," 2024.

[16] CMS Collaboration, V. Khachatryan, *et al.*, "Measurement of diffraction dissociation cross sections in pp collisions at $\sqrt{s} = 7$ TeV," 2015.

[17] H. D. Wahl. [Online; accessed 9-January-2024] http://www.hep.fsu.edu/~wahl/phy4822/expinfo/BC/bubchamber.html.

[18] H. Jiang, "Strange Hadron(K^0_S, Λ and Ξ) Production in d+Au Collisions at $\sqrt{s_{NN}} = 200\,GeV$ at rhic," 2005.

[19] J. Podolanski and R. Armenteros, "III. Analysis of V-events," 1954.

[20] M. Aleeksev, et al., "Collins and Sivers Transverse Spin Asymmetries for Pions and Kaons on Deuterons," 2008.

[21] P. B. Rodriguez, et al., "Calibration of the momentum scale of a particle physics detector using the Armenteros-Podolanski plot," 2024. [Online; accessed 29-December-2023] https://www.semanticscholar.org/reader/1666761ad3f4199dae3d3a6307c9edaddc9248db.

[22] CERN, "ROOT TTree Class Documentation," 2023. [Online; accessed 20-December-2023] https://root.cern.ch/doc/master/classTTree.html.

[23] The Carpentries, "ROOT TBasket Details," 2023. [Online; accessed 29-December-2023] https://hsf-training.github.io/hsf-training-scikit-hep-webpage/03-ttree-details/index.html.

[24] CERN, "Glossary," 2023. [Online; accessed 20-December-2023] https://public-archive.web.cern.ch/en/Science/Glossary-en.php.

[25] Brookhaven National Laboratory (BNL), "Rapidity and Particle Physics," 2023. [Online; accessed 29-December-2023] https://www.phenix.bnl.gov/WWW/publish/seto/tutorials/abbotkinematics.pdf.