Universidade Técnica de Lisboa
Instituto Superior Técnico

Λ, $\bar{\Lambda}$ and K^0_S Production in In-In and In-Si Collisions at 158 GeV/nucleon

Átila Alves Neves
(Licenciado)

Dissertação para obtenção do Grau de Doutor em Física

Orientador:	Doutor João Carlos Carvalho de Sá Seixas (IST)
Co-Orientador:	Doutor Carlos Manuel de Assunção Lopes Lourenço (CERN)
Júri:	
Presidente:	Reitor da Universidade Técnica de Lisboa
Vogais:	Doutor Robert Jan Veenhof (CERN)
	Doutor Carlos Manuel de Assunção Lopes Lourenço (CERN)
	Doutor Jorge Venceslau Comprido Dias de Deus (IST)
	Doutora Maria Paula Frazão Bordalo e Sá (IST)
	Doutor João Carlos Carvalho de Sá Seixas (IST)

Junho de 2007

Universidade Técnica de Lisboa
Instituto Superior Técnico

Λ, $\bar{\Lambda}$ and K^0_S Production in In-In and In-Si Collisions at 158 GeV/nucleon

Átila Alves Neves
(Licenciado)

Dissertação para obtenção do Grau de Doutor em Física

Orientador:	Doutor João Carlos Carvalho de Sá Seixas (IST)
Co-Orientador:	Doutor Carlos Manuel de Assunção Lopes Lourenço (CERN)
Júri:	
Presidente:	Reitor da Universidade Técnica de Lisboa
Vogais:	Doutor Robert Jan Veenhof (CERN)
	Doutor Carlos Manuel de Assunção Lopes Lourenço (CERN)
	Doutor Jorge Venceslau Comprido Dias de Deus (IST)
	Doutora Maria Paula Frazão Bordalo e Sá (IST)
	Doutor João Carlos Carvalho de Sá Seixas (IST)

Junho de 2007

Abstract

This thesis presents results on Λ and K^0_S production in In-In and In-Si collisions at 158 GeV per nucleon at the NA60 experiment at the CERN SPS. It shows the methods used to reconstruct the decays of the studied particles, which include strict data selection cuts to improve the signal to background ratio enough for them to be visible, as well as the subsequent background subtraction techniques. The necessary acceptance and efficiency corrections and their calculations are described, and transverse momentum and rapidity spectra presented for both particles.

The transverse momentum distributions were found to be consistent with a scenario of collective flow affecting the temperature in a mass dependent way. No significant changes are seen in the T inverse p_{T} slope parameter with collision centrality.

The K^0_S rapidity distribution is peaked at mid-rapidity and was well represented by a gaussian. The Λ rapidity distribution, however, is flat at mid-rapidity with a drop in yield at 3.5–4 units of rapidity. This is consistent with the superposition of the two main Λ production methods: associated production and baryon/anti-baryon pair production.

An attempt was made to measure the $\bar{\Lambda}/\Lambda$ ratio, which was found to be in agreement with previous measurements of this quantity in Pb-Pb collisions at 158 GeV per nucleon.

Keywords

Ultra-relativistic heavy ion collisions; strangeness producion; data acquisition and online monitoring; radiation-tolerant silicon pixel detectors; Λ hyperon; K^0_S meson

Resumo

Esta tese apresenta resultados sobre a produção de Λ e K^0_S em colisões In-In e In-Si a 158 GeV por nucleão na experiência NA60 no acelerador SPS do CERN. Ela mostra os métodos utilizados para reconstruir os decaimentos das partículas estudadas, que incluem os fortes cortes de selecção de dados para fazer com que a razão sinal/ruído se eleve o suficiente para que se tornem visíveis, assim como as subsequentes técnicas de subtracção de ruído de fundo. As correcções necessárias de aceitância e eficiência e seus cálculos são descritos, e os espectros de momento transverso e rapidez são apresentados para as duas partículas.

As distribuições de momento transverso são compatíveis com um cenário de movimento colectivo afectando a temperatura de uma forma que depende da massa da partícula estudada. Não foram detectadas mudanças de temperatura no que toca à centralidade da colisão.

A distribuição de rapidez do K^0_S exibe um pico a rapidez zero no referencial do centro de massa e é bem representada por uma curva gaussiana. A distribuição do Λ, no entanto, é uniforme nessa mesma região com uma queda a 3.5–4 unidades de rapidez. Este resultado é consistente com a sobreposição dos dois métodos de produção dessa partícula: produção associada e produção de um par barião/anti-barião.

Foi também feita uma tentativa de medir a razão $\bar{\Lambda}/\Lambda$, resultando num valor que está de acordo com resultados anteriores medidos em colisões Pb-Pb a 158 GeV por nucleão.

Palavras Chave

Colisões ultra-relativísticas de iões pesados; produção de estranheza; aquisição de dados e monitoramento em linha (*online*); detectores de píxeis de silício resistentes a radiação; hiperão Λ; mesão K^0_S

Acknowledgements

First and foremost I would like to thank my parents, without whom I would not even exist. To say that is all they did would do them a great disservice, however, since they have provided the love and support needed for all of my endeavours, and this thesis is certainly no exception. So Mum, Dad, I owe you all that I am and then some.

Great thanks go out to Rob Veenhof for supervising most of the work presented here. I could not have finished without him. The three Pedros provided witty banter during the day and were always there for the technical difficulties as well. Thanks also to Ruben Shahoyan for always having time to share his knowledge and expertise, to Sanja Damjanovic for office chat and helpful Physics explanations, and to Hans Specht as well.

Finally, I would like to thank Carlos Lourenço and João Seixas for giving me the opportunity to join NA60 and become a PhD student.

This work was supported financially by the Portuguese Fundação para a Ciência e Tecnologia (FCT) and Fundo Social Europeu (FSE) under grant SFRH/BD/5087/2001.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Relativistic Heavy Ion Collisions

Relativistic heavy ion collisions provide a way to investigate nuclear matter under extreme conditions of temperature and density. The main goal of this sort of study is to search for experimental evidence of a state of matter called Quark Gluon Plasma (QGP) believed to only have existed in nature in the early universe, or inside neutron stars. The search for QGP formation has justified the existence of a considerable number of experiments over several decades at BNL and CERN.

Ordinary hadronic matter is composed of gluon-bound states of three quarks, three anti-quarks or a quark/anti-quark pair, which correspond to baryons, anti-baryons and mesons, respectively. These quarks and gluons have never been observed in isolation due to the nature of the coupling constant α_s in QCD, which increases with distance. Its higher value as the strongly interacting particles are further away from each other causes confinement. At shorter distances the strong interaction is weaker, and this property is called *asymptotic freedom*. Quark confinement is explained phenomenologically by the MIT Bag Model [1], where the quarks are kept together by negative vacuum pressure.

Lattice QCD calculations predict that, once the temperature or density of the system are high enough [2], hadronic matter should undergo a phase transition to a state in which its constituent quarks and gluons are quasi free and no longer bound, analogous to electrons in a metal (see Fig. 1.1). In this new state, chiral symmetry should also be approximately restored. The threshold temperature for QGP formation is predicted to be between 150 and 200 MeV in lattice QCD. The exact value depends on lattice model details such as the number of quark flavours and colours, or the quark masses.

Once formed, detecting the plasma is not a straightforward task since the observed particles have subsequently undergone a hadronisation process. Plasma formation signatures that can be detected after the hadronisation

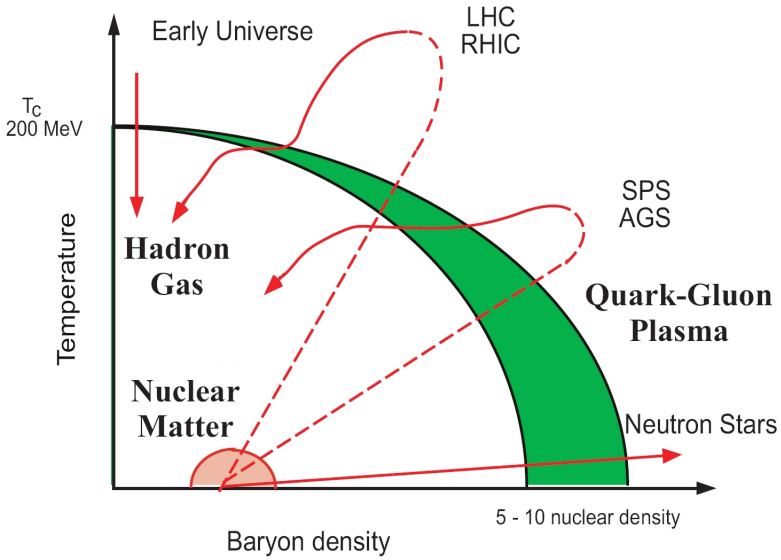


Figure 1.1: Phase diagram of the strongly interacting matter as a function of the baryon density and the temperature.

phase are then needed. Some of these include:

- Suppression of charmonium states due to colour charge screening.
- Emission of thermal dileptons and photons by the quarks and gluons in the plasma.
- Modification of the spectral function of the ρ meson due to (approximate) restoration of chiral symmetry.
- Strangeness enhancement due to a lower $s\bar{s}$ quark pair production threshold in the deconfined state.

The NA38 and NA50 experiments probed mainly the charmonium suppression signature by measuring J/ψ yields. The suppression found in p-A and S-U collisions can be attributed to the absorption of the J/ψ by normal nuclear matter while in Pb-Pb collisions it was found [3] that J/ψ production experienced an additional suppression with increasing collision centrality. The observed suppression exceeds what would be expected from extrapolations based on proton-nucleus data and hints at the presence of an onset. While there is no consensus on the origin of the anomalous suppression, current theoretical models (in particular, lattice calculations) suggest that the directly produced J/ψs survive up to T $\sim$ 1.5–2 T_c (presumably not reachable on the SPS) and that the onset of the anomalous suppression is

associated with the dissolution of the more fragile ψ' and χ_c mesons, which account for a significant fraction ($\sim$30–40%) of the observed J/ψ yield in pp collisions, through feed-down decays.

A dimuon excess was also found in the mass region between the ϕ and J/ψ that increases with the number of participant nucleons [4] in A-A collisions, despite the accurate description of p-A data by the superposition of Drell-Yan and simultaneous semi-leptonic decays of pairs of D mesons. This excess can be attributed to thermal dimuons indicative of the formation of a QGP phase [5, 6], although there are also less quantitative explanations by higher twist Drell-Yan contributions [7].

The CERES experiment found that in Pb-Au collisions at 158 GeV per nucleon, there was an excess in the measured dielectron mass spectrum at masses above 200 MeV when compared to the expected sources, known as the *hadronic cocktail* [8]. These expected sources, however, accurately describe the proton-nucleus data, as shown in Fig. 1.2. In the 300–700 MeV mass range, where the excess is most pronounced, it exceeds the hadronic cocktail by a factor of almost 6. The results have been taken to be an indication of changes in the mass and decay width of the ρ meson due to partial chiral symmetry restoration.

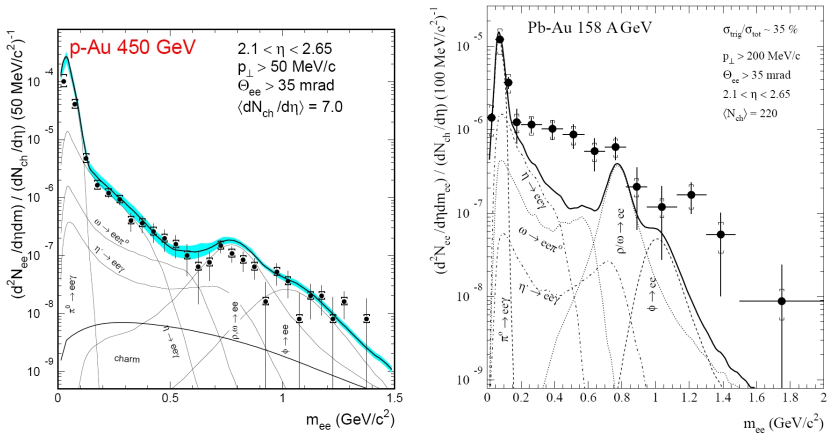


Figure 1.2: Dielectron mass spectrum measured by CERES in p-Au (left) and Pb-Au (right) collisions. Figures taken from [9] and [8].

The WA97 experiment reported an increase in the yield of strange and multi-strange baryons and anti-baryons in Pb-Pb collisions at 160 GeV per nucleon in comparison with p-Be data [10]. This enhancement was also found to increase with the strangeness content of the particle under study, which would be yet another indication of QGP formation at the SPS.

1.2 The NA60 Experiment

The NA60 experiment was conceived to answer some questions left open by previous experiments by providing new and more accurate measurements of a certain number of observables [11, 12]. Some of the questions it attempts to answer are:

- Is the intermediate mass dimuon excess observed in nuclear collisions due to thermal dimuons or increased charm production?
- Are the properties of the ρ meson really modified in heavy ion collisions and, if so, is this an indication of chiral symmetry restoration?
- What is the variable that governs the onset of charmonia suppression?

To help clarify these issues, NA60 collected dimuon data from indium-indium collisions at 158 GeV per nucleon over a period of 5 weeks in 2003. To establish a robust baseline to which the nucleus-nucleus data can be compared, as well as allowing for the study of the nuclear dependence of particle production, there were also proton-nucleus data taking periods in 2002 and 2004, at beam energies of 400 and 158 GeV. Various targets were used in the proton data, ranging from Be to U.

Thanks to the development of radiation tolerant high granularity silicon detectors, NA60 is able to track the charged particles produced in a high energy heavy ion collision and select amongst them the ones that correspond to the muons detected further downstream in the muon spectrometer. Measuring the muons close to the production point means an increase in the signal to background ratio and improved mass resolution, leading to more accurate results than the ones previously available. It also means that by measuring the displacement of the muon tracks, prompt dimuons can be distinguished from muon pairs from simultaneous $D\bar{D}$ decays in order to investigate the cause of the intermediate mass region dimuon excess.

1.3 Work Presented

The main part of this thesis is devoted to the study of the production of the strange quark carrying Λ hyperon and K^0_S meson in indium-indium and indium-silicon collisions at 158 GeV per nucleon. Strange particle production in heavy ion collisions is extremely interesting since strangeness enhancement is a predicted signature of quark gluon plasma formation. Strange quarks are suppressed relative to the lighter quarks in hadronic matter due to their higher mass. In a QGP, $s\bar{s}$ pairs can be formed at the energy cost of just twice the strange quark's mass, since there is no need for them to be in a confined hadron state. This enhanced strangeness content can then be detected in the hadrons that emerge from the fireball.

Baryon and anti-baryon production in relativistic nuclear collisions is also interesting independently of strangeness content because it is sensitive to two fundamental aspects of hadron production dynamics that are not yet fully understood: baryon/anti-baryon pair production and the transport of baryon number from beam rapidity to mid-rapidity. Both of these aspects are subject of intense theoretical and experimental interest.

A detailed explanation of the detectors in the NA60 apparatus and all of their components is presented in Chapter 2. Chapter 3 presents the data acquisiton system used by the collaboration, the logical structures used to store the data and the online monitoring and data reading framework developed by the author. Data reconstruction and the selection of events under study are discussed in Chapter 4, and the analysis of the selected data along with transverse momenta and rapidity spectra are presented in Chapter 5. Finally, the results are discussed and commented on in Chapter 6.

Chapter 2

The NA60 Apparatus

2.1 Overview

The NA60 experiment measures dimuon production in proton-nucleus and heavy ion collisions, which enables it to study vector meson production, the Drell-Yan process, thermal dimuons, and open charm production. This last study has been made possible for the first time in heavy-ion physics with the advent of radiation hard high-granularity silicon detectors. See Fig. 2.1 for a schematic illustration of the apparatus.

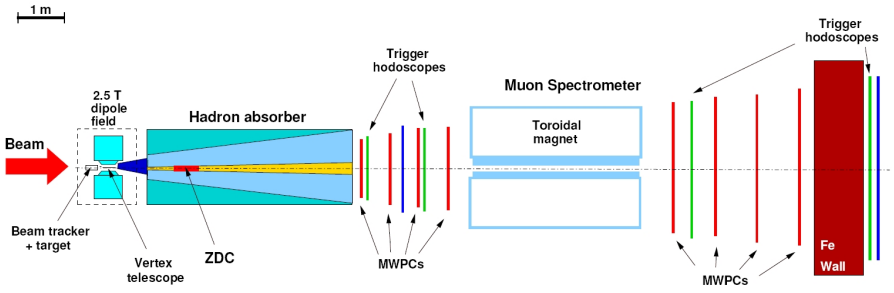


Figure 2.1: The NA60 apparatus.

NA60, being the successor to three other experiments (NA10, NA38 and NA50, in chronological order), inherits some of its detectors from its predecessors. One such detector is the Muon Spectrometer, responsible for identifying and tracking the muons, as well as supplying the highly selective trigger that allows NA60 to study processes with a very low production cross-section. Although the tracking elements of the Muon Spectrometer can track any charged particle, the particles that eventually make it to the detector are considered to be muons due to an absorber placed before the spectrometer. Since only muons are capable of passing through the absorber, they are the particles that produce a trigger and are subsequently tracked. This absorber, however, degrades the mass resolution of the detector since

it causes the muons to suffer energy loss and multiple scattering.

Another detector NA60 inherits from NA50 is the Zero Degree Calorimeter, which determines the centrality of the collision. It does so by measuring the energy deposited by the spectator beam nucleons.

To improve the momentum and position measurements, a new detector, called the Vertex Telescope, was introduced in NA60 to determine these variables before the effects of the absorber. It measures the kinematics of all the charged particles produced in the target region, and finds the muons responsible for the trigger by matching them to the variables measured in the Muon Spectrometer, both in position and momentum space. The Vertex Telescope is made of several silicon tracking Planes, that were mostly silicon strip detectors in the 2002 proton run and silicon pixel detectors in the 2003 indium run. The 2004 setup included detectors of both types.

The other main new detector in NA60 is the Beam Tracker, which is also a silicon detector. It is placed upstream of the target, and measures the transverse coordinates of the incoming beam particles in order to determine the impact parameter of the muons. This makes it possible to determine if the muons come from the primary interaction vertex or from the decay of a long lived hadron.

All of these detectors, as well as the beam and the target system, are described in greater detail below.

2.2 The Beam

The NA60 experiment is located in the ECN3 hall of the CERN SPS (the so-called North Area High Intensity Facility, NAHIF). Figure 2.2 shows an overview of the beam line, along with its collimators and bending and focusing magnets.

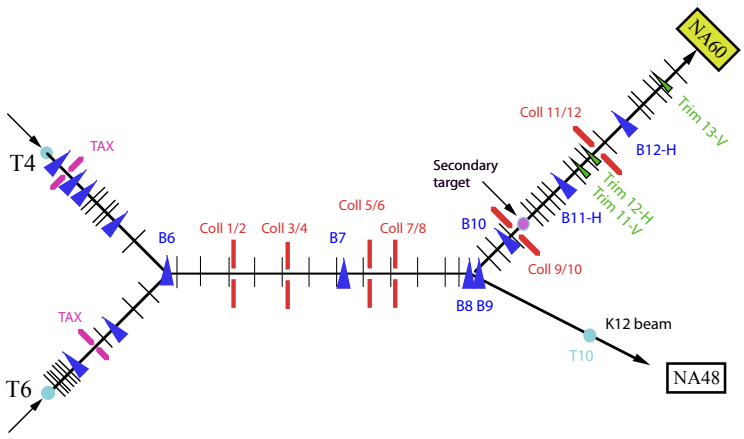


Figure 2.2: The NA60 beam line.

The experiment has taken data from both proton and ion runs in different data taking periods, with the proton beam having an energy of 400 GeV (in 2002 and 2004) or 158 GeV (in 2004) and the indium beam an energy of 158 GeV per nucleon. The energy of the indium beam was chosen so that a direct comparison can be made to the NA50 Pb-Pb data without having to rescale the energy, given that the Pb beam energy in that experiment was also 158 GeV per nucleon (corresponding to 400 GeV per charged nucleon, for a Pb ion). The beam intensities were between $\sim 2 \cdot 10^8$ (2002) and $\sim 2 \cdot 10^9$ (2004) protons per 4.8 s burst for the proton beam and $\sim 5 \cdot 10^7$ ions per 5.2 s burst for the indium beam.

The proton beam intensity was measured by three argon counters (known in the experiment as the "argonia"), which are multi-foil ionisation chambers. These detectors are placed a few metres upstream of the target and give a signal that is proportional to the integrated incident flux of one burst. The argon counters were calibrated during the October 2001 comissioning run with the aid of a scintillator detector at a low beam intensity ($\sim 10^6$ protons per burst). The need to operate at low beam intensities is due to the limitations of the scintillator detector. The linearity of the argon counters has been verified in the scope of the NA38 and NA50 experiments, up to the beam intensities used in the NA60 experiment.

2.3 The Target System

In NA60, instead of having just one thick target, a collection of sub-targets is used. The reason for using sub-targets is twofold: ion fragments from one collision might re-interact in a thick target and produce triggerable muons, and the produced muons would undergo multiple scattering.

Different target systems were used for the proton and ion runs. The proton target system used in 2004 is composed of nine sub-targets, ranging from 2 mm to 4 mm thickness, spaced roughly 1 cm apart from each other. The sub-targets span 12 mm in diameter except for the most downstream one, made out of lead, which has a radius of 1 cm. In order to study the nuclear dependence of particles such as the ϕ, ω, χ_c and D mesons, there are seven different target materials. The first sub-target is made out of aluminium, followed by targets made out of uranium, tungsten, copper, indium, then three berillium targets, and finally a lead target.

The ion run target system is composed of seven 1.5 mm thick In sub-targets, spaced by roughly 7.5 mm. The first sub-target is 12 mm in diameter (to fully cover the transverse beam profile); the other six span only 1 mm. These sub-targets are significantly smaller to minimise the number of reinteractions from fragments produced in upstream collisions. Due to their small diameter, however, they have to be aligned with the beam axis with a precision of ~ 100 μm. Also, it is necessary to place the sub-targets

in vacuum ($\sim$ 0.01 atm) to avoid interactions between the air and the ion beam.

The target system is mounted on a platform that can slide in and out of the beam by a fixed distance without changing the alignment of the other components in the vertex region.

Figure 2.3 shows the target region during the 2003 In run, with the target box placed between the beam tracker (on the right) and the vertex telescope (on the left).

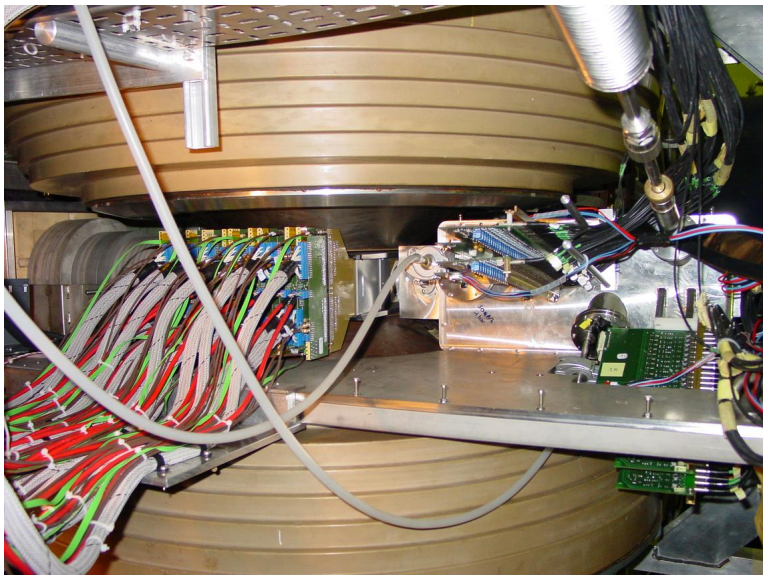


Figure 2.3: The NA60 target region in 2003.

2.4 The Muon Spectrometer

The Muon Spectrometer is responsible for providing the dimuon trigger, as well as measuring the kinematics of the muons that reach it, filtering out all the other particles produced in the target region. It was originally built for the NA10 experiment, and was also used by NA38 and NA50, experiments that preceded NA60. It consists of the following components:

- A hadron absorber, to let only muons reach the detector
- A toroidal magnet, to bend the muons' trajectories
- Eight multi-wire proportional chambers (MWPCs) to measure the muon tracks
- Four trigger scintillator hodoscopes

The alignment of these components is important to assure the quality of the data. Every new period of data-taking starts with a special alignment run, with no magnetic field and a low intensity beam. Typical displacements, from year to year, are of the order of 0.1 mm.

The spectrometer, schematically shown in Fig. 2.4, starts with the hadron absorber, followed by half of the trigger hodoscopes and MWPCs, the magnet, and the other half of the chambers and hodoscopes. These are all discussed in further detail below.

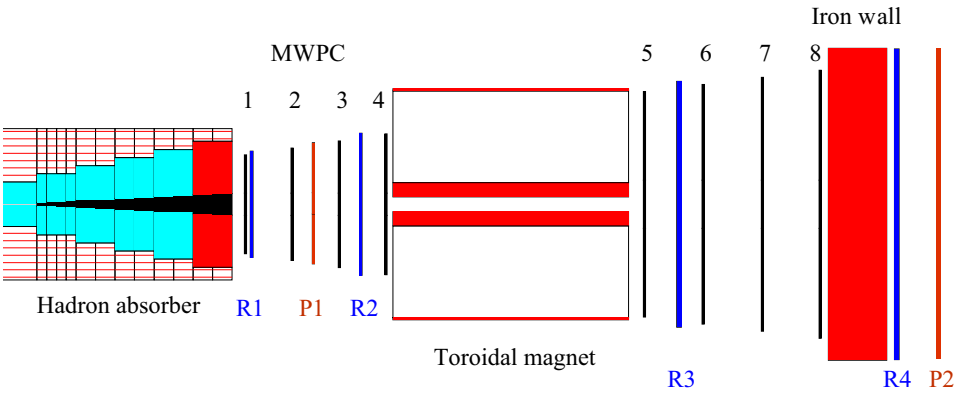


Figure 2.4: The Muon Spectrometer.

2.4.1 The Hadron Absorber

The hadron absorber's function is to ensure that only muons make it to the muon spectrometer, working as a particle identification system. Any particle that manages to punch through the absorber is assumed to be a muon.

At the centre of the absorber is the beam plug, made of tungsten and uranium and responsible for stopping the non-interacting beam particles. To prevent triggers from interactions in this beam plug, it starts roughly 2 m downstream from the target.

The rest of the hadron absorber is separated into two parts: the pre-absorber, made of BeO and Al_2O_3, and the main absorber, made of graphite and iron. The main absorber is surrounded by iron and concrete shielding. The materials were chosen from low atomic number elements to minimise the multiple scattering and energy loss suffered by the muons (see Table 2.1). This means that some energetic hadrons are able to punch through the absorber and cross the trigger hodoscopes. To prevent this from happening, the last hodoscope is placed behind a 1.2 m thick iron wall that absorbs any remaining hadrons and ensures a clean trigger. This wall does not, however, affect the dimuon mass resolution since it is placed after the last MWPC, and thus does not affect the tracking accuracy.

The pre-absorber starts as close as possible to the target, right after the vertex spectrometer, to decrease the yield of muons from the in-flight decays of pions and kaons (by absorbing these mesons), which are the main source of background.

material	ρ (g/cm^3)	Δz (cm)	λ_I (cm)	L/λ_I
BeO	2.81	41.0	35.85	1.14
Al_2O_3	3.52	25.4	32.65	0.78
C	1.93	440	44.70	9.84
Fe	7.87	40	16.76	2.39
Total				14.15

Table 2.1: Hadron absorber material composition.

2.4.2 The ACM Toroidal Magnet

The ACM (Air-Core Magnet) is a 4.8 m long toroid, centered at approximately 11 m downstream from the target. Its magnetic field is produced from 6 radial 4 m long iron poles, which induced the hexagonal geometry of the hodoscopes and chambers. This separates the acceptance into sextants (see Fig. 2.5). These iron poles cover 18° in azimuth, and muons crossing them are excluded from the offline analysis to avoid degrading the dimuon mass resolution (due to multiple scattering). With a total radius of 205 cm, the active part starts at an inner radius of 29.5 cm and stops at an outer radius of 154 cm, values that determine the angular acceptance of the experiment.

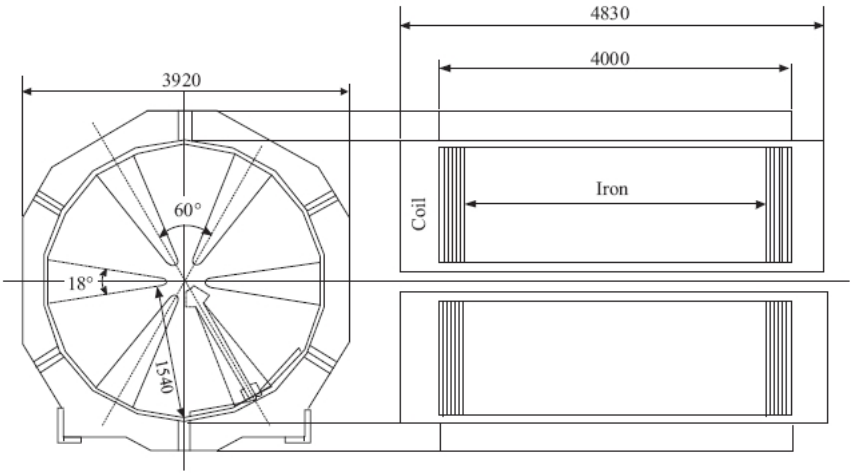


Figure 2.5: Frontal (left) and lateral (right) view of the ACM.

The field dependence is given by $\vec{B} = (B_0/r) \cdot \vec{e}_\phi$ and is essentially azimuthal. For a current of 4000 A (there were also runs taken with 6500 A), $B_0 = 0.219$ Tm. Having the axis of the magnet aligned with the beam axis causes the muons to keep their azimuthal angle. The deflection is therefore only in the polar angle θ, which for ds (where s is the curvilinear coordinate along the muon's path), is given by

$$\mathrm{d}\theta = \frac{\mathrm{d}s}{\rho(s)} = \frac{q \cdot B(r)}{p}\ \mathrm{d}s \quad ,$$

where ρ is the radius of the curvature of the helicoidal path of a particle of charge q and momentum p in a magnetic field B. For energetic particles the deflection is small and s can be replaced by $r/\sin(\theta)$, leading to

$$\Delta\theta = \int \frac{q \cdot B(r)}{p} \frac{\mathrm{d}r}{\sin(\theta)} = \frac{qB_0}{p_T}\ \log(\frac{z_2}{z_1}) \quad ,$$

with z_1 and z_2 being the entrance and exit planes, respectively, of the magnetic field, with respect to the target position. This last expression means that, for a given magnet current, the deflection angle is inversely proportional to the muon's p_T. Figure 2.6 shows the field strength as a function of the distance from the target, for five different distances from the beam line, in the transverse plane.

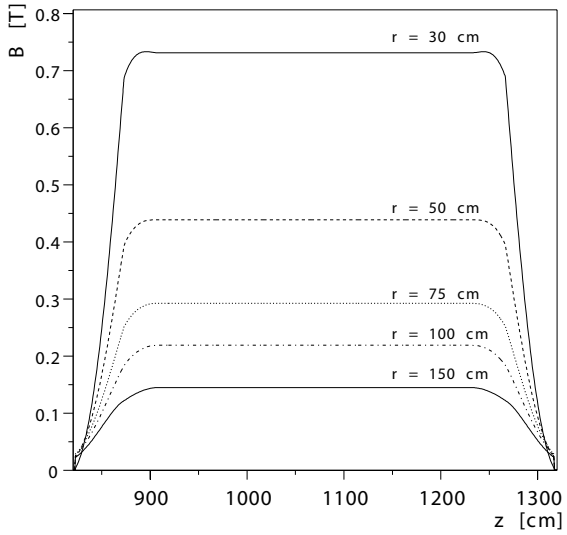


Figure 2.6: The ACM field strength.

The magnet current is pulsed and synchronised to the SPS cycle. In order to prevent any systematic effects arising from the sign of the magnetic field, the polarity was reversed every few runs.

2.4.3 The Trigger Hodoscopes

The NA60 trigger is provided by 4 hodoscopes, with the R_1 and R_2 hodoscopes placed upstream of the magnet and the R_3 and R_4 hodoscopes downstream of it. All of the hodoscopes have a hexagonal shape, and are split into 6 sextants, each of them having a certain number of scintillator slabs (30 for R_1 and R_2, 23 for R_3 and 32 for R_4) parallel to the outer edges. The R3 hodoscope has a small inactive area on one side (see Fig. 2.7), which slightly limits the acceptance in sextants 4 and 5.

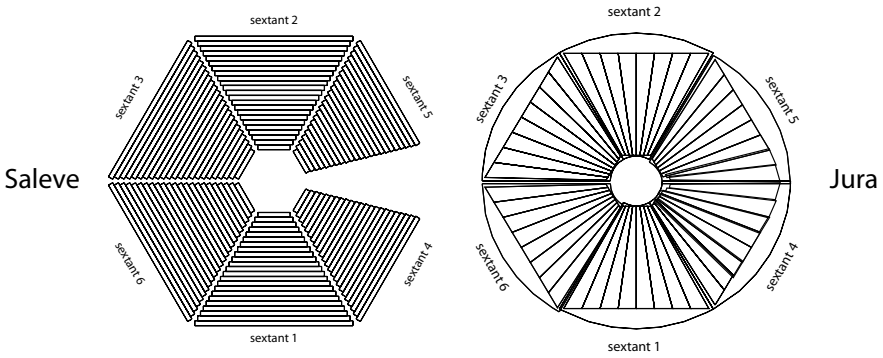


Figure 2.7: The R3 (left) and P1 (right) trigger hodoscopes.

Since the scintillators are read from only one end, in NA10 there were time fluctuations in the trigger (*jitter*), since the time it took for the light to reach the photomultiplier depends on where the muon hits the slab. To prevent this from happening, in NA38 the R_2 hodoscope was inverted with respect to its position in NA10, having its slabs read out from the opposite end of the corresponding R_1 slabs. Both signals are then sent to a "mean timer", avoiding the problem by making the trigger quite insensitive to specific muon trajectories (the jitter is reduced to ~ 3 ns).

The width of the R_1 and R_2 slabs follows a geometric progression, increasing radially. This means that in these two hodoscopes, slabs with the same index i, with i ranging from 1 to 30, correspond to the same polar angle. A particle that hits slab i in R_1 will also hit slab i in R_2. The single muon trigger condition requires the existence of the $V_i = R_1^i \times R_2^i \; or \; R_1^i \times R_2^{i-1}$ coincidence. The second condition for the V_i coincidence is to account for the finite extension of the target and multiple scattering of the muons in the hadron absorber. Also, the slabs in R_1 and R_2 have to belong to the same sextant. The dimuon trigger condition requires two V_i coincidences in different sextants.

The R_3 and R_4 hodoscope slabs have a fixed width of 5.5 cm. The $V_i \times R_4^j$ coincidence allows us to measure the deflection of the muon in the magnetic field, and thus its p_T, as seen above. The R_3 hodoscope provides redundant

information that makes it possible to reduce the number of triggers due to random coincidences between uncorrelated tracks. A special processor checks for the allowed $V \times R_3 \times R_4$ values and assigns a p_T bin to the muon.

The muon spectrometer also has two extra hodoscopes: P_1, placed upstream of the magnet, and P_2, placed behind the iron wall. These hodoscopes are only used in special runs to measure the efficiency of the R-based trigger system. The P hodoscopes, although also having a hexagonal shape, differ from their R counterparts in the orientation of their scintillator slabs. In the P hodoscopes, the slabs extend radially from the center to the outer edges of the sextants.

2.4.4 The Multi-Wire Proportional Chambers

The muons are tracked in the 8 MWPCs, 4 of which lie upstream of the magnet, with the remainder correspondingly downstream of the ACM. Like the hodoscopes, they have a hexagonal shape, and their transverse size increases with the distance to the target to cover the angular acceptance.

The chambers consist of 3 tracking planes, rotated at 60° with respect to each other, measuring the y, u and v coordinates (see Fig. 2.8). Each of these tracking planes have two graphited Mylar cathode planes, with a wire anode plane in between, 6 mm away from the cathode planes. The wire chambers contain 20 μm diameter gold-plated tungsten wires separated from each other by 3 mm.

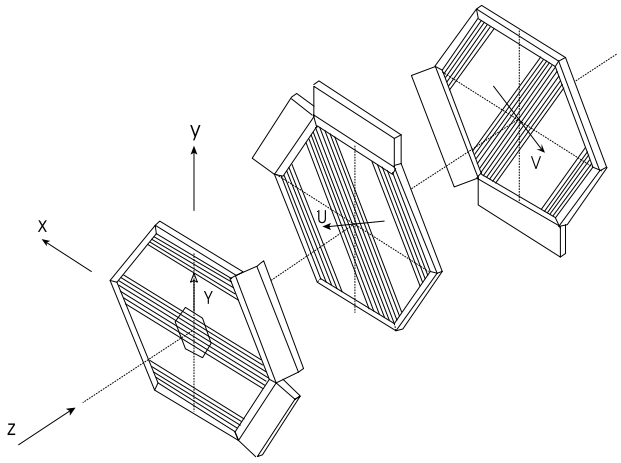


Figure 2.8: A multi-wire proportional chamber. The planes are shown far apart for the sake of visibility

The volume of the chambers is filled with a gas mixture consisting of $\sim$ 80% argon, of which 50% is flushed through 0.8% of isopropyl alcohol

(2-propanol), 18% isobutane iC_4H_10 used as a quencher, and 2% tetrafluorethane used as a "cleaning gas" [13]. A gas mixer rack measures and controls the flow of each of the gas components. An electro-valve is responsible for inserting each of the components in the pressure regulator, which are then transported to a mass flow controller (MFC).

The MFC measures the flow with respect to an external reference value. The difference is amplified and used to regulate the gas control valve, which ensures the correct mixture. Further mixing is needed to ensure that this mixture is homogeneous. This is achieved by passing the gases through a mixer tube. Metal strips in this tube cause turbulences that induce the necessary homogeneity.

Two gas distribution racks are used to supply each of the 24 separate chambers with the gas mixture. The gas distributors also measure the input and output flow to detect any possible leaks. There are also detector heads in the racks with the same purpose. The chambers are flushed with only argon during the periods when the beam is off, which can be done independently for each chamber.

2.5 The Vertex Telescope

The Vertex Telescope is used in the experiment to track the charged particles produced by the beam colliding with the sub-targets and match them to the muons measured downstream in the Muon Spectrometer, by matching the trajectories and momenta. This improves the dimuon mass resolution by significantly decreasing the influence of the smearing induced by the multiple scattering and energy loss fluctuations in the hadron absorber.

The information from the Vertex Telescope is also used to determine the transverse offset of the muons, with an accuracy of ~ 40 μm. This makes it possible to distinguish *prompt* dimuons, i.e. dimuons from the primary interaction vertex, from muons from decays of long-lived hadrons. This should help clarify the excess of dimuons measured in the mass region between the ϕ and the J/ψ in S-U and Pb-Pb collisions [4].

The Vertex Telescope is composed of either silicon strip detectors, in proton runs, or silicon pixel detectors, in ion runs. The low charge particle multiplicities in proton runs allow the use of the microstrip detectors, but only silicon pixel technology has the necessary granularity to track the hundreds of particles produced in a heavy ion collision. In the 2004 proton run, a hybrid setup was used, with both pixel and strip planes.

2.5.1 The Silicon Pixel Telescope

The Pixel Telescope is made up of 16 independent detector planes, of which there are 8 small and 8 large planes (see Fig. 2.9).

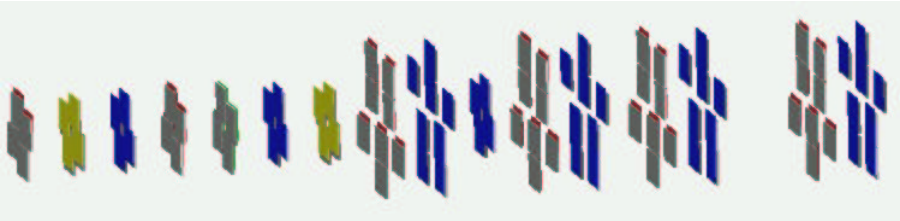


Figure 2.9: The Pixel Telescope used in the 2003 In-In run.

The planes are placed along the beam axis, the first of them at 6.6 cm downstream from the target, and the last at 31.2 cm. The smaller planes are placed closer to the target to match the Muon Spectrometer's angular acceptance, which is $35 < \theta < 120$ mrad or $3 < \eta < 4$ in pseudorapidity, in the laboratory reference frame. There is, however, one small plane placed after the first two large planes to improve the muon acceptance coverage by covering the large planes' beam hole.

Each plane consists of several single-chip pixel assemblies mounted on a planar ceramic support, made of Al_2O_3, known as a *hybrid*. The small planes have 4 such assemblies, whereas the large ones have 8 (see Fig. 2.10). In the large plane case, two hybrids are mounted back to back to form one 16-chip logical plane. The ceramic support is glued and wire bonded to a PCB that routes all the signals between the front-end electronics and the DAQ, control and power systems, as well as serving as mechanical support. The circuit boards, mounted on aluminium frames, fit into slots of a support box in the PT7 magnet gap, that defines the position of the planes relative to the target system. In the 2002 and 2003 runs, the modules were cooled by water at ~ 12°C that flowed through a copper tube attached to the back of the hybrid. In the 2004 run, the water was replaced by a liquid freon and the vertex telescope was placed in a dry atmosphere, inside a closed bag continuously flushed with dry nitrogen, so as to be operated at lower temperatures, and reduce the leakage current from the (heavily irradiated) silicon sensors.

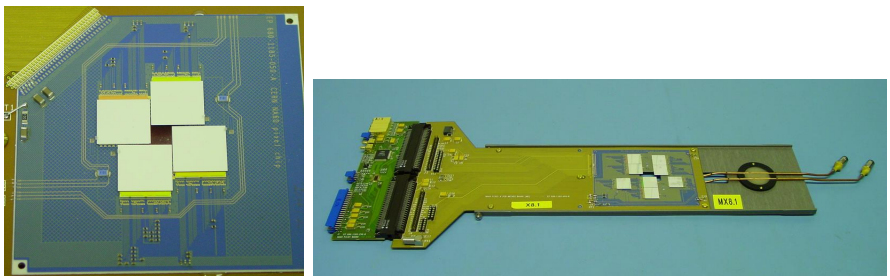


Figure 2.10: A zoomed view of a 4-chip plane (left), and a fully assembled 8-chip plane, with its cooling tube (right).

One of the requirements of the readout electronics is that it is sufficiently resistant to radiation damage, a task accomplished by the ALICE1LHCb pixel readout chip [14]. They have been shown to remain functional after exposure to a radiation dose of at least 12 MRad [15]. These 750 μm thick chips are bump bonded to the 300 μm thick silicon pixel sensor. This combination of readout chip and pixel sensor constitutes the pixel assemblies mentioned above. The active part of the sensor is a 32×256 matrix of pixel cells measuring 425×50 μm^2 each.

Since each cell is rectangular instead of square shaped, they measure one of the two transverse coordinates better than the other, depending on their orientation. To increase the accuracy of the measurements, the small planes have X and Y orientations that differ by a 90° angle rotation, and are defined by the direction of the 50 μm side. To maximise the acceptance coverage, some of the planes are inverted in the vertical sense, with their backs to the beam. These planes are designated as X' and Y'. All of the large planes are of the X kind, yielding better measurements in the direction perpendicular to the magnetic field lines. This is done in order to have a better momentum resolution.

Before the construction of the whole pixel telescope, some studies were done with 3 pixel planes in the October 2002 Pb-Pb run, with 3 Pb subtargets. The spatial resolution along the 50 μm pixel direction was found to be ~ 10 μm [16]. These 3 planes were used to determine the coordinates of the collision vertices. Figure 2.11 shows the z-vertex distribution, in which one can clearly distinguish the three lead targets, as well as the Beam Tracker exit cryostat window. The distributions for the 3 targets can be described as the convolution of the density functions of the targets and the gaussian resolution of the vertex reconstruction. Deconvolution leads to reconstruction resolutions of ~ 200 μm for the most downstream target and ~ 500 μm for the most upstream one. The error becomes larger for the most upstream target due to multiple scattering and to the increase of the extrapolation distance. The precision of the vertex reconstruction improves with the number of tracks.

Figure 2.11 also shows the correlation between the x vertex coordinate measured by the pixel planes and by the Beam Tracker. The correlation width is approximately 30 μm, including the vertex resolution of the pixels and the tracking resolution of the Beam Tracker. Assuming that the Beam Tracker resolution is 20 μm, the transverse vertex resolution of this limited pixel telescope was found to be approximately 20 μm for minimum-bias collisions. As in the case of the z-vertex, the resolution improves with the centrality of the collision.

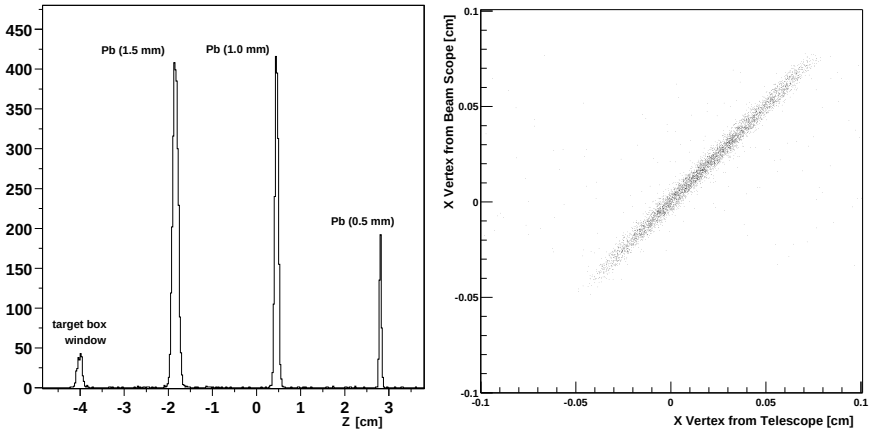


Figure 2.11: z-vertex distribution measured with the pixels (left), and the correlation between the x-vertex measurements of the pixels and the Beam Tracker (right)

2.5.2 The Microstrip Telescope

Although the original NA60 proposal described a vertex telescope with only pixel planes, delays in the delivery of the pixel assemblies led the collaboration to come up with an alternative for the 2002 proton run. Given the lower multiplicities in p-A collisions, it was possible to use silicon microstrip technology instead of the pixels to make up the Vertex Telescope. This technology also has some advantages, although they come with a set of drawbacks.

The first of these advantages is that the material budget is smaller for the Microstrip Telescope. Its sixteen 300 μm thick silicon sensors mean that any particles that traverse it go through 5.1% radiation lengths, compared to the Pixel Telescope's 34%.

The Microstrip Telescope can also provide good geometrical coverage at 40 cm from the target. To obtain the same acceptance as a microstrip plane at that distance, one would need roughly sixty pixel assemblies.

While the ALICE1LHCB pixel readout chips integrate digitised hits over 200 ns, the Microstrip Telescope uses 25 ns analogue sampling. This is very important at beam intensities of more than 10^8 protons per second, for it makes it possible to reject "pile-up" events.

Contrary to what happens in a digital chip, such as the ones found in the Pixel Telescope, where the discriminator threshold has to be carefully set *a priori*, in an analogue readout the full information of the deposited charge is stored, with no online cuts. This makes it possible to adjust the parameters governing hit identification *a posteriori* so as to achieve the best performance possible. The drawback is that extracting the hits from the sensor can be quite complicated.

The other disadvantage of analogue read-out is that the analogue data have to be digitised. Since all the information is kept, the data size can be quite large. This problem is overcome in NA60 by compressing the data, but this also entails complicated algorithms to read the compressed data.

Each microstrip sensor measures the position of the hits in one dimension. In order to obtain both transverse coordinates, each microstrip *station* consists of 2 sensors at angles 25° and −25° with respect to the y axis (see Fig. 2.12). The angles were chosen to strike a balance between the accuracies with which it would be possible to measure the track position and curvature [17].

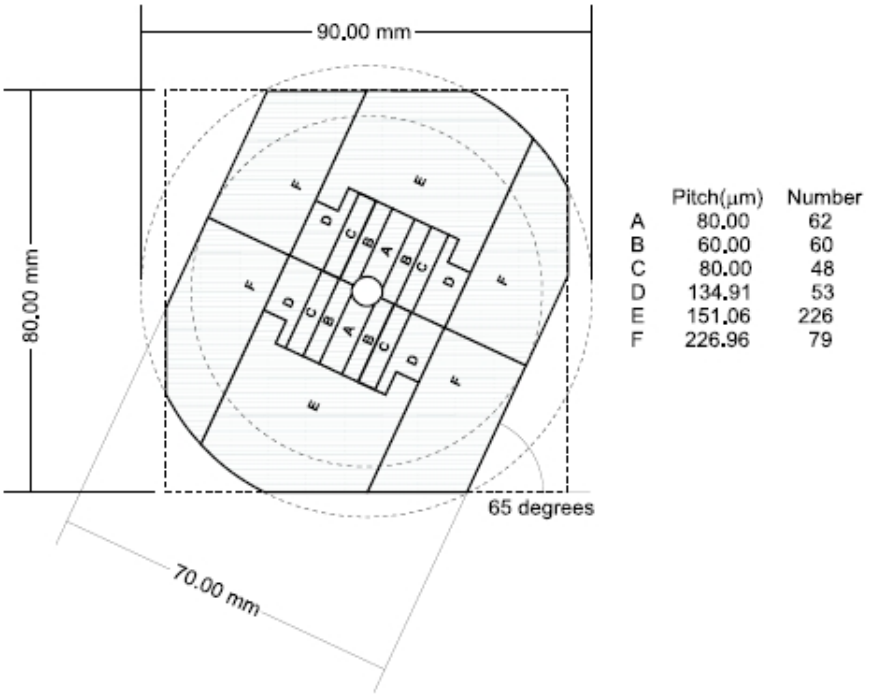


Figure 2.12: A microstrip sensor, showing how the pitch varies from region to region.

The sensors were designed so that the occupancy should remain constant across all its strips and below 3% even in p-Pb collisions. Given that the track density should be higher close to the beam axis, each sensor has several regions with different strip pitch and length, as can be seen in Fig. 2.12.

2.5.3 The PT7 magnet

The PT7 is a dipole magnet (see Fig. 2.13) used to measure the muons' momenta in the target region so a match to the values obtained by the Muon Spectrometer is possible. Unlike the ACM which operates with a current

that is pulsed to the SPS cycle, the PT7 magnet is operated in DC mode, up to a maximum current of 950 A. The value used during the data taking runs was 900 A. An iron ring of 2 mm thickness known as *shim* is mounted on the edges of both of the poles, to ensure the homogeneity of the magnetic field.

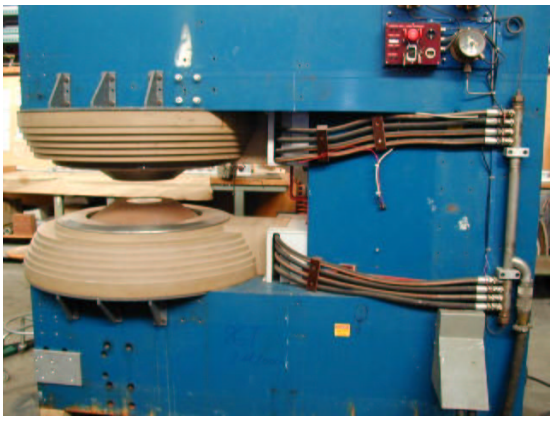


Figure 2.13: The PT7 magnet.

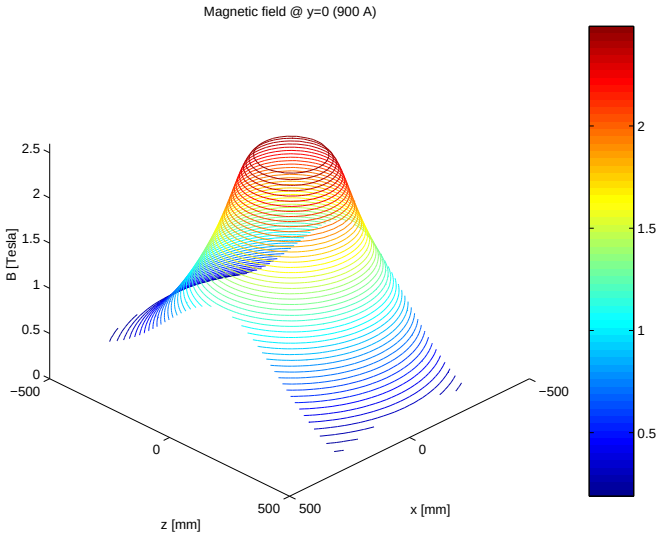


Figure 2.14: The PT7 magnetic field map with a current of 900 A.

The magnetic field was scanned by a Hall probe in the 3 axes, in steps of 1 cm. These measurements achieved a precision of $\Delta B/B \sim 0.01\%$ [18], and were parametrised and inserted into the offline analysis software (see Fig. 2.14). Two different methods were used to describe the field: a simple bilinear interpolation of the measured values (the default in the software),

and a fitting function. Due to the dipole symmetry only two components need be considered: the vertical component B_y and the radial component B_r in the x-z plane.

As with the ACM magnet, the polarity of the PT7 is reversed every few runs to reduce systematic uncertainties. During the field mapping an upper bound of 0.2 % was determined for any possible change in the magnetic field due to changing the polarity.

2.6 The Beam Tracker

The Beam Tracker, developed as a common project between the NA60 and RD39 collaborations, is used to determine the transverse coordinates of the incident beam particle that collided with the target by determining its flight path. It can determine these coordinates with a resolution of approximately 20 μm. Combining the Beam Tracker measurement with the tracks from the Vertex Telescope makes it possible to measure the transverse offset of the muon tracks with a resolution of $\sim$ 40 μm. Given its time resolution of 1.7 ns, it can also identify beam pile-up events.

The Beam Tracker is a silicon microstrip detector operated at 130 K, taking advantage of the effect known as the "Lazarus effect". This effect is the recovery of charge collection efficiency and position resolution of heavily irradiated silicon detectors when cooled down to cryogenic temperatures. In practice, this means that detectors that are no longer functional at room temperature due to radiation damage can recover their performance at these lower temperatures. Two other effects of cryogenic temperatures on such detectors are that signals are faster due to increased charge carrier mobility and that the leakage current becomes negligible. This simplifies the process of designing the front-end electronics in terms of noise requirements.

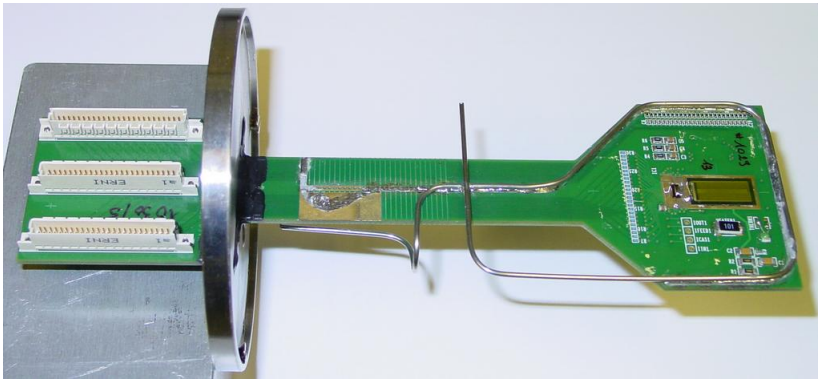


Figure 2.15: A Beam Tracker station.

The Beam Tracker consists of two tracking stations (see Fig. 2.15) in

a chamber with two 0.1 mm thick stainless steel windows on the beam axis. The chamber is kept in vacuum ($\sim 10^{-7}$ mbar) to provide thermal insulation and to minimise the amount of material the beam particles have to go through. The two stations are 20 cm apart and have a relative rotation of 45° to resolve ambiguities. Each of these tracking stations consists of a PCB with two 400 μm thick silicon microstrip detectors mounted back to back, with the strips perpendicular to each other to give the two transverse coordinates. The detectors have 24 1.2 cm long strips of 50 μm pitch plus four 500 μm pitch strips on each side.

The strip signals, after being discriminated, are fed into Multi-Hit Time Recorder (MHTR) FERA modules that sample the signals at an effective frequency of 600 MHz, giving the 1.7 ns time resolution. The discriminated signals are also counted by scalers, allowing for online beam profile monitoring.

Liquid nitrogen flows from a transport dewar vessel and feeds capillaries soldered onto the PCBs. The temperature of the detectors is controlled remotely by setting the nitrogen flow and through power dissipated by a heater on the PCB.

2.7 The Interaction Counter

The Interaction Counter is a detector placed at roughly 40 cm downstream from the target (between the vertex telescope and the hadron absorber) designed to detect the secondary hadrons and photons produced in the target region. The beam goes through its central 11 mm diameter aperture.

Its signal is proportional to the particle multiplicity, and its time resolution of 1.7 ns allows to determine whether or not the interaction it detected produced the trigger. It is useful for eliminating the triggers due to interactions in the dump. The Interaction Counter covers 2.25 units of pseudorapidity, in the range $2.75 < \eta < 5$.

It is composed of two 1 cm thick, 16 cm $\times$ 10 cm scintillators, and a 5 mm thick lead sheet of the same dimensions upstream of the scintillators. This lead sheet converts 30 to 40 % of the photons, adding to the charged particle signal. Requiring a coincidence between the two counters minimises the noise due to induced radioactivity in the scintillators.

2.8 The ZDC and Quartz Blade

The Zero Degree Calorimeter, or ZDC, is used in ion runs to determine the centrality of the collision by measuring the forward energy deposited by the beam spectator nucleons. It is also useful for providing "minimum bias" triggers.

This detector was inherited from the NA50 experiment and consists of 30 1.5 mm thick grooved tantalum slabs stacked to form a 5 × 5 × 65 cm^3 parallelepiped. Tantalum was chosen as the passive material due to its mechanical properties and its small interaction length. The active part of the detector is composed of 900 uniformly distributed quartz fibres, with a 1.5 mm pitch. Only the first 65 cm of the 180 cm long fibres, inside the tantalum, represent the active part of the detector (see Fig. 2.16). The rest of the fibres (bent at a 90° angle to the beam line) act as a light guide to photomultipliers 1 m away from the beam axis (to prevent background due to shower particles).

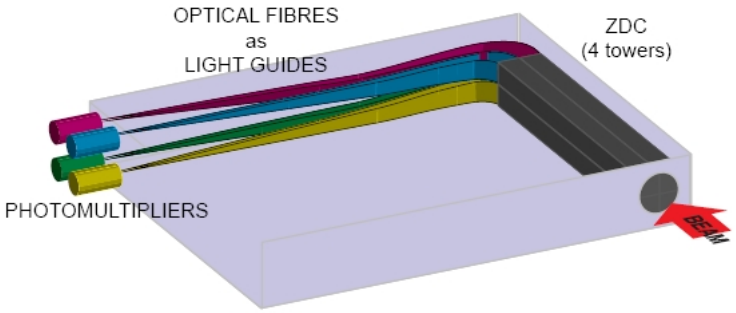


Figure 2.16: Mechanical structure of the ZDC.

The ZDC is divided into 4 towers, each comprising 225 fibres and read out by a separate photomultiplier. It is worth noting that there is always a residual energy measured by the ZDC even in the most central collisions.

The Quartz Blade is a Cherenkov counter that complements the ZDC. Its signal is proportional to the sum of the square of the charges of the particles traversing it (see Fig. 2.17). This makes it possible to identify the cases in which nuclear fragments resulting from a very peripheral interaction deposit the same amount of energy as a non-interacting beam ion.

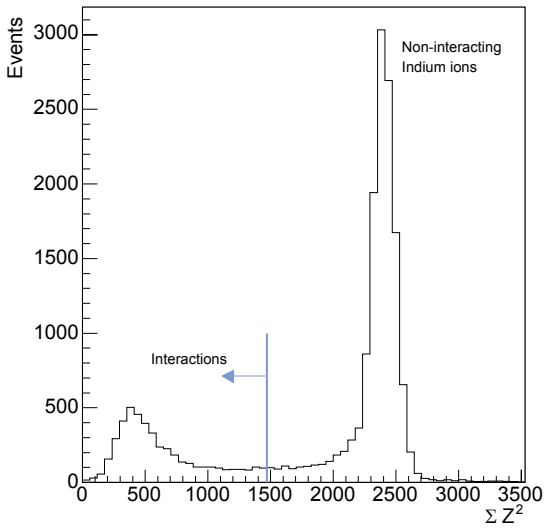


Figure 2.17: Quartz Blade distribution of charge squared.

Chapter 3

The NA60 DAQ and Monitoring Framework

3.1 The Data Acquisition System

3.1.1 Overview

The NA60 data acquisition system uses DATE [19], a package developed for and by the ALICE collaboration as a basis for prototyping the DAQ system components and as support for the ALICE test beams. The DATE (Data Acquisition and Test Environment) system is a set of different programs that implements many of the different requirements of a data acquisition system, such as readout, monitoring, error reporting and run control. All of these different components are designed to work in a distributed multi-processor environment.

The DATE system requires that the operating systems of all the hosts be Unix-like (including GNU-Linux); all the hosts must share the same file system; and they must be connected to a network supporting TCP/IP. The processors of the hosts, however, may be of any type. DATE also supports mixing machines of different architectures in the same setup (be them of different brand, operating system or endianness).

Besides controlling the data flow, the DATE system also provides the following features:

- Parametrisation of the hardware configuration
- Interactive control of the data acquisition parameters
- Run control
- Display of the run status
- Event monitoring

- Information reporting
- Run bookkeeping

The main computer components of a system based on DATE can be divided into 2 categories: the LDC hosts (Local Data Concentrator), and the GDC hosts (Global Data Collector).

An LDC is a host that is responsible for reading the front-end electronics of a specific detector or subsection thereof. The different LDCs manage all of the data streams concurrently — the triggering system provides the necessary synchronisation so that all of these streams refer to the same physics event. These streams are sent to the data acquisition network, being dispatched to all of the available GDCs (although not at the same time). This is done in order to balance the load and minimise the latency. The algorithm responsible for dispatching the LDC data streams not only manages the load balancing, but it also makes sure that all of the sub-events from each LDC (pertaining to the same occurrence of a trigger) make it to only one GDC.

The GDCs, on the other hand, are hosts dedicated to building a full event by collecting all of the different sub-events from the LDCs. A full event is simply the concatenation of the sub-events, preceded by a full event header, which, amongst other things, registers the detectors present in the event.

The GDC also performs the recording function after building the full events, which might mean writing to local disks or sending the full events over the network to a central server. In the NA60 case, the events are all written to local disks. A *daemon* then periodically (but asynchronously to the rest of the data acquisition) moves these files to CASTOR, CERN's central data storage facility, to be permanently stored on tapes.

It is possible to have a simple setup of just one LDC, with no GDCs. In this case, the lone LDC does not send its sub-event to another machine, and deals with recording it itself. This sort of setup is generally used to test the DAQ for a new detector before integrating it into the general, more complicated system.

3.1.2 Readout and Data Flow

There are many processes in the DATE system, each of them performing different tasks. One of them, `readout`, is responsible for reading the front-end electronics. It does so by waiting for a trigger, then reading the data and filling a circular buffer called the recording buffer. It may also fill a monitoring buffer with the same data if there is a monitoring program requesting events and monitoring is enabled in the Run Control panel.

A separate process, `recorder`, offloads the recording buffer and sends the sub-events to the proper device, which is either the disk (in the simple case

where there is only one LDC and no GDCs), or to the IP address of one of the GDCs. When there is more than one GDC, the `recorder` process cycles through all of them. To make sure that all the LDCs are synchronised, the dispatching algorithm checks the event identifier of each of the sub-events.

Every time the `recorder` process opens a socket, an internet daemon called `gdcServer` is created. This daemon gets the events from the socket and fills a circular buffer. There is one `gdcServer` and one circular buffer per LDC in the GDC. These buffers are offloaded by another process called `eventBuilder`. The `eventBuilder` process collects all of the sub-events from the buffers, builds the full event and then sends it to disk (or wherever else is specified in the configuration file).

3.1.3 Run Control

The whole system is controlled from one central point. This might be one of the machines involved in the data acquisition or, to decrease the load on the LDCs and GDCs, an independent dedicated one. The process responsible for controlling the DAQ is called `runControl`. The actual hardware configuration of the setup is stored in a text file, read by `runControl` at startup.

This process opens sockets to all of the machines declared in the configuration file, where internet daemons are created. The `rcServer` daemon controls the `readout` and `recorder` processes, while two other daemons (`ebDaemon` and `rcServer`) control `eventBuilder`. Communication with the controlled processes is achieved through the use of shared memory segments.

The most visible aspect of `runControl` is the main control window (see Fig. 3.1), displaying the status of all of the machines in the setup, as well as providing buttons and menu options to control the acquisition (such as starting and stopping it).

3.1.4 Information Logging

All of the DATE processes generate messages that are sent to a socket. As in the case of the run control, a certain machine is designated to receive and store these messages to disk, using the `infoDaemon` internet daemon. DATE also provides the `infoBrowser` application to view these messages (see Fig. 3.2) according to some selection criteria (such as where they came from, sorted by time, etc.).

3.1.5 Event Monitoring

A separate application (such as the ones discussed in the next section) may request copies of the events from the DAQ system by calling the monitoring library routines included in DATE. This monitoring library was written to offer a uniform interface to support user-written monitoring programs.

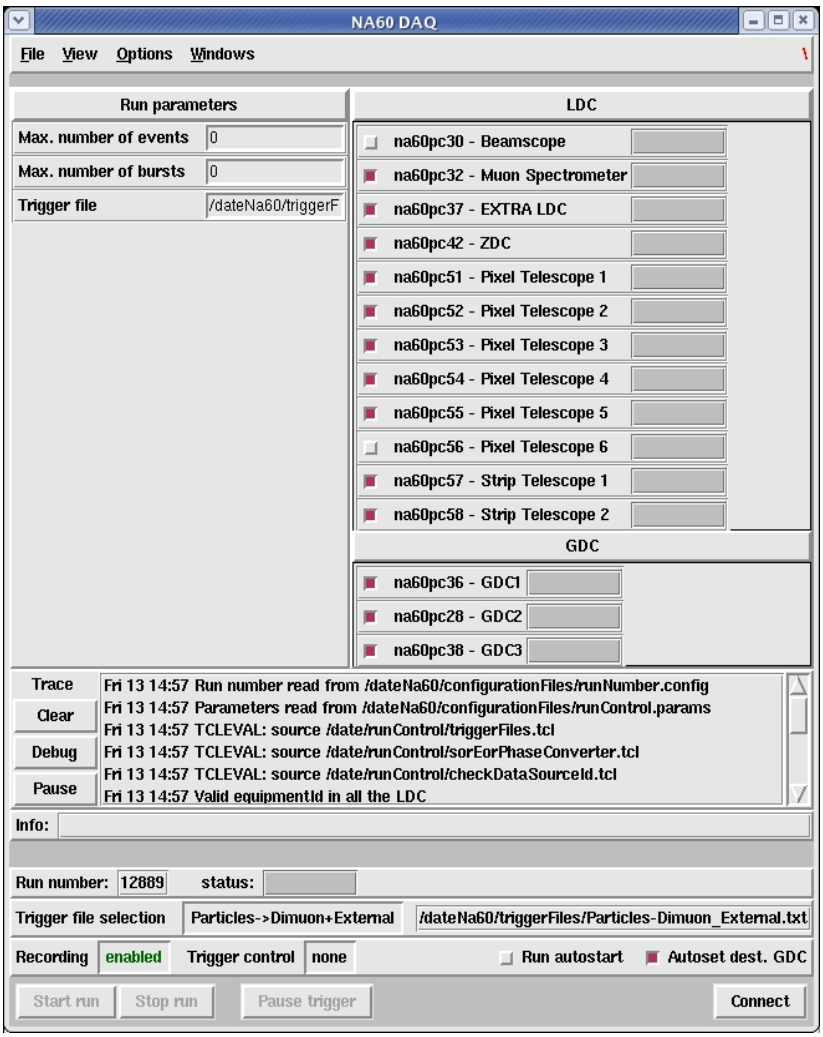


Figure 3.1: The Run Control panel.

This monitoring interface implements access to events coming from the online experimental stream or from a permanent data storage medium (such as a hard disk).

When a monitoring program requests the events via the monitoring library, a special buffer reserved for monitoring purposes is filled with the requested events (since the client application can request a certain selection of them). This buffer is either filled by `readout`, if the machine receiving the request is an LDC, or by `eventBuilder`, if it is a GDC. No copy of the events is made in the absence of a request.

The application requesting the events and performing the analysis may run locally on the machine it gets the events from. In this case it gets the events directly from the monitoring buffer. To avoid overloading the online

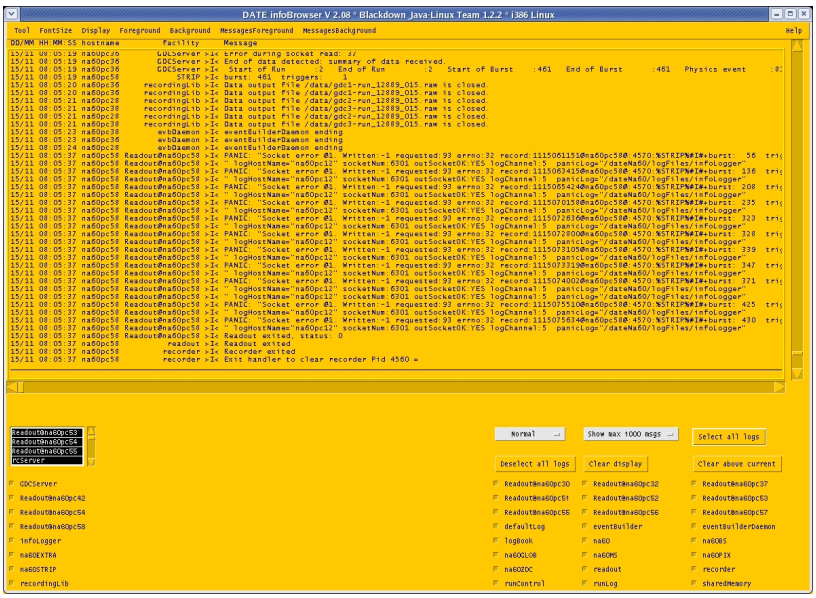


Figure 3.2: The `infoBrowser` application.

host producing the data stream (but at the cost of an increased load on the network), it is also possible to run the monitoring application on a separate machine, as is always the case in NA60. The events are then sent through the network to the client application via a process called `mpDaemon`. There may be several local and remote monitoring programs, running simultaneously and retrieving the data from the same source. Each of them can only receive the data from one source at a time, however. The DATE monitoring library routines can also access events stored on disk (local or remote) in the same manner as the online events by calling the same functions.

When a monitoring program connects itself with the data stream it is to analyse, it can declare a *monitoring policy*. This policy specifies what types of events are to be read, and it can exclude events of a certain type, require all events or only a statistical sample of them. The latter option is included to prevent overloading the DAQ system. Programs requesting all events of a given type have to process the information as fast as it is sent to them or they stall the acquisition. On the other hand, excluding certain events or only analysing a sample of them reduces the load on the online host and interconnecting network.

Monitoring programs also have the option of stalling if no data is available or continue execution normally, knowing, however, that no data has been received. Batch programs that do not require user interaction might opt to stall, but event-driven programs with graphical interfaces must continue their execution. If such programs stalled, no user intervention would be possible until another event arrived.

3.1.6 The DATE Data Format

Data Files

There are usually many files recorded in just one run. In DATE, a run is a distinct period of data-taking ranging from when a user commands the run to start until it stops, either because it was so chosen by the user or due to an error. In NA60, runs were periodically stopped to change the signs of the currents in both the ACM and PT7 magnets, for instance.

Due to the fact that it is possible to limit the maximum file size, the process responsible for recording the data (`recorder` in the LDC or `eventBuilder` in the GDC) opens a new file when the maximum size is reached. All files from the same run share the same run number, and there are always at least two files in a run. The first file, with index 0, contains only the start-of-run information. The subsequent files, starting with index 1, contain the experimental data. These indices make up part of the name of the files (e.g. `run3456_000`, `run3456_001`, etc.).

It is also possible to limit the total amount of kilobytes to be recorded in one run, and if that limit is reached, the run stops automatically.

The Data Format

The data format for a DATE event is defined by the following C structure:

```
struct eventStruct {
 struct eventHeaderStruct eventHeader;
 unsigned short rawData[1];
};
```

The `rawData` field acts as a pointer to the event data that was read out, whereas the event header information is contained in the `eventHeaderStruct` structure, shown below (where `long` means a 32-bit integer):

```
struct eventHeaderStruct {
 long size;
 unsigned long magic;
 unsigned long type;
 unsigned long headLen;
 unsigned long runNb;
 unsigned long burstNb;
 unsigned long nbInRun;
 unsigned long nbInBurst;
 unsigned long triggerNb;
 unsigned long fileSeqNb;
 unsigned long detectorId[MASK_LENGTH];
 unsigned long time;
 unsigned long usec;
 unsigned long errorCode;
 unsigned long deadTime;
 unsigned long deadTimeusec;
 unsigned long typeAttribute[ATTRIBUTE_WORDS];
};
```

The meaning of the fields of the event header is explained in Table 3.1.

The Event Types

The `type` variable in the event header identifies each event and usually determines the type of processing (see Table 3.2). It is used by `eventBuilder` to decide whether or not sub-events should be assembled into a full event. Only sub-events of the type `PHYSICS_EVENT` get packed together into a full event. Sub-events of all the other types only have a header added to them.

Type Attributes

The `typeAttribute` field, along with some C macros included to manipulate this field, allows users to label certain events with user-defined attributes. Each of these attributes must be boolean, since they are stored in a single bit of `typeAttribute`, allowing for 64 different user-defined attributes. The main purpose of this feature is in the selection of events, based on these attributes and/or the event type in the offline analysis or online monitoring.

The Full Event Format

The data format described applies to LDC sub-events, LDC full events (when there are no GDCs), and GDC full events. The main difference between GDC full events and both types of LDC events is that there is a GDC

Field	Meaning
`size` (32 bits)	Size of the event in bytes
`magic` (32 bits)	Arbitrary number used for consistency check set by `readout`
`type` (32 bits)	The type of the event (explained below)
`headLen` (32 bits)	Size of the header in bytes
`runNb` (32 bits)	Run number
`burstNb` (32 bits)	Burst number; set to 0 by `readout` and updated through the user routine `ReadEvent`
`nbInRun` (32 bits)	Event number in the run; also set by `ReadEvent`
`nbInBurst` (32 bits)	Event number in the burst set by `ReadEvent`
`triggerNb` (32 bits)	Trigger number for this detector incremented by `readout` only for physics events
`fileSeqNb` (32 bits)	Sequence number of the data file when there is a maximum file size
`detectorId` (96 bits)	Detector identification mask; each bit identifies one LDC. Valid detector bits range from 0 to 94
`time` (32 bits)	Time in number of seconds since 0:00 GMT 01/01/1970
`usec` (32 bits)	Time in microseconds to be added to the previous field
`errorCode` (32 bits)	Experiment dependent error code
`deadTime` (32 bits)	Set to 0 by `readout`. May be used to measure the dead time in seconds of the readout
`deadTimeusec` (32 bits)	Analogous to `usec`
`typeAttribute` (64 bits)	User defined attribute(s) associated to the event

Table 3.1: Event header fields.

header. After this GDC header, the full event consists of the concatenation of all the sub-events, including the sub-event headers, as seen in Table 3.3. The only difference between an LDC sub-event and an LDC full event lies in the `detectorId` field, as shown in Table 3.4.

3.1.7 Readout Programs

The `readList`

The `readout` process contains all the detector-specific code necessary to read out the front-end electronics, which is specified in a separate module called the `readList`. The `readList` has to be compiled and linked with the main readout program, and it consists of the following C functions:

- `ArmHw` — called at the start of the run to perform any necessary initialisations

START_OF_RUN
END_OF_RUN
START_OF_RUN_FILES
END_OF_RUN_FILES
START_OF_BURST
END_OF_BURST
PHYSICS_EVENT
CALIBRATION_EVENT
END_OF_LINK
EVENT_FORMAT_ERROR

Table 3.2: List of event types.

Full Event Header
Sub-event A header
Sub-event A data ... ...
Sub-event B header
Sub-event B data ... ...
Sub-event C header
Sub-event C data ... ...

Table 3.3: GDC Full Event.

	sub-event	**LDC event**	**GDC event**
Detector Id bit 95	0	0	1
Detector Id bits 0 to 93	Detector Id of the corresponding detector	0	Logical OR of the detector Ids in the event

Table 3.4: Detector Ids in the different event types.

- `EventArrived` — called in the main event loop to determine whether or not there was a trigger
- `ReadEvent` — the routine responsible for extracting the information from the hardware after the occurance of a trigger
- `DisarmHw` — called at the end of the run to "shutdown" the hardware, if necessary

Both `ArmHw` and `DisarmHw` return `void` and take an arbitrary number of arguments. They must still be defined even when they are not needed. `EventArrived` has to return a positive integer if there was a trigger, and 0 if not. It is continuously called by the `readout` process in a loop that only stops when there was a trigger.

The most complicated of these functions is `ReadEvent`. It returns the number of bytes read, and has 2 pointer parameters passed to it: one pointing to the event header, and another that it must fill with the data from the electronics. It must also set the following fields in the event header: `nbInRun`, `nbInBurst`, `burstNb`, and `type`.

All of these functions signal error conditions through two global variables: `readList_error` and `readList_errorSource`.

The Generic `readList`

DATE has a generic `readList` that may be used instead of one specific to an experiment. In this scenario, the detectors in the experiment are partitioned into subparts called *equipments*. Each of these equipments has its own readout code, with its own version of the four `readList` functions. The advantage of this approach is that it makes the readout software more modular, and hence, more readable. There may be several copies of one equipment in the experimental setup, and all these copies share the same readout code. Although the functions will be the same, they will be called with different parameters for each one of these copies.

The definition of the equipments in the setup and the detectors they belong to is contained in a plain text file called `detectors.config`. This file also specifies the parameters that distinguish copies of the same equipment, as well as the parameters pertaining to a whole detector.

There are two main types of equipments: trigger equipments and readout equipments. The difference is that readout equipments do not have to define their own version of `EventArrived`, which is the responbility of trigger equipments. Although there may be many trigger equipments declared in `detectors.config`, only one of them may be active per detector.

The Equipment Header

When the generic `readList` is used, each LDC subevent is composed of several equipment data blocks (see Table 3.5), just as each GDC full event is composed of LDC sub-events. As in the full event, each one of these blocks will be preceded by a header.

Sub-event Header
Equipment 1 header
Equipment 1 data ... ...
Equipment 2 header
Equipment 2 data ... ...

Table 3.5: Sub-event structure with the generic `readList`.

The equipment header is defined by the following C structure:

```
struct equipmentHeaderStruct {
 short headerExtLen;
 short type;
 char reserved;
 char rawByteAlign;
 short equipmentId;
 long rawDataLen;
};
```

The meaning of the fields is explained in the Table 3.6.

Field	**Meaning**
`headerExtLen` (32 bits)	the length in bytes of an optional header extension (defaults to 0)
`type` (16 bits)	the type of the equipment as declared in the configuration file
`reserved` (8 bits)	reserved byte
`rawByteAlign` (8 bits)	length of the words read from the hardware (in bytes)
`equipmentId` (16 bits)	equipment identifier; used to distinguish copies of the same equipment type
`rawDataLen` (32 bits)	length of the data block in bytes (header excluded)

Table 3.6: Equipment header fields.

3.2 The Monitoring Framework

3.2.1 Motivation

A monitoring program's purpose is to analyse the signal coming out of a certain detector as the experiment is running. They can be used to detect errors, ensure that the performance of the detectors is satisfactory, and even to infer information about the beam. There are several monitoring programs, one or more per detector. This is because sometimes it might be desirable to analyse the same data in a different way, e.g. by displaying it in a different manner. An example of this would be the Beam Tracker, for which different monitoring programs were written to separately display timing and spatial information.

The framework's conception was motivated by the many similar requirements that monitoring programs have. Instead of repeating the same code over and over again, OOP (Object Oriented Programming) was used in the C++ programming language to group together the common functionality in a few classes. These classes then act as reusable modules, implementing the common behaviour in all of the programs, which also makes updating them all at once a simple and easy task. The ROOT [20] software framework is used for manipulating data and displaying graphics, as well as in the implementation of the GUI (Graphical User Interface). By using ROOT the monitoring framework benefits from a wealth of funcionality that does not need to be implemented, such as creating and fitting histograms or dealing with mouse input.

As NA60 uses DATE for its data acquisition, the online streams analysed by the monitoring programs and the files containing the physics data have the same format, making it easy for the monitoring programs to read either one of them in the same way. Due to the same data format, some of the functionality implemented by the classes of the monitoring framework is also required when doing offline analysis.

Since some of the common problems are solved by the monitoring framework, the revelant classes were included in the offline analysis software of the experiment, NA60ROOT. In this way the framework plays a dual role in NA60: not only does it implement common behaviour and functionality for the monitoring programs, it also shares some of its data reading capabilities between the monitoring programs and the offline software.

3.2.2 The Monitoring Modules

A monitoring program uses objects from several classes, the main ones being `NaMonitor`, `NaDateDecoder`, `NaSource`, and `NaDateEvent`. `NaMonitor` is responsible for the user interface and deals with the common behaviour expected from all monitoring programs. `NaDateDecoder` is a base class for reading DATE streams and decoding the raw data into something more

understandable to human beings (such as columns and rows in the pixel telescope) called *digits*. `NaSource` is an abstract class that defines an interface for the different types of data sources that can be read from. Finally, `NaDateEvent` encapsulates the event data from the DATE event structure.

A general monitoring program defines a class derived from `NaMonitor` to graphically display the data it receives; a class derived from `NaDateDecoder` to retrieve the data from a DATE stream and decode it into digits; and also a digit class to contain the decoded information passed from the decoder to the monitoring class. All of these are explained in more detail below.

3.2.3 The Monitoring Control Panel

`NaMonitor` deals with all the functions that are not specific to any program in particular. Some of these functions include: allowing the user to look at the next event; keeping track of whether or not the results from different events should be integrated or not; and reseting and updating the display. These operations are handled generically, without any knowledge of what the program outputs to the screen or the kind of the data it receives. All of the application-specific functionality is contained in virtual methods, delegating the responsibility to child classes.

The actual graphical interface is handled by `NaMonitorGui`. This class is responsible for creating all the graphical *widgets* (such as buttons and text boxes) that compose the interface, as well as linking all of them to the `NaMonitor` methods that execute the actions they represent. Keeping track of when the display should be updated and reset is the responsibility of another auxiliary class, `NaMonitorRefresh`.

Every time `NaMonitor` handles a new event, it first extracts the event header information from the digits array of that event, then calls its virtual method `DigitsHandler` to update the application specific data structures. The actual graphics are handled by the virtual methods `UpdateDisplay` and `ResetDisplay`. These methods are defined in the `NaMonitor` child classes, implementing the desired behaviour of the programs that use the child class in question.

Two classes derive from `NaMonitor`: `NaCanvasMon` and `NaTabMon`. The former is used for monitoring programs that only need one screen of output, whereas the latter is used for multiple screens that can be selected, as the name implies, by tabs. All of the NA60 monitoring programs implemented using this framework derive from one of these two classes. Figure 3.3 shows, as an example, the monitoring control panel for the pixel telescope.

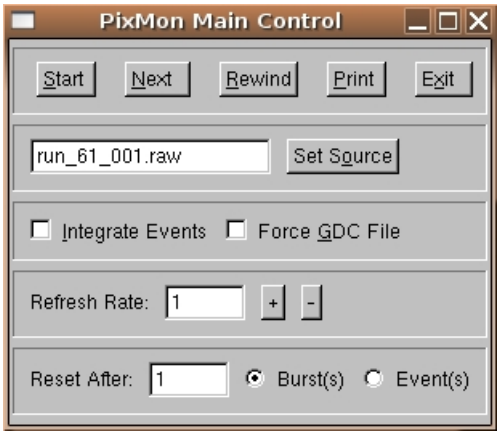


Figure 3.3: Monitoring control panel for the pixel telescope.

3.2.4 Reading the Data

Overcoming the DATE Library's Limitations

Although DATE has its own C library to read the data files and online streams, it has 2 shortcomings: first, only one stream can be opened at the same time (online or offline); secondly, the library gets the data from one detector in one huge block, leaving it to the user to navigate it as he or she sees fit. There is a DATE monitoring package based on ROOT and written in C++ that can open more than one stream at once, but in its current version only works with ROOT 2.23 or older.

Whereas reading from only one stream is not a big problem from an online monitoring perspective, in the offline analysis it is necessary to open more than one stream for several reasons. One of them is the fact that the experiment uses 3 GDCs. This means the data are spread across files from these 3 computers in a non-sequential manner, which means that all 3 files need to be opened at the same time.

In other cases in the offline analysis it is necessary to choose whether or not to analyse an event depending on data stored in special *end of burst* events, which, by definition, come after the physics events being considered. Both events must then be read at the same time, which means having to open the same file twice.

Due to all of the aforementioned problems, a class was written with the express purpose of reading from several DATE files at once. The DATE library, however, is still used for online monitoring.

Fetching the Revelant Equipment Data

As seen above, a GDC event contains data from all of the detectors in the experiment, by collecting data from all of the LDCs. Each of these LDCs

in turn has its data partitioned into equipments.

At each of these hierarchical levels there are headers embedded in the data, which contain some information about the subsection they refer to, as well as marking where it begins. Equipments read out by a certain PCI card conceived by the NA60 collaboration have an extra header and also a *trailer*, marking the end of that equipment's block of data.

Trying to reach the revelant information in this environment is difficult and prone to error if done manually. To automate this process, code was written to find the start of the equipment data one wants to analyse. To do that, however, it is necessary to specify which equipment in the setup is to be analysed. Each of the equipments in the experimental setup can be uniquely identified by 3 numbers: the detector number, the equipment type and the equipment number (the equipment number is needed to distinguish multiple copies of the same equipment type).

These 3 numbers are encapsulated in the `NaEqParams` class, which provides methods for getting and setting its 3 member variables. Data from several equipments is accessed by specifying an array of `NaEqParams` objects.

The class that actually navigates the data according to the `NaEqParams` object is `NaDateEvent`. This class represents a DATE event, keeping track of its length and whether it is a GDC event or an LDC sub-event, as well as allowing client code to access the event data and event header. By calling its `GetEqData` method and passing to it an `NaEqParams` object, client code can access the data relative to a certain equipment via an `NaEqData` object.

The `NaEqData` class contains, and allows access to, the length of that equipment's data, the equipment index (its order in the array of equipments), the event header (which is common to all equipments in the same LDC), and a pointer to the data.

The Data Source Classes

As mentioned above, offline applications need to open more than one stream at once. Online monitoring programs, however, read from only one source. In both cases, the behaviour that the client code expects is to simply obtain the next event. When reading events from an array of sources, attention must be paid to the distribution of the events among the data files. To make all of this bookkeeping transparent, the classes responsible for reading the data all derive from the abstract class `NaSource`. This class defines a common interface that the child classes must implement, the most important method of which is `CreateNextEvent`, responsible for allocating memory for and fetching the next event from the source or sources in question.

One of these classes is `NaDataSource`, meant to read from only one source of data, be it an online stream or a data file. One of its child classes is `NaDateMonSource`, which is merely a *wrapper* class. All its methods do is call the functions from the DATE C library. The other class that derives

from `NaDataSource` is `NaFileSource`, designed to read only data files. It has child classes itself, `NaDateFileSource` and `NaNonDateFileSource`. The reason for the existence of `NaNonDateFileSource` is that, at one time, the Microstrip Telescope's data acquisition had not yet been integrated with the rest of the NA60 DAQ. This meant that the data files generated were not in the DATE format. This class was then written to read these different files and return `NaDateEvent` objects just as the other source classes do, making it all transparent to the client code.

`NaFileSource`, however, was written to read DATE files in the same way that the library functions do, but circumventing the limitation of only being able to open one file at a time through the possibility of creating several `NaFileSource` objects. Since there is a maximum file size for the data files in most runs, the data from the whole run is distributed into several smaller files. When `NaFileSource` reaches the end of a file, it verifies if there is another of these files in the same directory. If such a file exists, it closes the original, opens the new file and keeps on fetching events from the same run in a manner that is completely transparent to the client code.

The `NaDataSource` static method `CreateDataSource` creates objects of `NaDateFileSource`, `NaNonDateFileSource` and `NaDateMonSource` according to the name of the source passed to it. Since it returns a pointer to a `NaDateSource` object, it makes handling the sources even more transparent since it doesn't even need to know the class of the object passed back by `CreateDataSource`. Instead, it accesses the events through the common interface that `NaSource` declares.

Depending on the name of the source, `CreateDataSource` creates an object of one of the three classes above (which means an online stream, a DATE file or a data file of another type). It is also necessary to specify a monitoring policy. This is an array of text strings that designates the type of events to be read from the data source (e.g. physics events, end-of-burst events). More than one type of event can be selected, hence the need for an array of strings as opposed to just one.

The other class that derives from `NaSource` is `NaSourceArray`. This class is the one responsible for reading from more than one GDC data file. It does not, however, read the files directly, using instead one data-reading member variable per source to do that. Since it is necessary to look at the next event in each of the sources in the array to determine which one is the oldest (and thus the next event overall), these member variables are objects of another class: `NaBufferedSource`.

This last class has an `NaDataSource` member variable it uses to access the data. It reads the next event from this data source variable and stores it in a buffer. When `CreateNextEvent` is called, it returns the event in the buffer and replaces it with another one from the data source.

The `CreateNextEvent` method in `NaSourceArray` is then very simple. It loops over all its buffered sources to check which one of them has the

oldest event in their buffer and returns it (by calling that source's own `CreateNextEvent`).

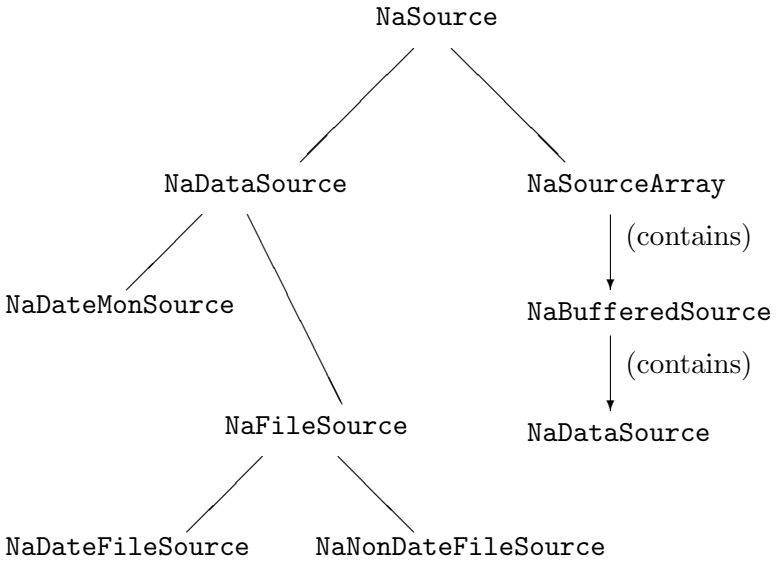


Figure 3.4: The data source class hierarchy

The Decoder Class

The decoding of the raw data is done with the `NaDateDecoder` class. Its virtual method `DecodeDigits` is used to do the application specific decoding, which is called automatically every time it has to handle the data from an equipment. This function, implemented in child classes, must not only decode the information but also correctly fill its array of digits, a member variable of the class `NaDigitsArray`. This member variable contains the event header of the equipment data as well as the array of decoded digits.

`NaDateDecoder` derives from `NaDateReader`, the class that actually reads the DATE streams. This base class does mainly three things: it manages the source(s) one wants to read from; it stores the array of equipment parameters to identify the equipments to be analysed; and it manages the access to the events in the stream.

There are three main parameters to be passed to the `NaDateReader` constructor: the source, the monitoring policy, and the number of equipments to be read. If the source is an integer, or a string representing an integer, it is considered to be a run number. In this case, it automatically opens the 3 GDC files corresponding to that run in a `NaSourceArray` object. If the source does not represent a number, it creates a single source via the

`NaDataSource` static method `CreateDataSource`, which could mean reading either a single data file or an online host.

The number of equipments to be read is passed to the `NaDateReader` constructor so that it can allocate the necessary memory for the array. The values for the detector number, equipment type and equipment number for each equipment must be set in the child class, for that depends on the application. That is why the `fEqParams` member variable is *protected*.

`NaDateReader` also acts as an event manager. It has a method called `HandleNextEvent` that creates an event, processes it, and subsequently deletes it. It is also possible to loop over all the events by calling `Run`, which calls `HandleNextEvent` until it is no longer possible.

An event is processed by looping over all of the `NaEqParams` objects in the array and calling the virtual method `RawDataHandler`. This method takes as its argument the `NaEqData` object returned by the event for that equipment, and should be defined in child classes to process the data according to the application's desired behaviour.

In the case of `NaDateDecoder`, `RawDataHandler` adds the event header information to the digits array and then calls `DecodeDigits`. Simpler applications that do not need the monitoring control panel might derive directly from `NaDateReader` and use `RawDataHandler` to print out information from the event. In these simpler cases, there is no need to pass decoded information to another object, and hence no need to define a class that derives from `NaMonitor`.

3.2.5 ROOT

ROOT [20] is a software framework, written at CERN and in the context of the NA49 experiment, designed to tackle the problems faced in data analysis in High Energy Physics. It is an object oriented framework, written in C++, and it implements the functionality needed to handle and analyse large volumes of data. Some of its features are:

- A C++ Command Line Interpreter
- A script processor
- Histograms and fitting
- Graphical User Interface widgets
- 2D and 3D graphics
- File input/output of C++ classes
- Collection classes

In the monitoring framework, it is used for the GUI and graphics, most of which consist of displaying 1D and 2D histograms. One of its container classes, `TClonesArray`, is also used in the communication between decoder and monitoring objects, via its child class `NaDigitsArray`. It was necessary to derive from `TClonesArray` to use its functionality and to add a member variable for the event header.

3.2.6 Program Examples

This subsection presents screenshots from three of the monitoring programs used in the NA60 experiment. All of these programs use the framework described in this chapter.

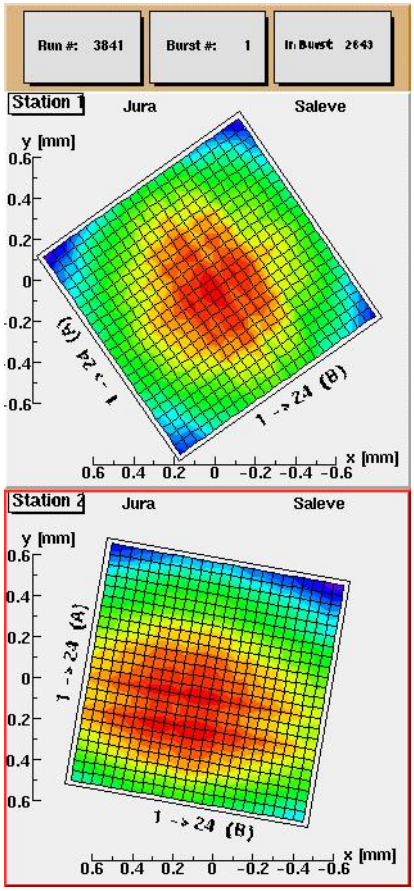


Figure 3.5: Beam Steering Monitoring.

Figure 3.5 is a screenshot of a program that monitors the beam tracker. Both planes of the detector are represented by a colour coded 2D histogram, in which the closer a colour is to red, the higher the value in that histogram bin. The histograms are also rotated to mimic the spatial orientation of the

planes in the actual experimental setup. This application allows the users to see where the beam is hitting the detector (if at all) and to steer it into the centre of the planes.

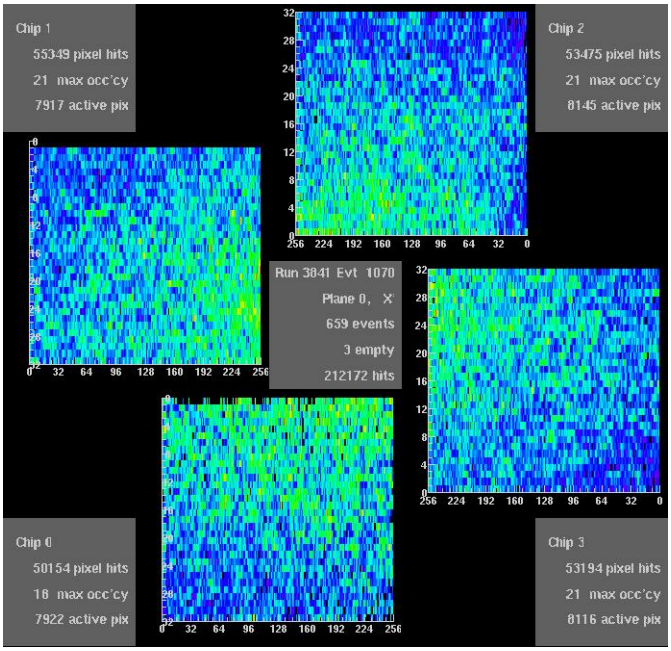


Figure 3.6: Pixel Telescope Monitoring.

Figure 3.6 was taken from the pixel telescope monitoring program. Here, also, each one of the pixel chips is represented by a 2D colour coded histogram. This screenshot was taken after integrating over several events, since individual events have few hits.

Lastly, Figure 3.7 represents an example of a monitoring program that derives from `NaTabMon` to show several screens of data. This program also monitors the beam tracker, but shows the time information from the MHTRs. It is possible to select which screen is to be displayed by clicking on the corresponding tab. The tabs are part of the GUI window and are not shown.

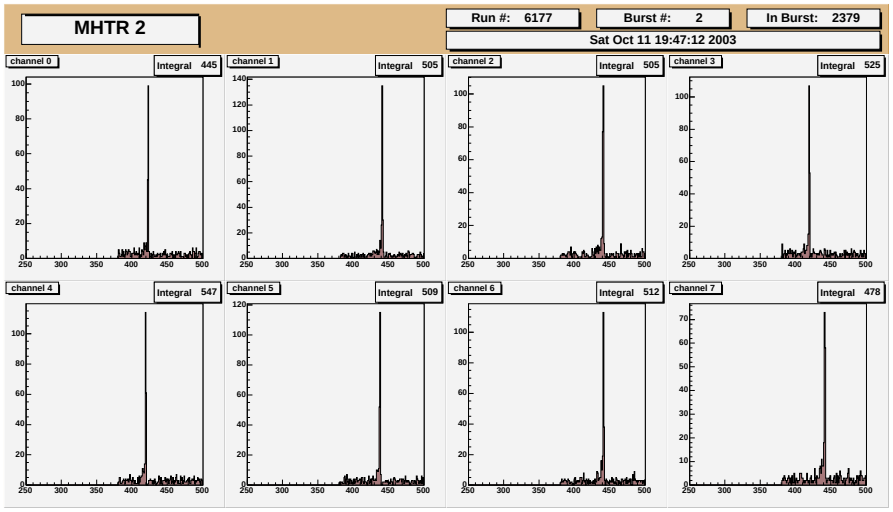


Figure 3.7: MHTR Monitoring.

Chapter 4

Data Reconstruction and Selection

This chapter first describes how the raw data read out by the DAQ and stored in DATE files is converted into physically meaningful information. It focuses on the Vertex Telescope, since it is that detector that is the basis of the current study. The following sections describe the data selection criteria, from whole runs down to individual events and tracks reconstructed in the Vertex Telescope.

4.1 Data Decoding

As explained in Chapter 3, each DATE GDC event contains the information from all of the detectors in the experiment. This large amount of data can be made sense of by the headers that indicate the start of specific detector and sub-detector data sections along with their respective size. The software developed initially for online monitoring purposes makes it very easy to access individual sections of data by giving each one of these its own address. The minutiæ of how to actually access each individual block is implemented by the `NaDateReader` class described in Section 3.2.4, so that the client code only has to specify which blocks to read from a particular event.

Each NA60 detector has its own digital data format in accordance with its needs. All of them compress the data to facilitate its transmission and storage. As a result, each block of data must be interpreted, or decoded. The decoder classes used in the online monitoring programs were integrated into NA60ROOT, the experiment's offline analysis software, to read the experimental data. Other decoder classes were also written to interpret data from detectors not covered by the NA60 Monitoring Framework. All of these bridge the gap between low-level digital data, sometimes compressed to the point of using individual bits to represent information, into meaningful

higher level concepts. In the Pixel Telescope's case, for instance, these would be a list of hits in an event, characterised by the row, column and pixel plane in which they happened.

4.2 Vertex Telescope Reconstruction

4.2.1 Clustering

A single charged particle may generate a signal in more than one sensing element, which would be pixels from the same chip in the Pixel Telescope's case. One of the reasons this may happen is that particles may emerge from the 300 μm thick sensor sufficiently displaced in the x-y plane from its entrance point to deposit charge in one or more neighbouring pixels. This is especially true if the inclination is aligned with the pixel's smaller 50 μm dimension.

Another reason a particle could cause more than one pixel to have a signal above threshold is related to the magnetic field of the PT7 magnet. This causes the charge carriers to be affected by the Lorentz force and be deflected from their expected collection point [21]. A measure of this deflection is the Lorentz angle, given by

$$\tan\theta_{\mathrm{L}} = \frac{\Delta x}{d} = \mu_{\mathrm{H}} B \quad ,$$

where d is the detector thickness, Δx is the shift of the centre of charge (since the electric field $\vec{E}$ is oriented along the z axis), μ_{H} is the Hall mobility and B the magnetic field strength. The value of μ_{H} depends on whether the charge carriers are electrons or holes and increases with decreasing temperature, which in turn means greater displacements. This does not cause a problem for the Pixel Telescope since it is operated at 30°C–40°C. For silicon at room temperature, the value of μ_{H} for holes (the charge carriers in the ALICE1LHCb pixel chip) is 370 cm^2/Vs, corresponding to a Lorentz angle θ_{L} of 5.3° for a magnetic field of 2.5 T [22].

Figure 4.1 shows the average cluster size for both of the PT7 magnet's polarities for the third pixel plane in the 2003 setup. The cluster size minima correspond to when the track angle matches the angle induced by the $\vec{v} \times \vec{B}$ term of the Lorentz force $\vec{F} = q(\vec{E} + \vec{v} \times \vec{B})$ experienced by an electric charge in an electromagnetic field. Clusters are the logical building blocks used in the reconstruction code to calculate particle hit coordinates, which minimises the effects described above.

4.2.2 Track Reconstruction

During track reconstruction, the VT stations are organised into two different kinds of groups: optional and mandatory, where there are multiple

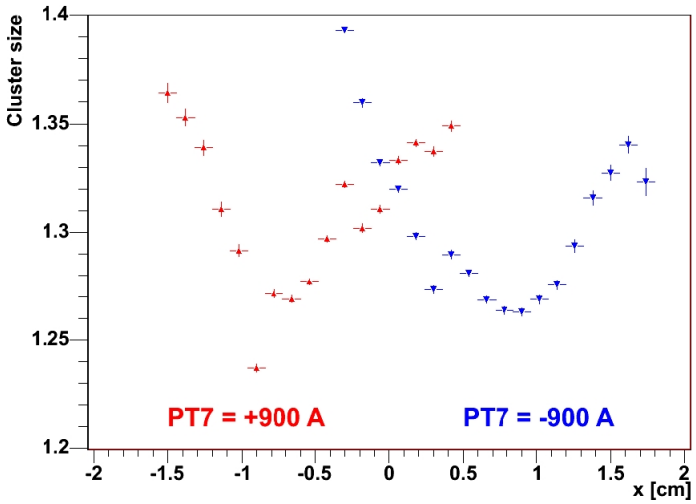


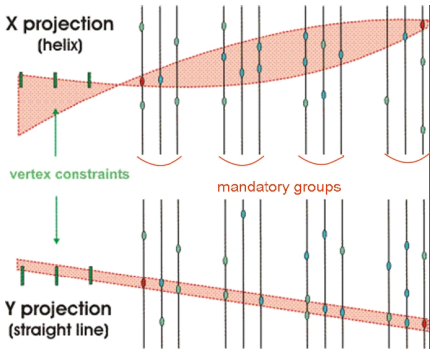
Figure 4.1: Average cluster size for a pixel plane at 9.9 cm for both PT7 polarities. Figure taken from Ref. [22].

instances of each kind of group. The groups may also overlap. As suggested by their names, at least one station in every mandatory group must contribute to every track, and no such imposition is made on the optional groups.

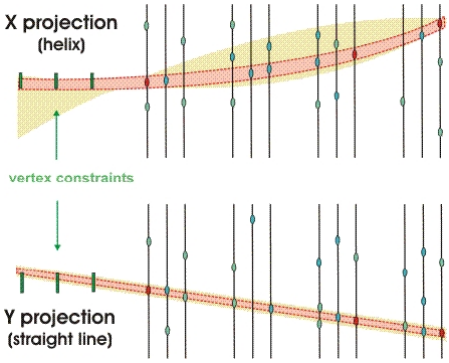
In order to reduce the combinatorics involved in searching for tracks in all clusters from all VT stations, which would basically make tracking intractable, reconstruction is done in several steps. The first of these involves creating *track seeds*, preliminary tracks that use only a few clusters. Since the Vertex Telescope planes are placed in the PT7 magnet's dipole field, a bare minimum of three space points is needed to measure a track's slopes in the x and y directions, as well as its curvature.

The dipole nature of the PT7 field and its orientation mean that there is no deflection in the y-z plane. Consequently, track projections in that plane will be straight lines (see Fig. 4.2). The first step in creating a track seed is to combine a cluster from an upstream station with another cluster from a downstream station. A relaxed cut is applied to ensure that the line defined by these two clusters points back to the target region. If so, an area is defined between these first two points in which a third cluster is searched for.

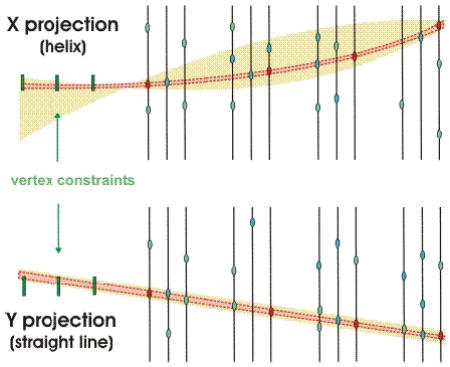
With no curvature measurement at this point, it is impossible to know the charge of the particle that caused the hits and so the search area has to account for both positive and negative hypotheses in the x-z plane. This is done by choosing maximum curvature values that correspond to a particle with 1 GeV/c momentum, one value for each charge. If another cluster is



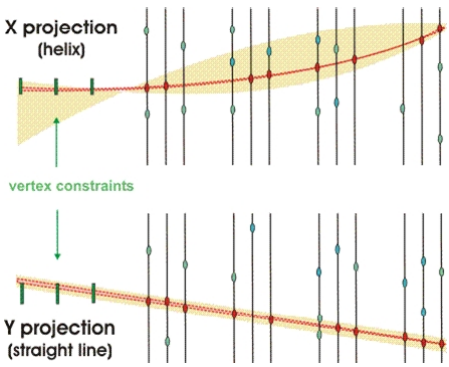
(a) Seeds pointing to the target in the non-bending y plane are selected and a search area in the bending plane is created.



(b) Clusters downstream are added and vertex constraints verified.



(c) Further clusters in the search cone are added iteratively until all mandatory groups contribute at least one cluster.



(d) The track is extrapolated to not yet used stations and clusters found in those are added.

Figure 4.2: Track reconstruction steps. Figures taken from Ref. [22].

found in this search area, a track fit is done on this three-cluster track seed, which is required to pass a loose χ^2 cut using the centre of the targets as a constraint.

Track seeds that pass the cuts are used to define a more restrictive search area, or "road width" for new clusters. Clusters are added first from mandatory, then optional stations, if they satisfy the following conditions:

- The cluster's χ^2 distance to the track has to be small.
- The refitted track must have a low χ^2.
- The new fit must have a curvature that corresponds to a momentum of at least 1 GeV/c.
- The extrapolation of the track to the target region must be close enough to one of the sub-targets

The track candidate composed of all the added clusters is accepted if it has at least one cluster in each mandatory group, and a certain minimum number of clusters overall, which may include optional groups. Once a list of track candidates has been compiled, cluster sharing is verified. If two or more tracks share a number of clusters that is greater than a pre-defined threshold, then only the one with the lowest χ^2 is kept.

For In-In data the reconstruction scheme outlined above is applied three times in a row with small modifications. Clusters used in the earlier steps are not used in the subsequent ones in yet another effort to reduce combinatorics, and at most one cluster is allowed to be shared among tracks.

The first step reconstructs tracks that have clusters in a large number of stations, with tight criteria. These are robust high quality tracks that are instrumental in the determination of the production vertex (explained in Section 4.2.3 below). The second step uses looser requirements to reconstruct the shorter tracks. Their limited size can be a result of their production in secondary interactions or the limited acceptance of the detector. These tracks are also required to point to the preliminary vertices reconstructed in the first step. In the last step the vertex convergence criteria are relaxed so as to reconstruct tracks with a displaced vertex. This is a fundamental feature for NA60's open charm studies, as well as for the analysis presented in this thesis.

The very last part of the track reconstruction process is to refit all fully reconstructed tracks using a Kalman filter. The implemented filter makes use of the knowledge of the detector components in estimating multiple scattering and energy loss in the VT planes, which in this case is the process noise, as well as the measurement error caused by the finite resolution of the detectors. After the application of the Kalman filter, which improves the momentum resolution of the Vertex Telescope, a smoothing step makes it possible to estimate the cluster coordinates with better precision.

Figure 4.3 shows a fully reconstructed event in the Vertex Telescope.

4.2.3 Vertex Finding

The vertex finding algorithm in NA60 uses reconstructed VT tracks to determine the spatial coordinates of the interaction(s) in an event. A least

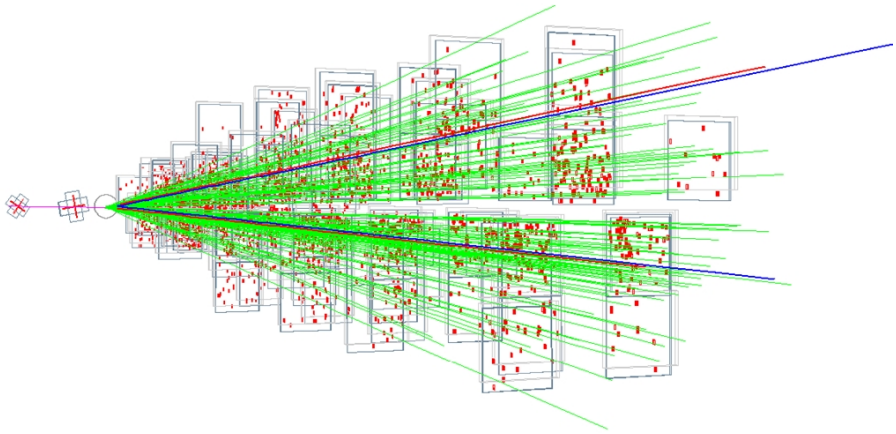


Figure 4.3: A fully reconstructed event in the VT. The blue lines represent matched muons. The Beam Tracker is shown on the left.

squares method, which was not used, assumes that the residuals are normally distributed. This is only true of a background free sample, which is not the case in NA60. The abundance of VT tracks and possible multiple interactions add contributions with large residuals that lead to unrealistically large errors in the estimation of the vertex position. For these reasons, the NA60 vertex finding algorithm is a variation of the iterative method mentioned in Ref. [23], which was developed to minimise the problems mentioned above. In this method, the influence of points with large residuals (which are mostly background), called *outliers*, is minimised.

Vertices are found in two steps. After the first track reconstruction step, a preliminary vertex is found using only tracks with a minimum of 5 clusters and with a track fit χ^2 of less than 1.5. After all VT tracks are reconstructed, the final vertex is determined using all tracks with more than 3 clusters and a track fit χ^2 of less than 2. The iterative process's stopping condition is met if the coordinates determined in the last two iterations differ by less than 100 μm.

Tracks with a small weighted difference to the found vertex are attached to it, which means that it is assumed that this vertex is their common point of origin. Once a vertex is found, a new search begins to find any other possible vertices with the remaining tracks (i.e. the ones that were not assigned to one of the previously found vertices). Once there are no more suitable tracks for vertex finding purposes, the algorithm attempts to attach any left-over tracks to one of the vertices. As this is not always possible, some tracks are not attached to any vertex in the event. These tracks are the ones analysed in this study, as shall be seen in the rest of this thesis. Tracks that are assigned to a vertex have a weight attributed to them that

corresponds to the likelihood that they actually originate from the vertex they are attached to. This weight is a number that ranges from 0 to 1.

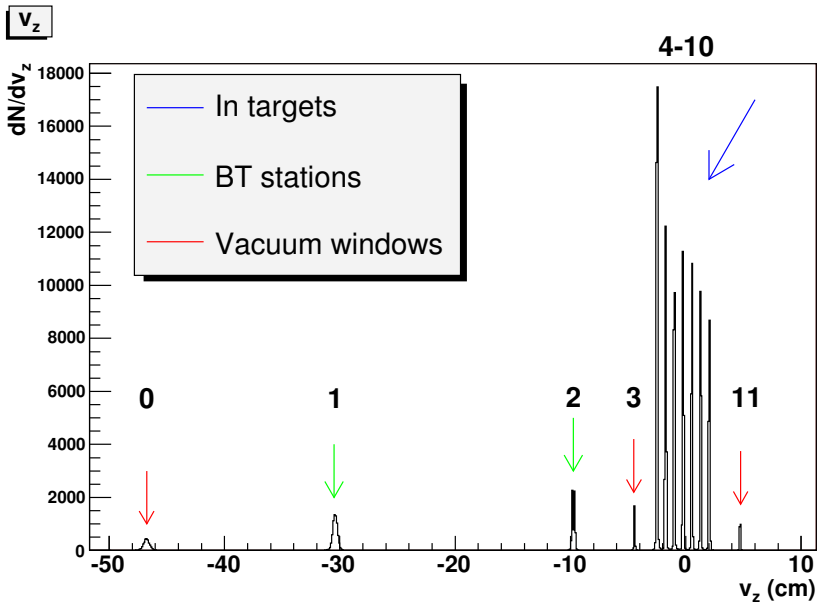


Figure 4.4: Vertex distribution along the z axis. The indium targets are shown on the right as well as the target box windows to either side of them. Upstream peaks caused by interactions in the Beam Tracker stations and its vacuum window are seen on the left. The numbers represent the target indexing used by the author.

Found vertices are assigned to a particular sub-target if they are within 3σ of its boundaries. Figure 4.4 shows the interaction vertex distribution along the z axis. The accuracy is such that the individual sub-targets are clearly visible, as well as the Beam Tracker stations and vacuum windows that are in the beam's path. A sizeable portion of the data, roughly 8% of the total, actually originates in these sources that are upstream of the indium targets. Attempts were made to analyse these data and are shown in Section 4.5.7. Each of the shown peaks is indexed after NA60ROOT reconstruction in the present analysis, ordered by increasing z and starting at 0. In this way, the first indium sub-target would be logical target 4.

4.2.4 Muon Matching

Muon matching in NA60 is the process through which the muons in the VT are identified. This is done by comparing the slopes and momenta of the VT tracks to the ones measured by the Muon Spectrometer, which are always taken to be muons since no other particle should be able to cross the hadron absorber.

This procedure is extremely important in most of the analyses done in the NA60 collaboration in muon studies, but it has no effect on the work done on this thesis. One could always exclude the matched muons from the V^0 analysis, since those tracks are almost guaranteed to not correspond to K^0_S or Λ decays. As seen in the Section 4.5 below, this would have a very small effect and so was not done.

4.3 Run Selection

Amongst the runs recorded to tape are special alignment and test runs. These are not used in the present analysis and include runs in which the ACM and/or PT7 currents were 0. Runs with fewer than 10 bursts are also ignored. The remaining runs were scanned to identify runs in which one or more detectors were not operating as they should. Runs were grouped together according to the sign of the PT7 and ACM currents and their proximity in time. A group of runs consists of a sequence of data taking periods with the same combination of PT7 and ACM magnet currents that ends when one of the polarities is changed.

A group of variables was selected to be indicative of good operating conditions. These variables were required to be stable within a run and across runs in the same group. Runs that presented abrupt changes were rejected. Furthermore, the values of the variables under study for run selection had to exhibit reasonable absolute values.

Table 4.1 shows the reasons used for run rejection along with their respective frequencies. In it, PC stands for "Proportional Chambers" and refers to dimuons reconstructed in the Muon Spectrometer, or MS.

Reason	**Frequency**
Pixel Plane Occupancy	52%
Muon Matching Rate	42%
PC Dimuons per beam particle	29%
VT Tracks per PC Dimuon	27%
Reconstructed PC Dimuons	24%
Beam Position	20%
MS Sextant Asymmetries	17%
Beam Intensity	15%
Like Sign Dimuon ratio	12%

Table 4.1: Run rejection frequencies per reason [22].

The rejection frequencies add up to more than 100 % since a run can be rejected for more than one reason. In fact, 60 % of the rejected runs are marked as such for 2 or more of the reasons shown. This is to be expected due to correlations between some of the studied variables that provide a measure

of redundancy. That is why some runs were rejected due to problems in other detectors even though the present analysis uses only VT data. Over half of the rejected runs were due to the occupancy of the Vertex Telescope being too high, which was a result of one or more planes being misconfigured. In the end, the total run sample used for analysis consisted of 417 runs.

4.4 Event Selection

This section describes the criteria used in selecting which events in a run were used for Physics analysis. It is divided in two parts that refer to two distinct regions in the z axis. Originally this study focused on finding Λ and K^0_S events in vertices upstream of the targets. It was thought, at the time, that the small distance between the targets and the Vertex Telecope, compared to their lifetime, made it difficult for them to be detected since most of them would decay further downstream. Once the signals were successfully found in these upstream vertices, however, similar sets of cuts allowed them to be found in the indium target produced collisions, and the focus shifted. Nevertheless, the event selection used for the In-Si event sample is discussed below.

4.4.1 In-In Event Selection

To obtain a clean event sample, only events with one reconstructed vertex were selected. This avoids contamination and background from, for instance, secondary interactions produced by the spectator nucleons from a first interaction. The sole event vertex had to have been assigned to logical targets 4 through 10, which are the 7 indium sub-targets that can be seen in Fig. 4.4. This resulted in an event sample of approximately 19.9 million events from the 417 runs, or an average of ∼47,700 events per run.

4.4.2 In-Si Event Selection

The main difference in the selection of events produced upstream of the targets is that the only event vertex has a z coordinate of less than -8 cm and more than -40 cm. It uses logical targets 1 and 2, which are the second and third peaks (from left to right) identified in Fig. 4.4. This event sample consists of 1.1 million events.

4.5 Track Selection Cuts

Mass distributions are obtained from the reconstructed tracks of an event by combining each track with all the other tracks in the event that have the opposite charge. The pair mass is calculated by assuming that the tracks

have the mass of one of the decay products one is interested in according to these decay channels:

$$K_S^0 \rightarrow \pi^+\pi^-$$

$$\Lambda \rightarrow p\pi^-$$

$$\bar{\Lambda} \rightarrow \bar{p}\pi^+$$

In the case of the K_S^0, for instance, that means assuming that all tracks in the VT are pions and have a mass of 139.57 MeV/c^2. For the Λ, positive tracks are assigned the proton's mass and negative tracks the pion's. In the absence of track selection cuts, the errors incurred by the assumptions made on the particles' identities add up to the extent that the signal peaks are not even visible (see Figs. 4.5 and 4.6).

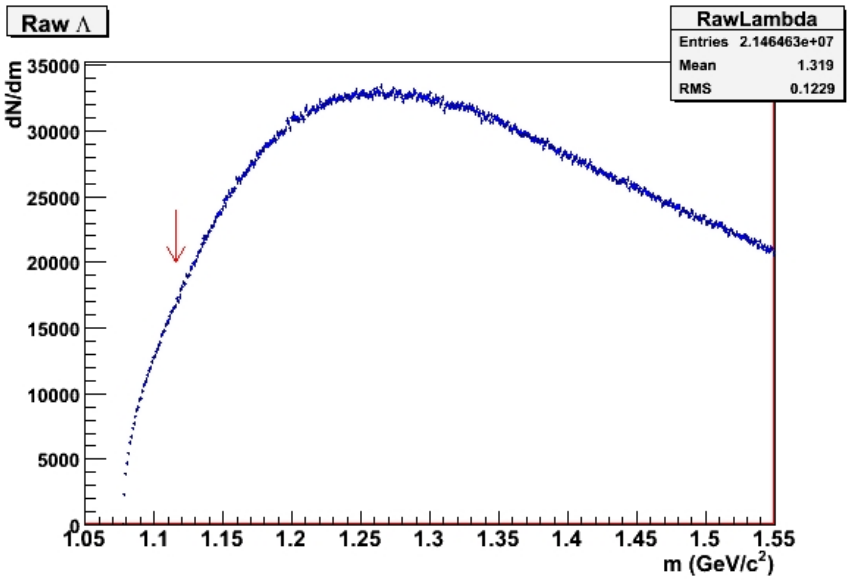


Figure 4.5: Λ mass distribution with no cuts.

This is because the NA60 apparatus was designed to detect and trigger on dimuons. The extremely clean signal present in the Muon Spectrometer due to the hadron absorber makes it possible to determine which Vertex Telescope tracks correspond to these muons, as explained above. This allows dimuon analysis to concentrate only on the matched muons in the VT.

In the present analysis, however, there are no additional data that single out which tracks are from Λ and K_S^0 decays, and they become a veritable needle in a haystack, lost in a sea of VT background tracks. For that reason, a series of track selection cuts have to be used so that the signal to background ratio becomes high enough for the peaks to be seen.

Figures 4.5 and 4.6 show the mass distributions obtained by combining the Vertex Telescope tracks in the manner described in the above para-

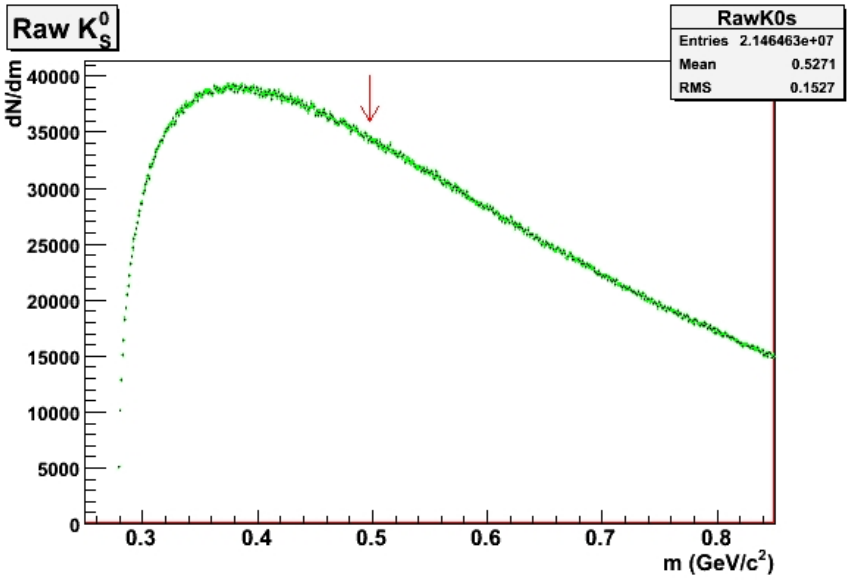


Figure 4.6: K^0_S mass distribution with no cuts.

graphs, with no track selection cuts. The red arrows represent the nominal mass for each particle. These figures show no distinctive characteristics in what should be the signal region.

4.5.1 Sets of cuts

There are three different sets of In-In track selection cuts and one for the In-Si data. Each of the In-In sets was an attempt to improve upon the signal to background ratio of the one(s) that preceded it. Even though in some cases using one particular set of cuts yields higher quality results than its predecessor, all the three sets were used to prune the data in order to calculate an estimate of the systematic error of the measurements.

The track selection cuts are explained below, together with the motivation behind each of them. The values the cuts take are presented in a table for each different cut. Sets 1, 2 and 3 refer to In-In cuts whereas set 4 refers to the cuts used for the upstream vertices. They were found partially guided by Monte Carlo studies that showed where it would be advantageous to cut, and partially by simply varying the cut values and verifying what effect they would have on the results. Unless otherwise noted, in the remainder of this thesis all In-In results refer to the third set of track selection cuts.

4.5.2 Collision Vertex cut

This is the only single-track cut used in the data analysis. All of the other cuts are track pair cuts, pertaining to kinematical variables of the hy-

pothetical mother particle of both tracks. This first cut attempts to identify background tracks and reject them if they appear to have been produced in the collision as opposed to in a V^0 decay.

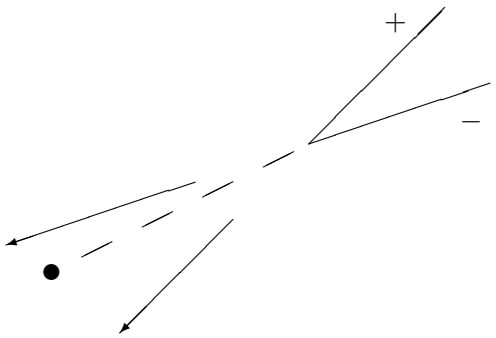


Figure 4.7: A V^0 decay. The neutral Λ or K^0_S is represented by the dashed line, and its daughters by solid lines. The extrapolations (arrows) of the daughter tracks do not necessarily pass close to the collision vertex, represented by the circle.

The neutral Λ and K^0_S particles are invisible to the Vertex Telescope and are detected through the tracks of their charged decay products. Although the mother particles themselves are primary particles, having been produced in the collision, their daughter particles are not, and as such should not appear to originate from the collision, as shown in Fig. 4.7.

Most of the tracks in the VT are caused by primary particles and are background for the purpose of this analysis. All tracks that are attached to the event vertex during the recontruction process are rejected. The one exception to this rule was the In-Si Λ. In its case it was found that accounting for tracks that may have been incorrectly assigned to a vertex increased the signal with respect to the background. This is not surprising given the large distances over which the measured tracks have to be extrapolated. The cut used for these data consisted of rejecting any track with a vertex attachment weight (as explained in Section 4.2.3) of more than 0.2.

4.5.3 Track Distance cut

The distance between two tracks is defined here as their spatial separation at the point of closest approach. Daughter particles of a V^0 decay originate at the same point in space, but will not appear to be at zero distance to each other to the detector due to its finite resolution. The goal here is to choose a cutoff value that is low enough to reject most of the background without sacrificing too much of the signal. Obviously there is a trade-off involved in the selection of the cut. This cut, while important,

cannot single-handedly cut away enough background for the signal peaks to become visible. This is because of tracks that are coincidentally close to each other, a common enough occurrence when the number of tracks in the Vertex Telescope is high.

Since the In-In and In-Si collision systems are very different with respect to their position along the z axis, the corresponding resolution with which charged particles originating from them can be tracked varies substantially. The kinematical distribution of decay products from the K^0_S meson and Λ baryons is also different enough to warrant different cut values for each of them. The Table 4.2 summarises the values used for the track distance cut.

Particle	**Set 1**	**Set 2**	**Set 3**	**Set 4**
Λ	50 μm	150 μm	130 μm	200 μm
K^0_S	75 μm	200 μm	160 μm	200 μm

Table 4.2: Track distance cuts.

Figure 4.8 shows how the background (taken to be the data without cuts) and Λ and K^0_S MC signals are distributed with respect to daughter track distance. The K^0_S signal is less peaked due to its smaller opening angle, resulting in a more smeared out distribution at distances approaching the detector's resolution.

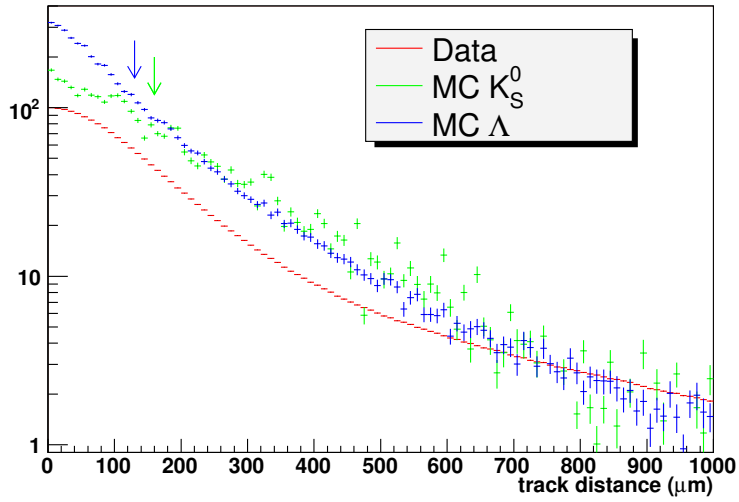


Figure 4.8: Track distance distribution for MC Λ signal (blue), K^0_S (green), and data (red). The arrows show the threshold for both particles for the third set of cuts.

4.5.4 Δz cut

The particles this study focusses on have decay lengths of the order of a few centimetres in the centre of mass frame (7.9 cm for the Λ and 2.7 cm for the K^0_S). With the average energies these particles are created at, according to VENUS, these numbers become 1.7 m and 72 cm, respectively, in the laboratory frame. The decay vertex can then be distinguished from the collision vertex by their spatial separation. Setting a minimum separation for the z coordinate of the point of closest approach and the collision vertex position along the beam, in which the former is always downstream of the latter, rejects background whilst keeping most of the signal. Figure 4.9 shows the distribution of MC-generated Λ and K^0_S as well as the background from the data. In the data's histogram peaks due to the In targets can be seen. This is to be partially expected since the targets are indeed the true source of (or close to the real source of) most of the detected particles. Table 4.3 collects the values used.

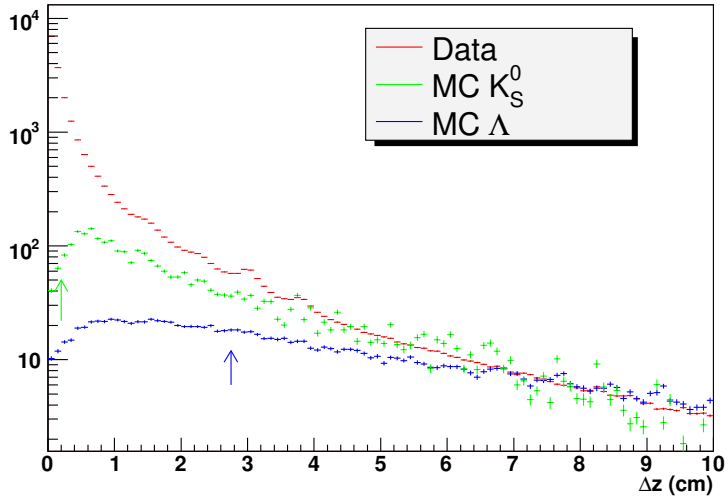


Figure 4.9: Δz distribution for MC signal (green and blue) and data (red). The arrows point to the values used for both particles in the third set of cuts.

Particle	Set 1	Set 2	Set 3	Set 4
Λ	2.5 cm	3.5 cm	2.75 cm	0 cm
K^0_S	0.875 cm	0.2 cm	0.2 cm	1.5 cm

Table 4.3: Δz cuts.

4.5.5 $z_{\max}$ cut

Figures 4.10 and 4.11 show how the data's mass distributions vary with the z coordinate of the point of closest approach for both the Λ and K^0_S. For both particles, there is an upwards mass shift with increasing z, accompanied by an increase of the peak width. There is also evidence of increased background in these downstream distributions, which lead to large systematic uncertainties. All of these effects are reproduced by Monte Carlo simulations. In an attempt to improve the quality and mass resolution of the collected data, a cut on the maximum z component of the V^0 decay vertex was imposed. Table 4.4 collects the values used for the $z_{\max}$ cut.

Particle	**Set 1**	**Set 2**	**Set 3**	**Set 4**
Λ	4 cm	4 cm	5 cm	n/a
K^0_S	1 cm	1 cm	1.5 cm	n/a

Table 4.4: $z_{\max}$ cuts.

The K^0_S distributions in particular show a peak at low mass which increases in importance for higher z. It is clearly visible in Fig. 4.12 which shows the K^0_S mass when no $z_{\max}$ cut is applied. Given this peak's location near the K^0_S production threshold, it is possible that it is due to photon conversions into an electron/positron pair. In this case the threshold mass would be "acquired" by assuming that the electron and positron both have the mass of a pion. Figure 4.13 seems to support this hypothesis, showing that this low mass peak is also located at low energy, which is to be expected if the pair has, in fact, mass zero.

4.5.6 Δxy cut

This last cut is designed to only analyse pairs of tracks that point back to the collision vertex. This means that their hypothetical mother particle has a momentum such that, when extrapolated back to the z coordinate of the collision vertex, has a distance to the said vertex that is small in the xy plane. Once more, this will generally be true of signal tracks and only coincidentally satisfied by the background ones. Table 4.5 collects the values used for the Δxy cut.

Particle	**Set 1**	**Set 2**	**Set 3**	**Set 4**
Λ	100 μm	50 μm	80 μm	1000 μm
K^0_S	200 μm	90 μm	140 μm	1000 μm

Table 4.5: Δxy cuts.

Figure 4.14 shows the Δxy distributions of the Λ and K^0_S signals (from the Monte Carlo) and of the measured data.

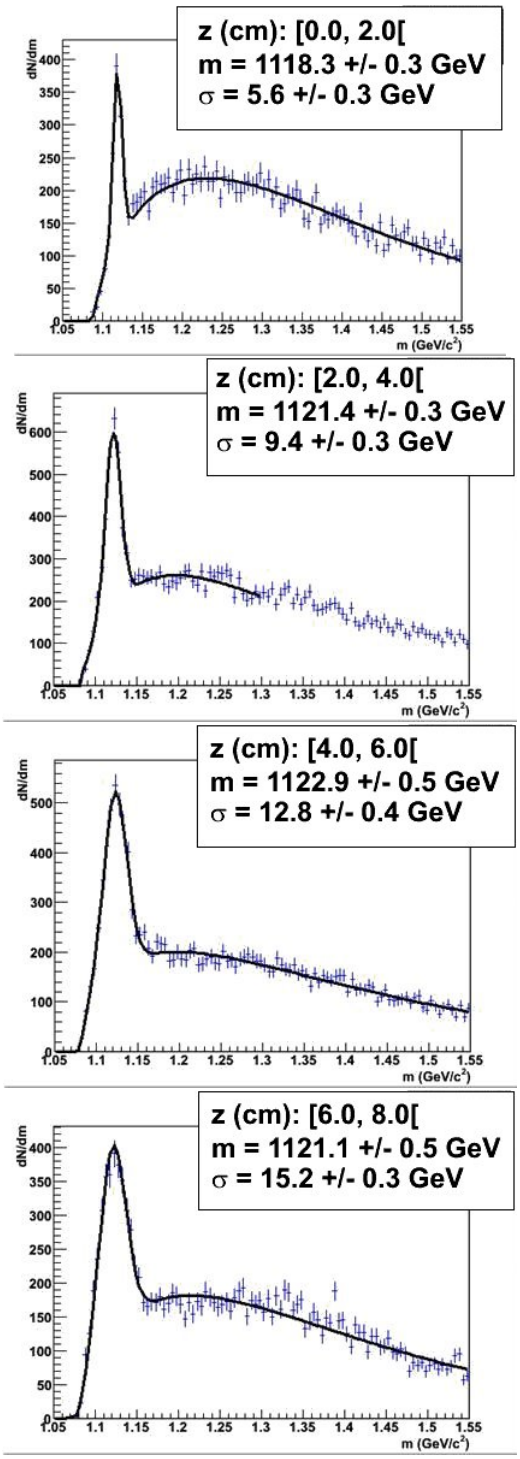


Figure 4.10: Λ mass distribution vs. z. The first plot starts at 0 cm and each plot spans an interval in z of 2 cm.

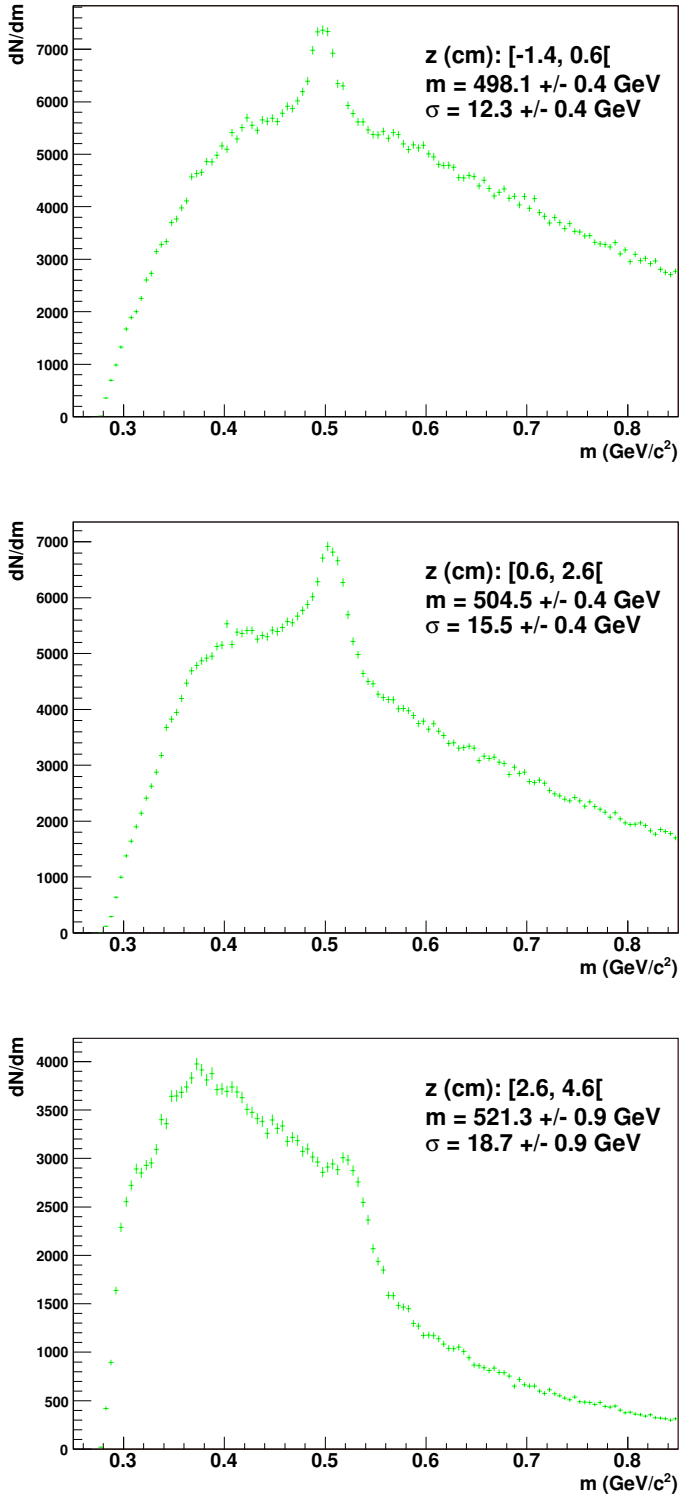


Figure 4.11: K^0_S vs. z. The first plot starts at -1.4 cm and each plot spans an interval in z of 2 cm.

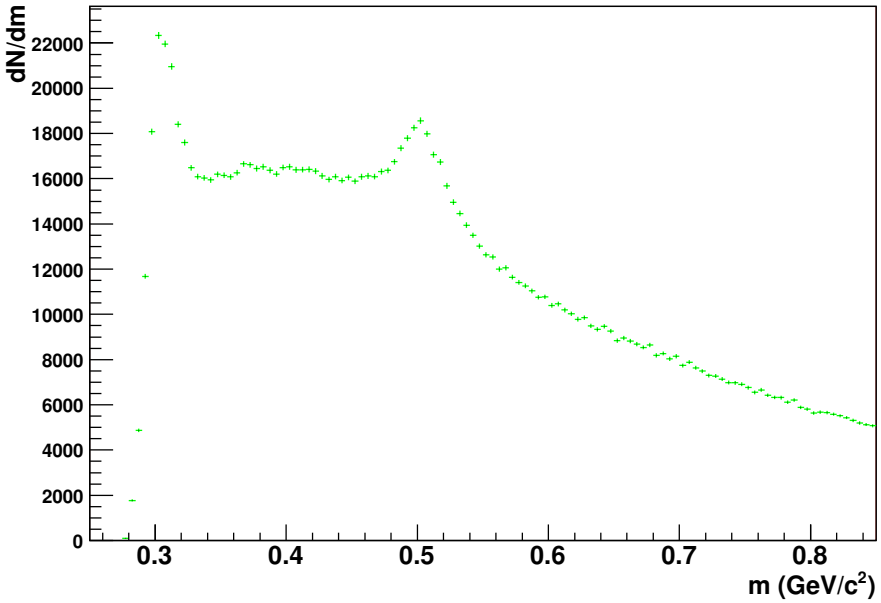


Figure 4.12: K^0_S mass distribution in the absence of $z_{\max}$ cut.

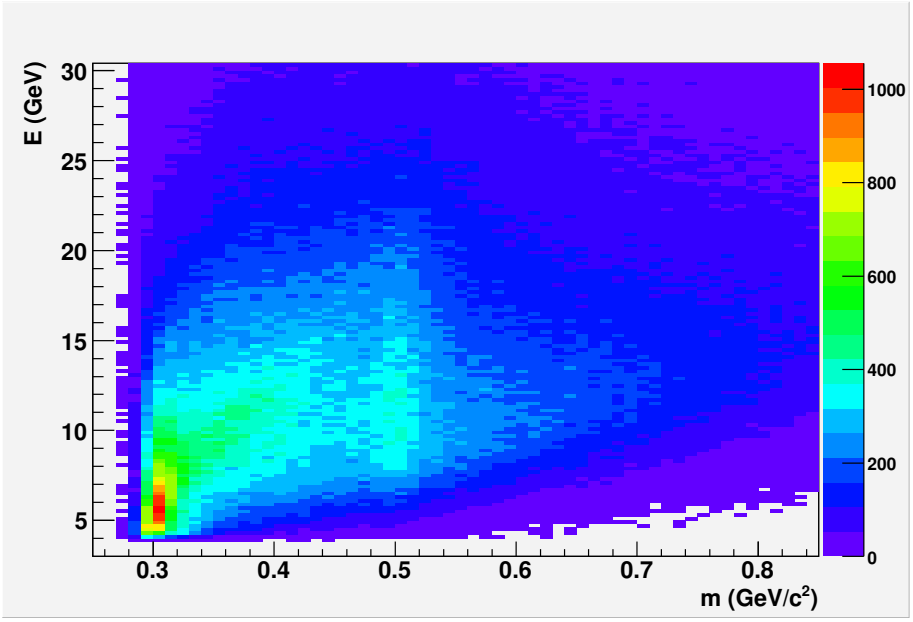


Figure 4.13: K^0_S mass vs. energy in the absence of $z_{\max}$ cut.

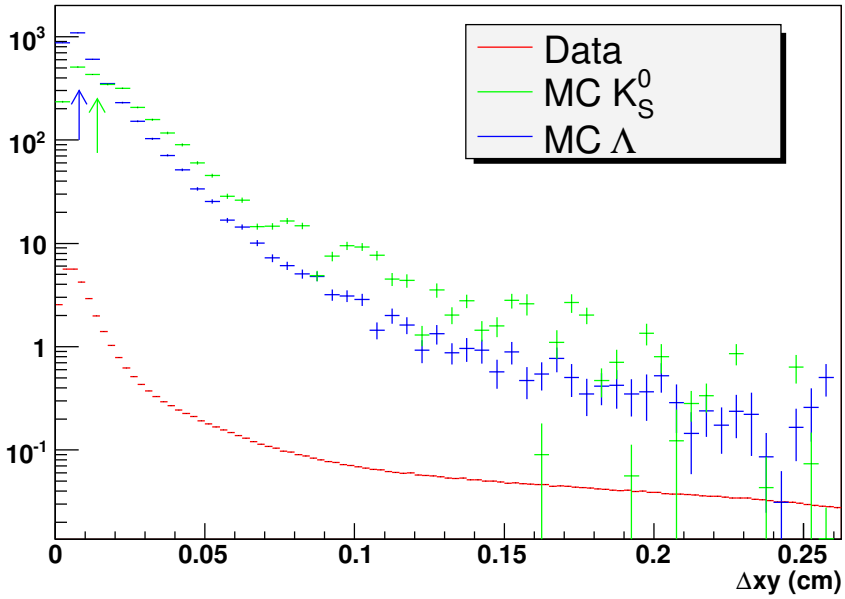


Figure 4.14: Δxy distribution for MC Λ signal (blue), K^0_S signal (green), and data (red). The arrows point to the values used in the third set of cuts for each particle.

4.5.7 Mass Distributions After All Cuts

Figures 4.15 to 4.17 show the mass distributions obtained from the selected In-In data, applying the third set of cuts previously described in this chapter.

Figure 4.18 is the counterpart to Fig. 4.15 for the upstream event sample, which consists of events caused by interactions in the Beam Tracker. Only one set of cuts was used for this sample.

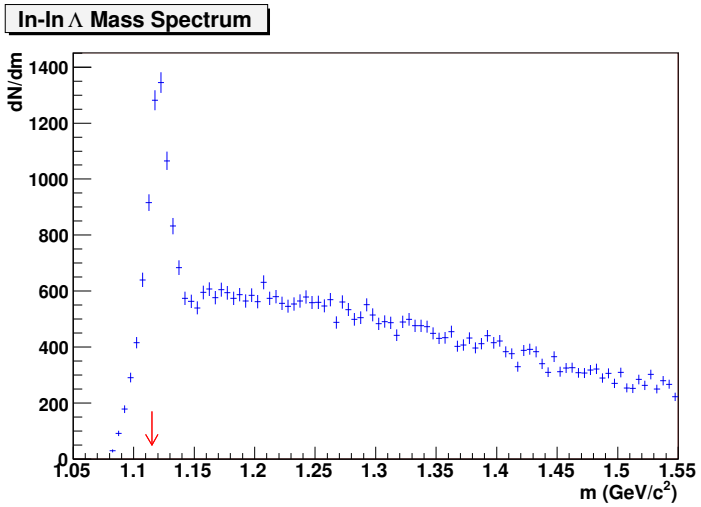


Figure 4.15: In-In Λ mass distribution before background subtraction. The red arrow indicates the nominal mass.

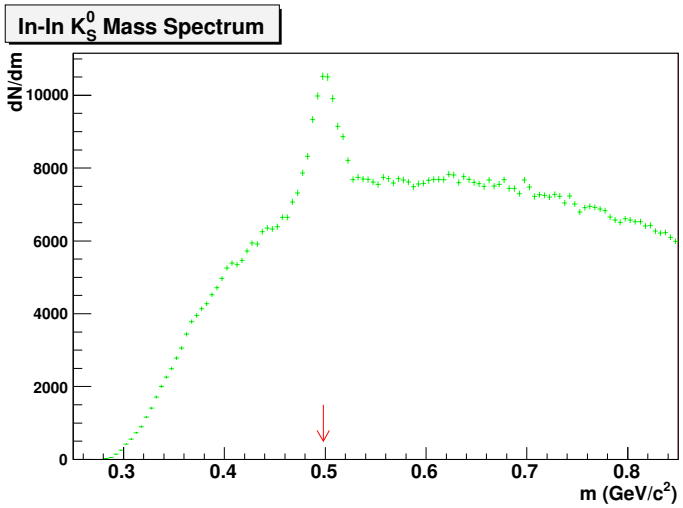


Figure 4.16: In-In K^0_S mass distribution before background subtraction. The red arrow indicates the nominal mass.

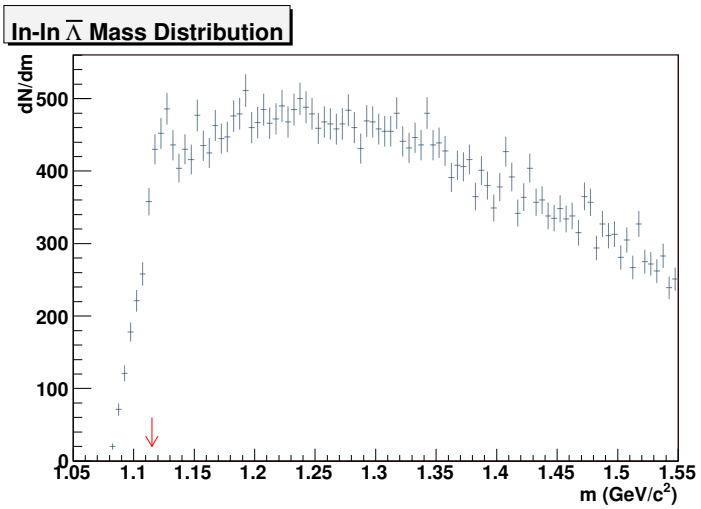


Figure 4.17: In-In $\bar{\Lambda}$ mass distribution before background subtraction. The red arrow indicates the nominal mass.

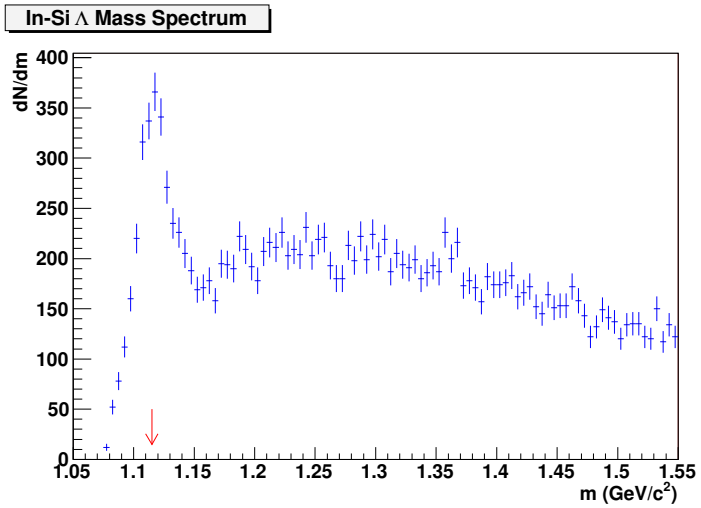


Figure 4.18: In-Si Λ mass distribution before background subtraction. The red arrow indicates the nominal mass.

Chapter 5

Analysis and Results

5.1 Overview

This chapter describes how the transverse momentum and rapidity spectra are extracted from the raw data collected by the Vertex Telescope (VT). It starts with the background subtraction techniques in Section 5.2. The track selection cuts do a remarkable job of raising the signal to background ratio to acceptable levels, but even when using strict cuts some background remains and must be subtracted. The process by which the particle counts are determined after this subtraction is done is also explained.

Section 5.3 explains how the corrections for acceptance and efficiency of the Vertex Telescope were calculated. This is needed so that the production p_{T} and rapidity spectra can be inferred from the measured values and involves Monte Carlo simulations, which are also described.

The final distributions are shown in Sections 5.4, 5.5 and 5.6, which consist of the result of applying the aforementioned corrections to the raw data yields in transverse momentum and rapidity.

5.2 Background Subtraction

The percentage of VT tracks in any given event due to background processes overwhelms the Λ and $\mathrm{K}^0_{\mathrm{S}}$ signals by orders of magnitude. Monte Carlo studies show that on average 0.17 tracks from $\mathrm{K}^0_{\mathrm{S}}$ decays are reconstructed per In-In event. This is to be compared with the measured average VT multiplicity of ~ 116. The track selection cuts described in the previous chapter reject most of the background contributions. The tracks that do pass the cuts combine to produce a mass distribution in which the signal peaks are clearly visible, but in which the background still plays a significant role, as shown in Figs. 4.15 and 4.16.

In order to extract the particle yields from the mass distributions this residual background must be subtracted, and for that to be done its contri-

bution must be estimated. This can be done in a variety of ways, a few of which are explained below.

5.2.1 Mixed Event Background

The mixed event background estimation method has the advantage of not having to rely on Monte Carlo simulations. Instead, it uses experimental data to combine random track pairs that belong to different events. These pairs are assumed to be background, and can be used to statistically determine the underlying background shape. It is used extensively in NA60 to determine the shape of the combinatorial background in dimuon studies [24], the main source of which are uncorrelated decays of charged pions and kaons.

Great care has to be taken to only combine tracks from events with similar characteristics so that the results are not inadvertently biased. This is because the real background's source is tracks from the same event. Belonging to the same event means that certain correlations between them exist that must be reproduced in the mixed event sample. The two events to be mixed must then be different, but have as many similar characteristics as possible.

In order to do so, events need to be categorised according to all variables that may have an impact on kinematics, multiplicity, or reconstruction efficiency. Candidate variables range from total VT track multiplicity, sign of the ACM and PT7 magnet currents, subtarget in which the collision took place, and many others. The more categories used, the more likely it will be that events belonging to them, when mixed together, will yield a faithful representation of the real background distribution.

This techique eventually proved to be unnecessarily cumbersome compared to the method described below. Given that the latter produced high quality results, the mixed event background subtraction was not used in this analysis.

5.2.2 Side Bin Fitting

In this method the mass spectrum is fitted to a function in a mass region where the signal contribution is considered to be negligible. First a window is defined where most or all of the signal resides, with some underlying background which is unknown. Then the two regions to either side of this signal window, the side bins, are used to determine the background shape. The peak window range does not contribute to this fit in any way except for defining where the lower bin ends and the higher bin begins.

The fit thus obtained is extrapolated to the central signal region where it is subtracted from the original mass histogram's bin contents on a bin by bin basis, calculating what the fitted function's value would be at the centre

of each mass bin. The result of this subtraction is fitted to a gaussian, which in turn is integrated in order to extract the signal yield.

At this stage an estimate of the systematic error of the yield calculation is done by also fitting the unsubtracted histogram to a function equal to the background shape added to a gaussian. All of the parameters in this second fit are free but are initialised to the values from the previous fits. Integrating the gaussian part of the function results in a second value for the particle yield. The third value used for this systematic error evaluation is the result of adding up the bin contents of the subtracted histogram in the signal region, without performing any fit. This third value is used in order to avoid any error introduced by the assumption that the shape is gaussian.

The two functions used to fit the background shape are presented below.

Fourth order polynomial fit

The first function that was used in an attempt to fit the background was a fourth order polynomial, $p_0 + p_1 m + p_2 m^2 + p_3 m^3 + p_4 m^4$. This produced satisfactory results when fitting the background in the K^0_S plots. The Λ mass distributions however, presented greater difficulties, probably due to the proximity of the signal peak to the proton+pion mass threshold.

In Figs. 5.1 to 5.4, the larger histogram shows the total fit (gaussian + background function), the top-right histogram is the signal histogram that is fitted to a gaussian (equal to the larger histogram minus the background function fit), and the bottom-right plot is the side bin background fit.

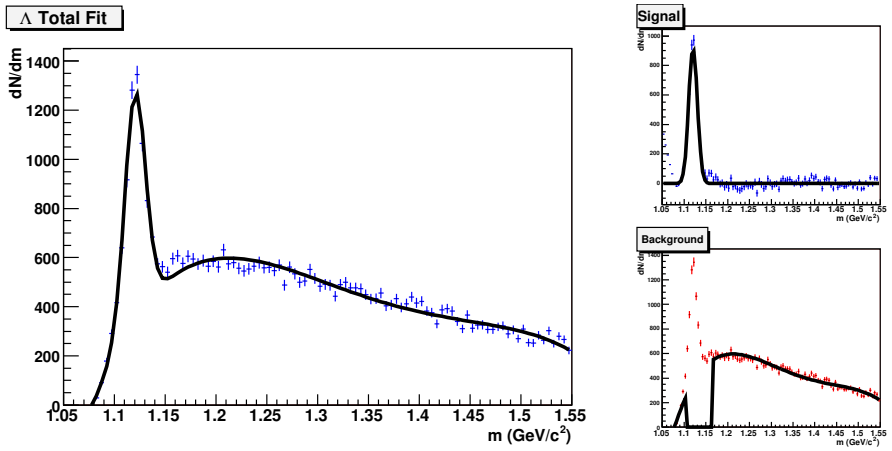


Figure 5.1: Polynomial background fit of the In-In Λ. The gaussian fit yields a mass of 1120.5 $\pm$ 0.2 MeV with a σ of 10.2 $\pm$ 0.2 MeV.

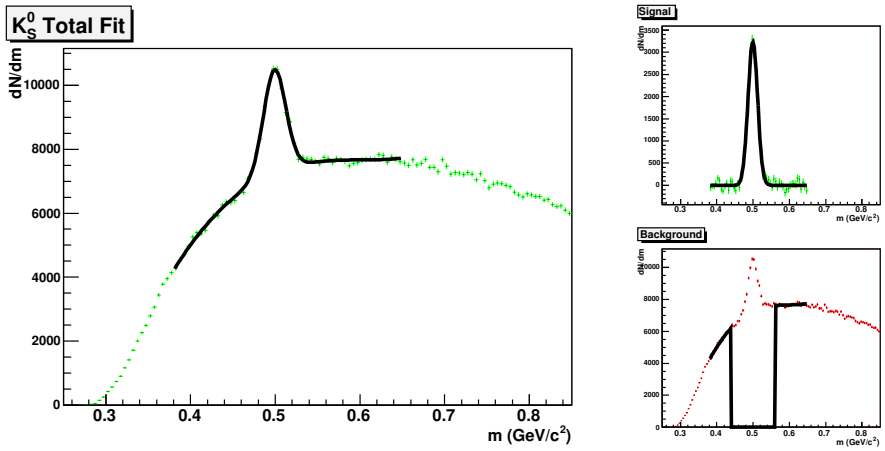


Figure 5.2: Polynomial background fit of the In-In K^0_S. The gaussian fit yields a mass of 499.3 ± 0.3 MeV with a σ of 13.3 ± 0.3 MeV.

Γ-function fit

In an attempt to improve the Λ background fit, another function was considered, the Γ-function given by

$$\Gamma(m) = (m - m_0)^n e^{-\alpha m} \quad ,$$

where m_0 is the mass threshold, n the rise at threshold and α the decay at higher masses. Using this function on the Λ distributions yields much better results than the previous polynomial fits, as can be clearly seen in Figs. 5.3 and 5.4.

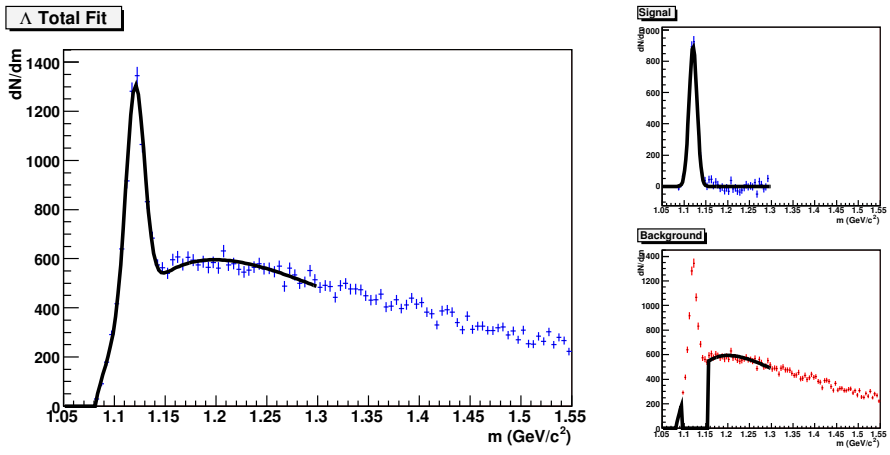


Figure 5.3: Γ background fit of the In-In Λ. The gaussian fit yields a mass of 1120.2 ± 0.2 MeV with a σ of 9.3 ± 0.2 MeV.

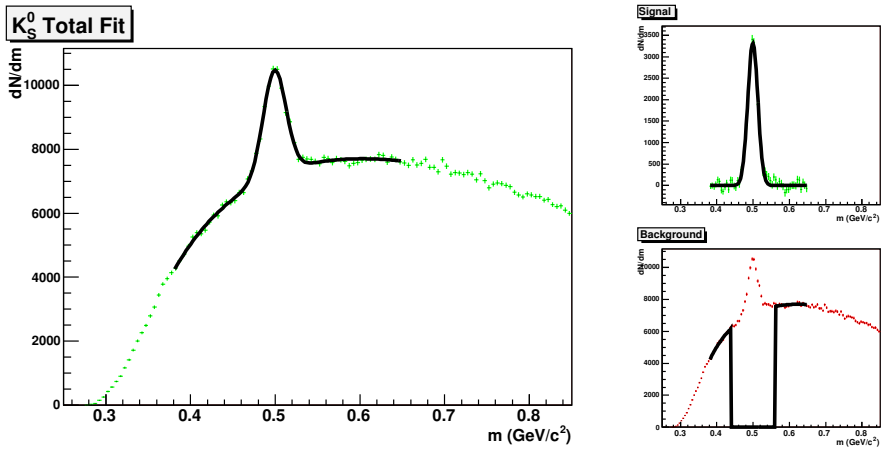


Figure 5.4: Γ background fit of the In-In K^0_S. The gaussian fit yields a mass of 499.3 ± 0.3 MeV with a σ of 13.9 ± 0.3 MeV.

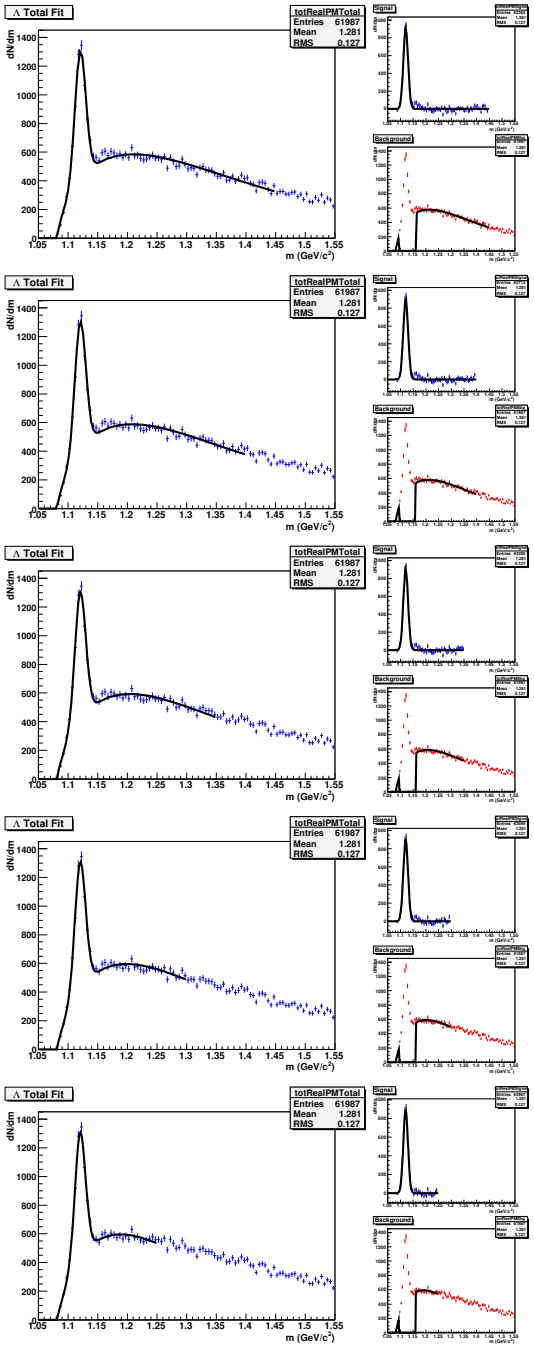
Although the K^0_S background was well described by a fourth-order polynomial, the Γ-function was also used in its background determination since that provides a further estimate of the systematic error associated with the process. Only the Γ-function was used in the Λ background fits since using the less adequate polynomial fits in this case would bias the results.

Systematic Errors

Two methods for estimating the systematic error have already been described, one of which applies only to the K^0_S (using both polynomial and Γ functions in its background determination). The most significant component of the systematic error has been determined to be the interval over which the background is fitted, with a smaller contribution from the peak range, i.e. the range excluded from the background fit.

Similar background fit intervals, even those that end or start far away from the signal peak, (e.g. [1.0,1.5] and [1.0,1.4]) sometimes result in particle counts that differ by as much as 5% or 10%. Each particle yield was obtained using several different background fit intervals, with the final value equal to a weighted average in which the weights were the inverse errors of each separate "measurement". The systematic error was then taken to be the RMS, similarly weighted, of all these different fits.

Badly behaved fits, identified visually and/or by their unusually large errors, were excluded from this process. In the case of the K^0_S the fit intervals were first varied for the polynomial fit then once more for the Γ-function fit. Figures 5.5 and 5.6 show the Λ yield and systematic error calculation for one particular mass distribution.



[1.05,1.45]: $n_\Lambda = 4220 \pm 160$

[1.05,1.40]: $n_\Lambda = 4220 \pm 160$

[1.05,1.35]: $n_\Lambda = 4200 \pm 160$

[1.05,1.30]: $n_\Lambda = 4160 \pm 150$

[1.05,1.25]: $n_\Lambda = 4090 \pm 130$

Figure 5.5: Case study of Γ-function background fit of the Λ mass spectrum. The yields calculated from each fit are shown to the right of each plot, as well as the interval used for the fit itself, in GeV. The error of each yield calculation contains the systematic error evaluation explained in the text. The total number of Λs in this particular mass spectrum is then taken to be 4180 ± 160, where the systematic error contribution to that error is 50.

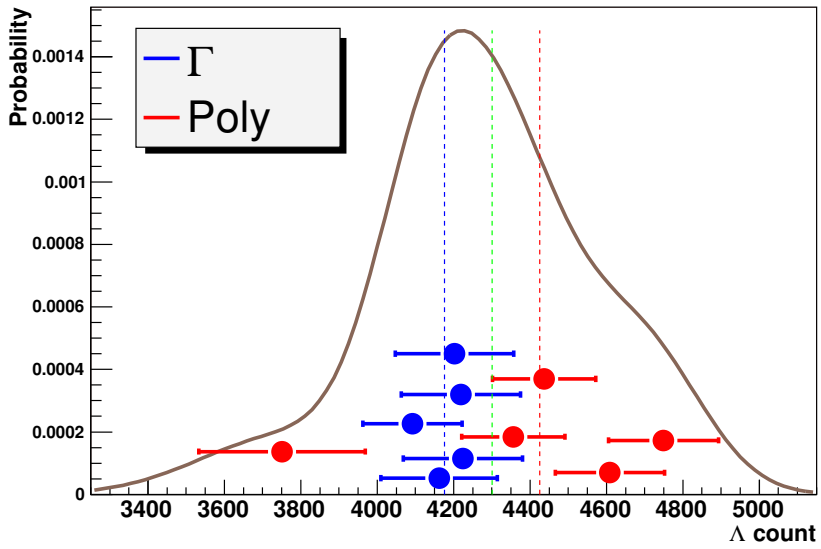


Figure 5.6: Case study of the background subtraction systematic error vs. fit function. The red points are the results of subtracting the background using a polynomial fit, and the blue points the yields obtained with a Γ background fit. The red and blue dashed lines are the Λ count that would be obtained by only using the polynomial or Γ fits, respectively. The green dashed line is the extracted Λ yield using points from both background fits. Only the Γ fit was used in the present analysis.

5.3 Acceptance and Efficiency Corrections

5.3.1 Introduction

The measured data do not correspond exactly to the physical distributions of the particles that emerge from the collision, due to several factors. First and foremost, the Vertex Telescope has limited acceptance. The region of phase space that it covers depends on the PT7 magnetic field, as well as on the geometrical placement of its planes. Furthermore, the selection cuts used to reduce the background might also place constraints on which signal tracks cross enough VT planes to be reconstructed.

The efficiency with which the tracks are reconstructed also affects the part of the signal that is effectively "seen" by the detector. A track may lie within the acceptance of the detector but not be reconstructed, which may happen more for some regions of phase space than others.

In order to arrive at the correct rapidity and transverse momentum distributions of the particles produced in the collision, correction factors must be calculated for both detector acceptance and reconstruction efficiency,

which can be done with the aid of Monte Carlo (MC) simulation. The following sections explain how these corrections were calculated in the present study.

5.3.2 Event Generation

The Vertex Telescope acceptance is a function of the rapidity and transverse momentum distributions of the particles produced in the In-In collision. A correction factor must be calculated for each bin in p_{T} and y by generating an initial phase space distribution. Given sufficiently small bins such that the variation of this distribution is negligible across each bin, any initial function can, in principle, be used. To further limit any systematic bias involved in using unrealistic spectra, the VENUS event generator was chosen to provide In-In collisions from which the simulated Λs and $\mathrm{K}^0_{\mathrm{S}}$s were taken.

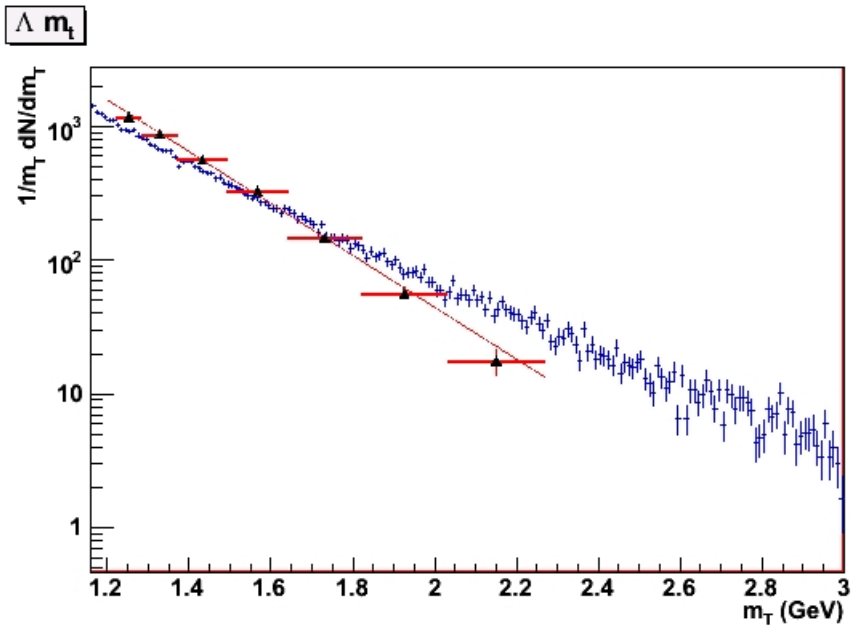


Figure 5.7: Generated In-In Λ m_{T} spectrum (blue) compared to NA57 Pb-Pb data from Ref. [25] (red/black). $T_{\mathrm{data}} = 289 \pm 30$ MeV, $T_{\mathrm{MC}} = 297 \pm 7$ MeV.

Figures 5.7 to 5.10 show the Λ and $\mathrm{K}^0_{\mathrm{S}}$ p_{T} and y generation distributions from the VENUS In-In collisions compared to experimental data collected in Pb-Pb collisions by NA57 and NA49. The rapidity distributions for both particles have peaks at midrapidity, which is approximately 2.91 at 158 GeV. The Λ's rapidity plot is much flatter, corresponding to its main production mechanism, which is to "transform" a target or beam proton by exchanging

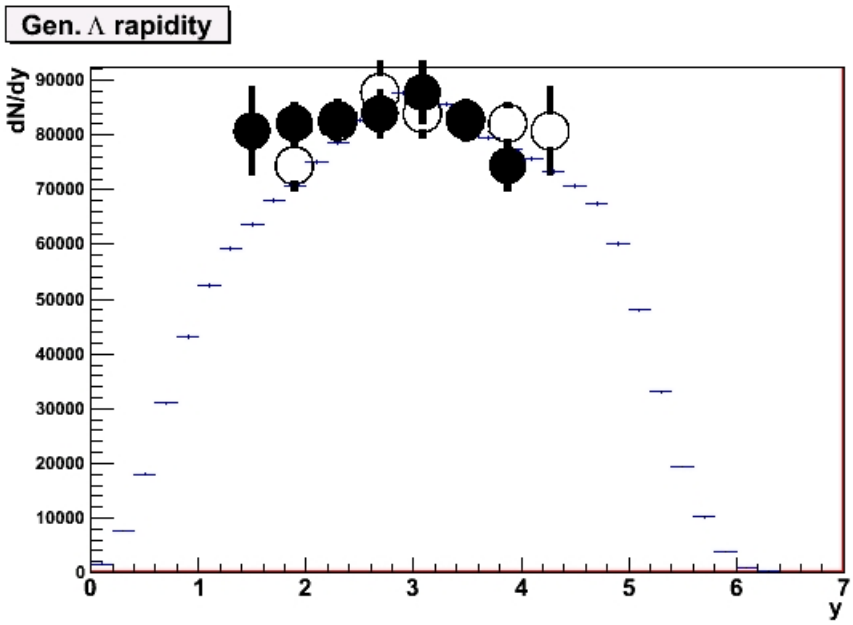


Figure 5.8: Generated In-In Λ rapidity spectrum (blue) compared to NA49 Pb-Pb data taken from Ref. [26] (black). Open points represent the measured (full) points reflected with respect to mid-rapidity (2.91).

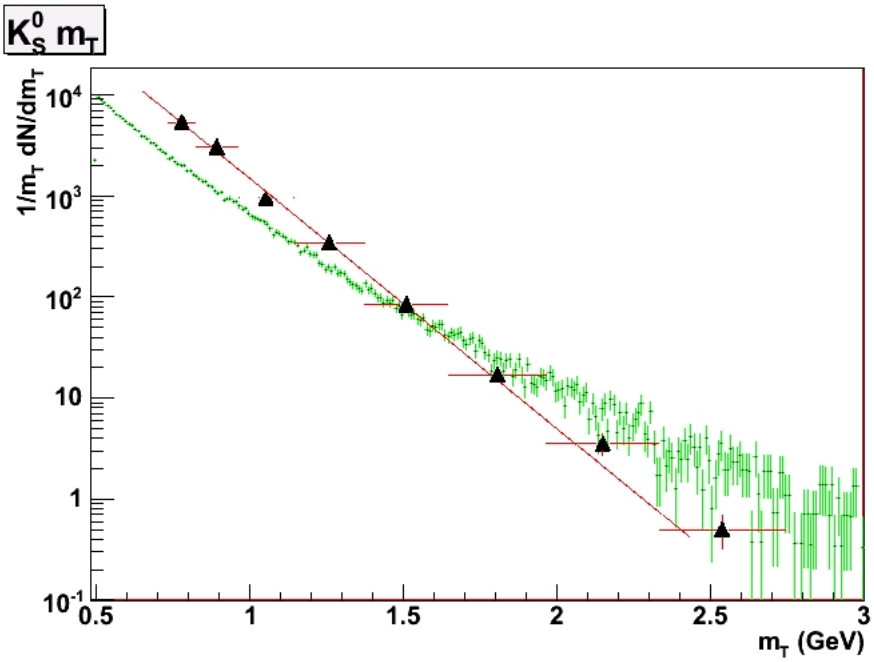


Figure 5.9: Generated In-In K^0_S m_T spectrum (green) compared to NA57 Pb-Pb data from Ref. [25] (red/black). $T_{\mathrm{data}} = 237 \pm 24$ MeV, $T_{\mathrm{MC}} = 250 \pm 18$ MeV.

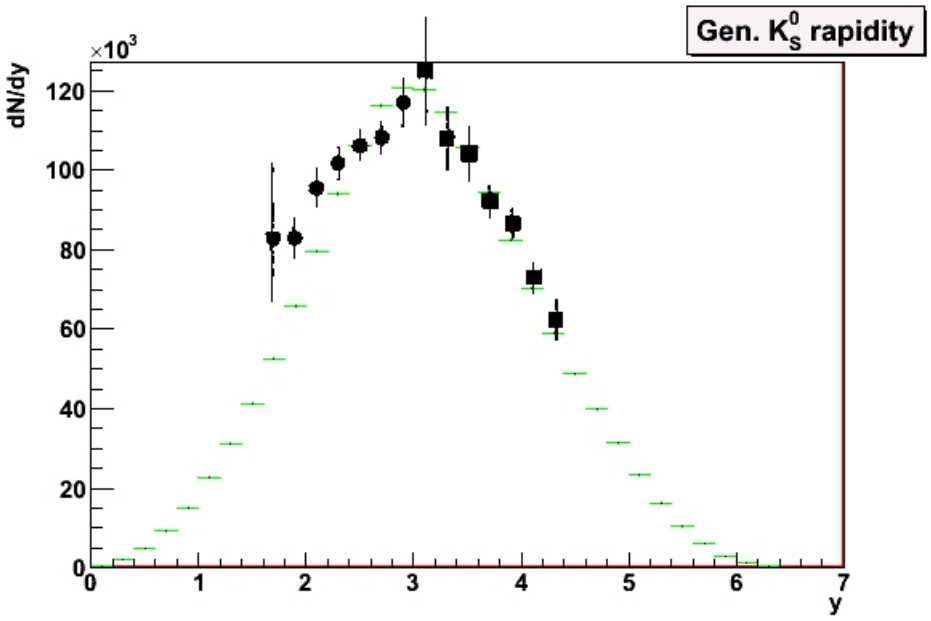


Figure 5.10: Generated In-In K^0_S rapidity spectrum (green) compared to NA49 Pb-Pb data taken from Ref. [27] (black). Square and circle points correspond to two different measurements.

an up quark for a strange quark. These protons lie at low and high rapidity, respectively.

5.3.3 Tracking

GEANT [28] is a program that simulates how particles are propagated through matter, in this case the experimental setup. The detectors are all described by geometrical volumes composed of a certain material, some of which are designated as sensitive volumes. These correspond to the active parts of the detectors, the ones responsible for detecting particles. The VENUS generated particles are transported through NA60's detectors by GEANT, simulating interactions between them and the detector elements, depositing charge and generating hits. These Monte Carlo hits are stored in the same format as used for the data so that they can be reconstructed by the same NA60ROOT routines.

Only a small fraction of the particles that result from an In-In collision are a Λ or a K^0_S. So, to prevent needlessly spending CPU time on all these other particles, they were not tracked. When simulating K^0_S decays, the charged pions it decays to (but not other pions) and the kaons themselves were tracked, but not any other particle. In a similar manner, only protons (anti-protons), π^- (π^+) and the Λs ($\bar{\Lambda}$s) themselves were tracked when

calculating the Monte Carlo corrections for the Λ ($\bar{\Lambda}$).

5.3.4 Standard Monte Carlo

To optimise the time taken on acceptance and efficiency calculations by ensuring the maximum number of reconstructable tracks per Monte Carlo event, the K^0_S and Λ tracks were forced to decay by imposing a maximum z coordinate on the decay vertex. The two-body V^0 decay was calculated assigning the ϕ and θ spherical angles in the centre of mass frame according to a uniformly distributed random function in ϕ and $\cos\theta$. A weight was also attached to each decay. This decay weight is the product of the branching ratio of the decay channel and the probability that the decay would have taken place. This last term, the decay probality, is given by

$$P = 1 - e^{-\frac{l}{\gamma c \tau}}$$

where $l = (z_{\text{max}} - z_{\text{vertex}})/\cos\theta_{\text{lab}}$ is the distance travelled by the mother particle from its production point to the decay vertex.

The decay weight, apart from increasing statistics per unit of CPU time, has the added benefit of labeling each V^0 decay. It prevents background from, for instance, pions from different K^0_S decays in the same event. This is because pions from the same decay must necessarily have the exact same decay weight.

To ensure that no systematic errors are introduced by forcing the studied particles to decay, the acceptance corrections were also calculated with unforced decays (i.e. handled by GEANT). Results are shown for both forced and unforced decay MC correction techniques in Fig. 5.11.

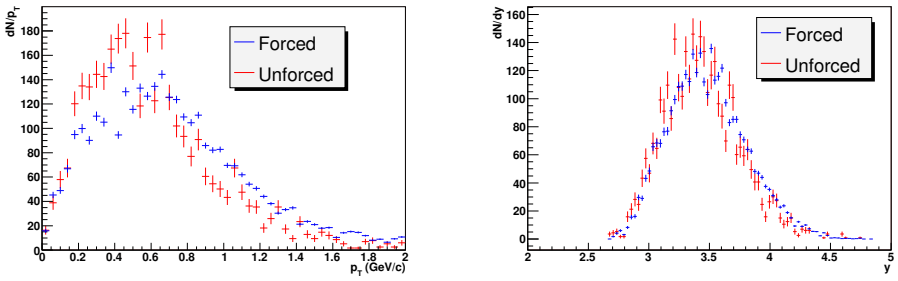


Figure 5.11: Forced vs. unforced decay overlay MC in p_T (left) and rapidity (right). In both plots the unforced MC is represented by the red histogram.

The generated Monte Carlo events are subjected to the same reconstruction procedures as the real data is, turning the simulated hits in the VT planes into reconstructed tracks that can be analysed just as the data was. This analysis includes all of the data selection cuts, except for the collision

vertex one. None of the charged particles present in the MC come from the interaction that took place in the target, so there is no way of computing the vertex appropriately nor of estimating if any of these tracks would have erroneously been assigned to it. In this context, the collision vertex cut loses meaning and so it is only used in the efficiency calculations described below. Essentially, this cut affects and depends on the reconstruction efficiency, not the detector acceptance.

After obtaining the reconstructed MC tracks, p_{T} and y spectra are produced by pairing up all of the tracks that pass the cuts in each event and calculating the relevant quantity. With forced decays there is no danger of mixing daughter particles from different mothers. Here only tracks with the same decay weight, which is propagated from generation to reconstruction, contribute to the relevant histogram. All other track pairs have weight zero. The unforced MC, however, has to rely on the data selection cuts to eliminate background from mixed decays and/or secondary interactions.

Figure 5.12 compares the data spectra with the standard Monte Carlo distributions.

5.3.5 Overlay Monte Carlo

Overlay, or embedded, Monte Carlo is a technique which involves using real data and simulation tracks together. The MC data, which in NA60ROOT has the same format as the real data before reconstruction, is mixed with raw data events and then these hybrid events are reconstructed. To ensure full compatibility with the data, the MC events are generated on interaction vertices taken from the data they are to be mixed with.

Ideally, to ensure the minimum possible disturbance and therefore deviation from reality, only two simulated tracks should be added to each data event (i.e. one V^0 decay). This is not very practical since it implies a lot of CPU time to obtain reasonable results. The average track multiplicity in the Vertex Telescope of the event sample under study is 116.2, and so adding a few more tracks per event should have a negligible effect on the calculations. This is confirmed by the similar results obtained from forced and unforced decay overlay MC events. The unforced case, of course, has fewer tracks since not all of the particles in each event decay upstream enough for its daughter particles to be detected by the Vertex Telescope. Figure 5.11 compares the forced decay overlay Monte Carlo to the unforced simulation in both p_{T} and rapidity.

The overlay MC should theoretically give more realistic results than the standard simulation technique described above. It has the added advantage of calculating the efficiency correction as well as the acceptance correction, since the simulated VT clusters are reconstructed together with the real data. Because of that, here the interaction vertex cut is used, and the efficiency with which a signal track is assigned correctly to the vertex included

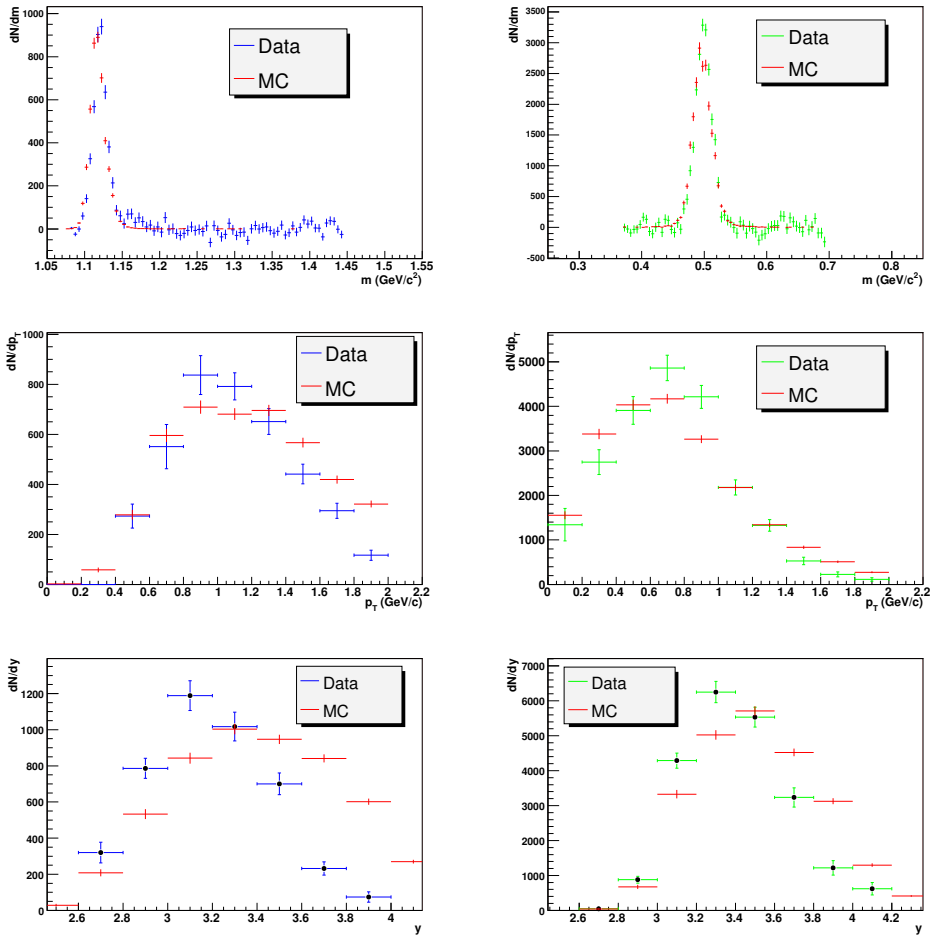


Figure 5.12: Λ (blue) and K^0_S (green) data vs. standard MC (red) in mass (top), p_T (centre), and y (bottom).

automatically in the p_T and y histograms.

When filling the reconstructed histograms for overlay Monte Carlo, special care has to be taken to disconsider the data tracks, which comprise most of the embedded event. As seen in the data itself, the tracks selection cuts are not enough to ensure a background free mass distribution. During reconstruction the MC tracks are flagged as such by a dedicated parameter, which is then used after reconstruction to identify the simulated tracks, which are the only ones used to compute the spectra.

Figure 5.13 compares the data spectra with the overlay Monte Carlo distributions.

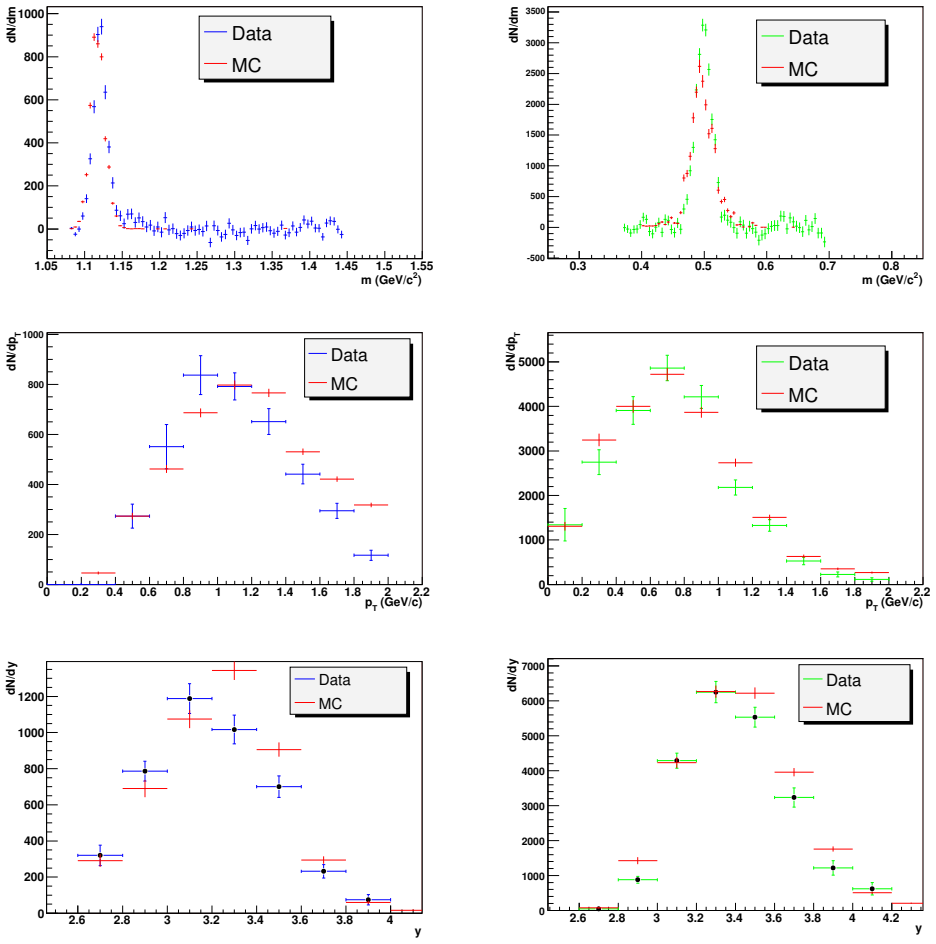


Figure 5.13: Λ (blue) and K^0_S (green) data vs. overlay MC (red) in mass (top), p_T (centre), and y (bottom)

5.3.6 Acceptance Distributions

Figures 5.14 to 5.18 show the acceptance and efficiency functions for both Λ and K^0_S. They are the result of dividing the reconstructed MC spectra by their equivalent generation histograms. The spectra relevant to the produced particles is then obtained in turn by dividing the measured yields per p_T or y bin by the corresponding bin content of the acceptance histograms. This means that the histograms shown in Figs. 5.14 to 5.18 need to be rebinned prior to correcting for acceptance so that they match the dN/dp_T and dN/dy data graphs, which have a much smaller number of data points.

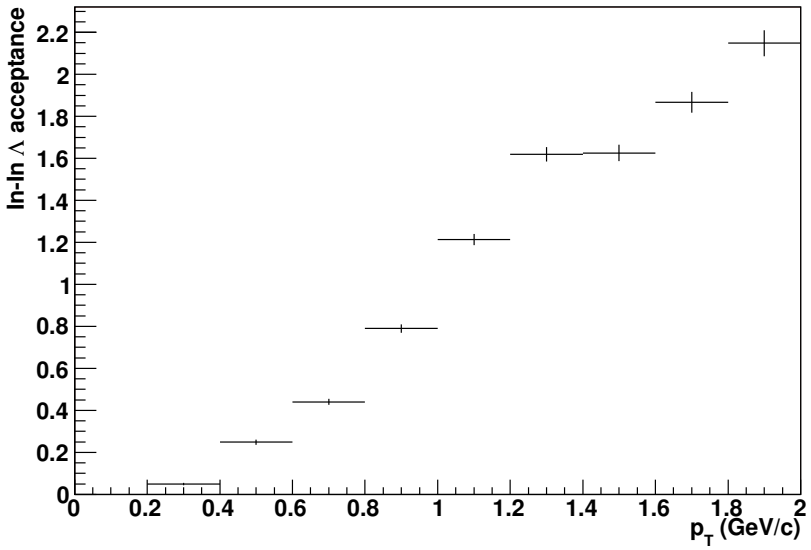


Figure 5.14: In-In Λ acceptance vs. p_T (in arbritrary units).

The errors from the acceptance histograms are propagated through the division. After the correction has been applied, the error in each p_T or y bin is given by

$$dN^2_{acc} = \frac{dN^2}{a^2} + \frac{N^2 da^2}{a^4} \quad ,$$

where dN_{acc} is the error of the particle yield in a given p_T or y bin after acceptance correction, N is the particle yield in that same bin before dividing by the acceptance histogram, and a is the acceptance value to be divided by. Their corresponding errors are dN and da, respectively.

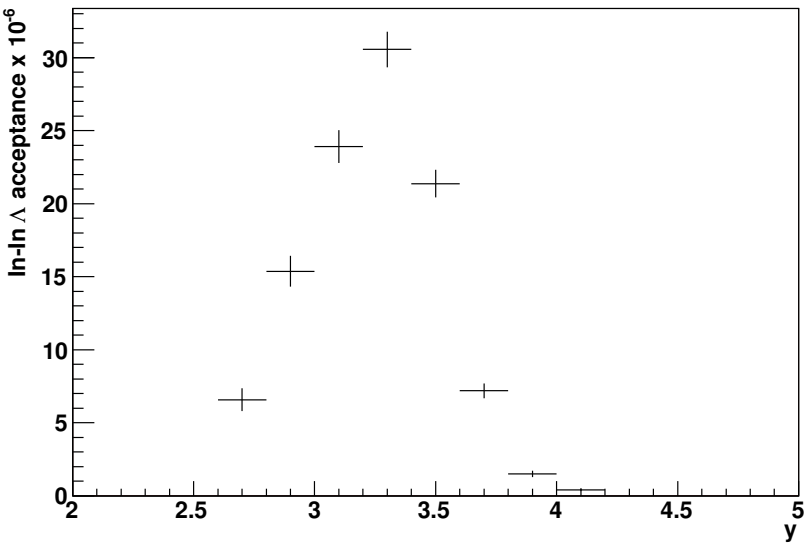


Figure 5.15: In-In Λ acceptance (convoluted with efficiencies) vs. lab rapidity.

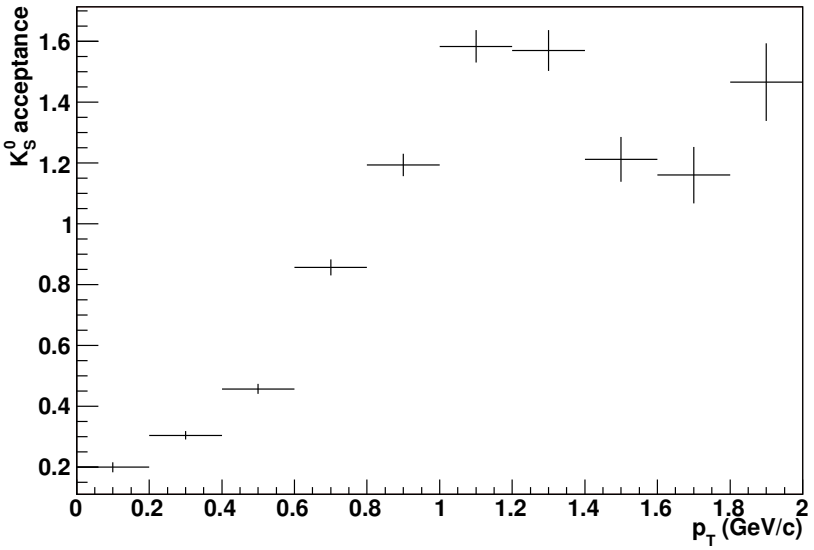


Figure 5.16: In-In K^0_S acceptance vs. p_T (in arbitrary units).

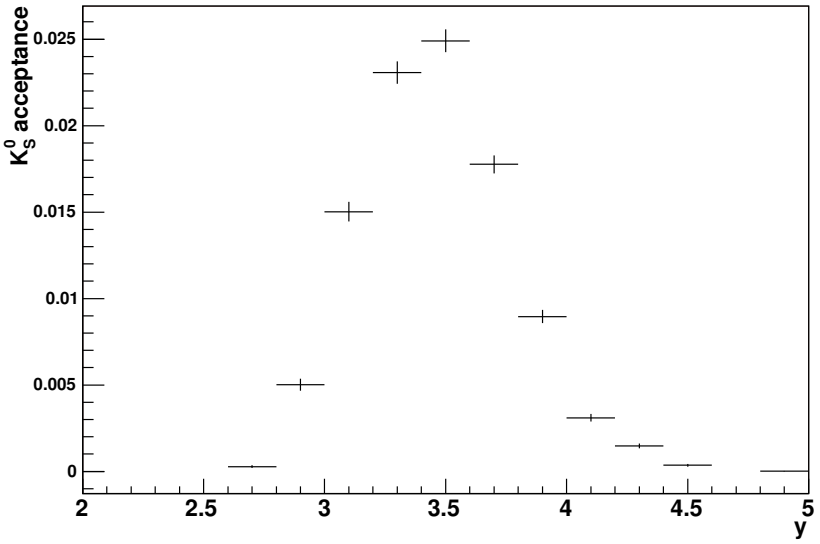


Figure 5.17: In-In K^0_S acceptance (convoluted with efficiencies) vs. lab rapidity.

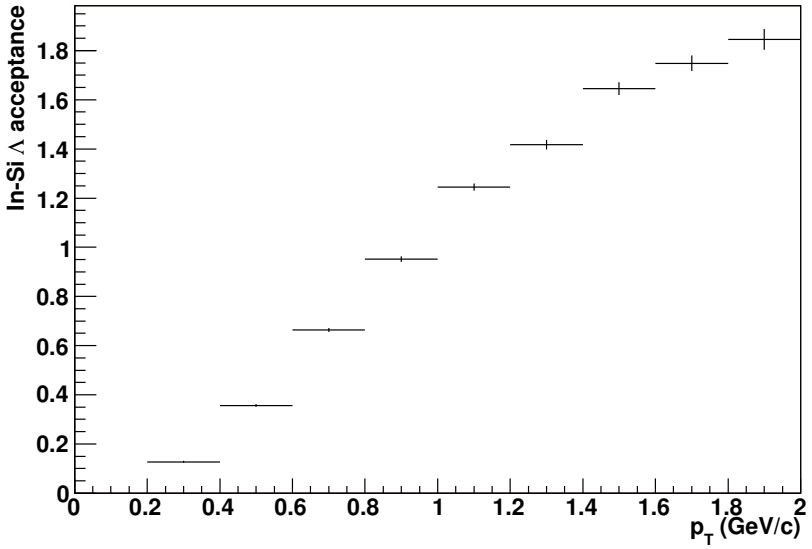


Figure 5.18: In-Si Λ acceptance vs. p_T (in arbitrary units).

5.3.7 Feeddown correction

A fraction of the detected Λs are secondary particles resulting mostly from Ξ^- and Ξ^0 decays, which should be estimated and subtracted from the measured Λs if direct production is to be studied. Previous studies [26, 27] have shown the effect of these corrections to be small and, hence, they have not been applied. Feeddown into K^0_S accounts for an even smaller percentage of the measured particles and can be safely neglected.

5.4 Transverse Momentum Distributions

The p_T spectra are obtained by subtracting the background and extracting the measured yields in each p_T interval. These values are then corrected for acceptance by using the MC simulation methods described above. The corrected spectra are fitted by using the parametrisation

$$\frac{1}{p_T}\frac{dN}{dp_T} \propto \exp(\frac{-m_T}{T})$$

to measure the inverse slope parameter T, where m_T is the transverse mass $m_T = \sqrt{p_T^2 + m^2}$. To estimate systematic uncertainties of the the fitting method, each transverse momentum histogram was fitted using different intervals by excluding one or two points from the high and low p_T regions. Another method used for analysing the systematic errors associated with measuring the p_T slope was to calculate its value for each of the three sets of cuts. The values are summarised in Table 5.1, corresponding to the plots shown in Figs. 5.19 to 5.26.

Particle	**Set 1**	**Set 2**	**Set 3**	**Set 4 (In-Si)**
Λ	248 ± 23	240 ± 21	241 ± 21	207 ± 23
K^0_S	177 ± 19	210 ± 14	207 ± 13	n/a
$\bar{\Lambda}$	n/a	n/a	212 ± 36	n/a

Table 5.1: Inverse slope parameter T for all particles and sets of cuts. All values in MeV. The quoted errors include statistical and systematic errors.

5.5 Transverse Momentum in VT multiplicity bins

The p_T spectra were also studied in three bins of total Vertex Telescope multiplicity. The definition of these multiplicity bins is shown in Table 5.2. The number of VT tracks in an event is a good measure of the number of participant nucleons in the collision [29].

Figures 5.27 and 5.28 overlay the p_T spectra for all three multiplicity bins. The values obtained by fitting to the function described in section 5.4 are summarised in Table 5.3.

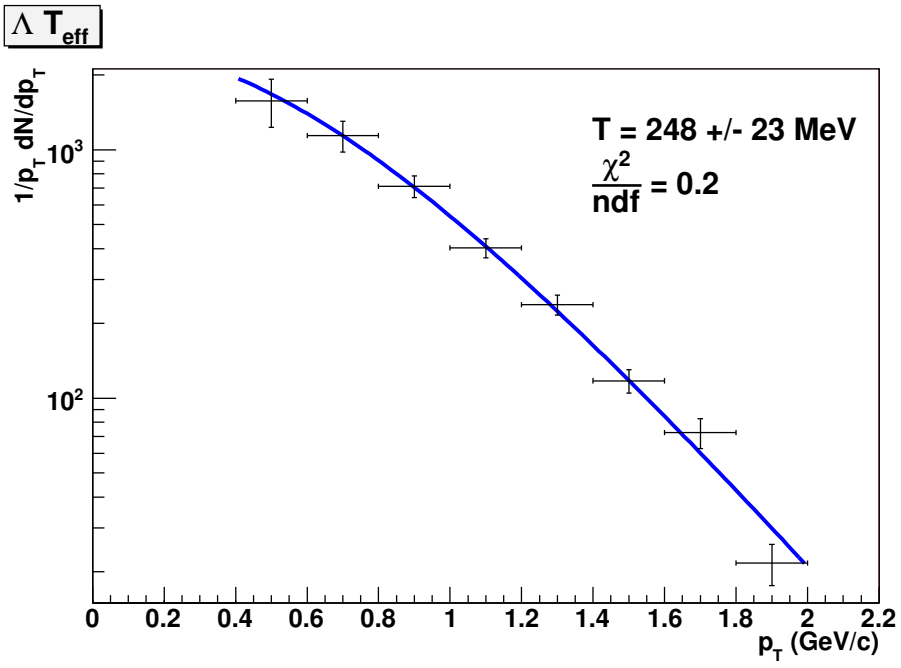


Figure 5.19: In-In Λ p_{T} distribution for the first set of cuts.

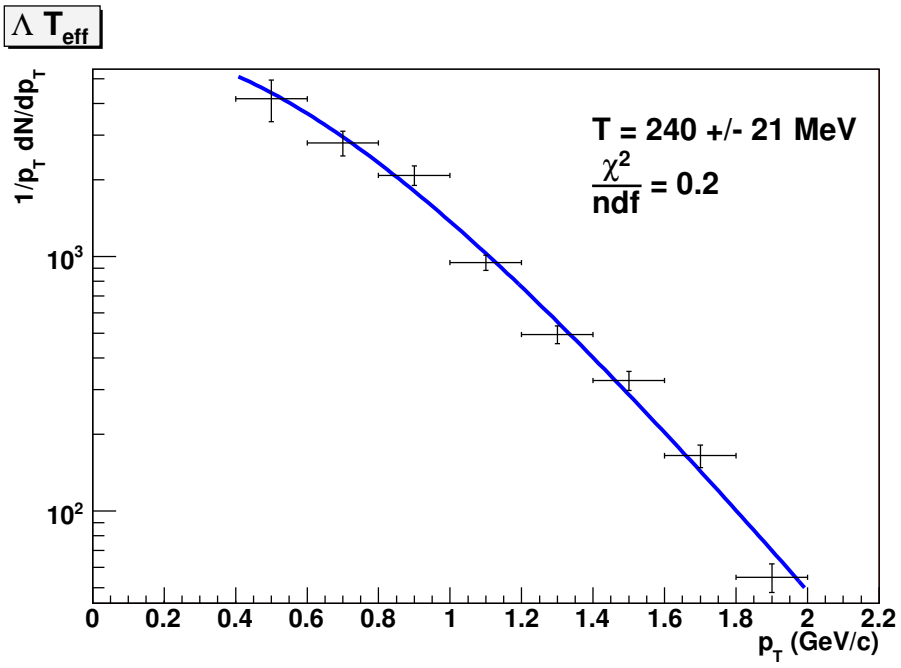


Figure 5.20: In-In Λ p_{T} distribution for the second set of cuts.

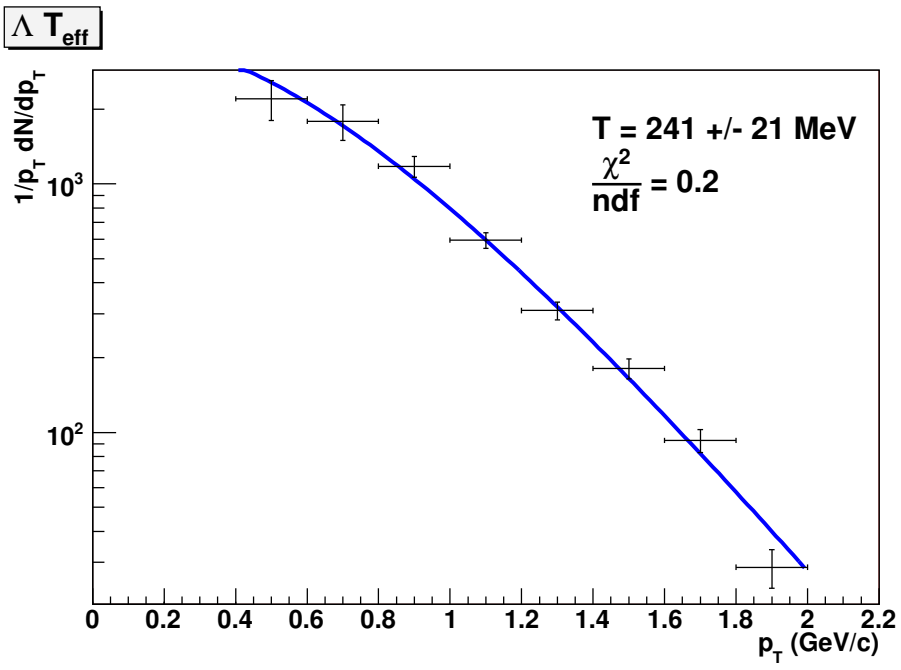


Figure 5.21: In-In Λ p_{T} distribution for the third set of cuts.

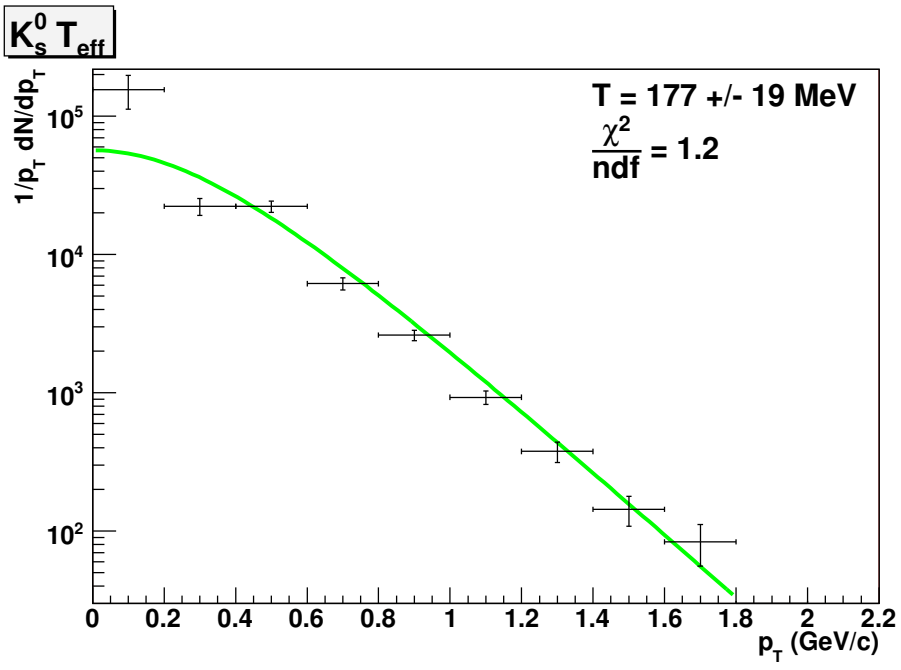


Figure 5.22: In-In $\mathrm{K}^0_{\mathrm{S}}$ p_{T} distribution for the first set of cuts.

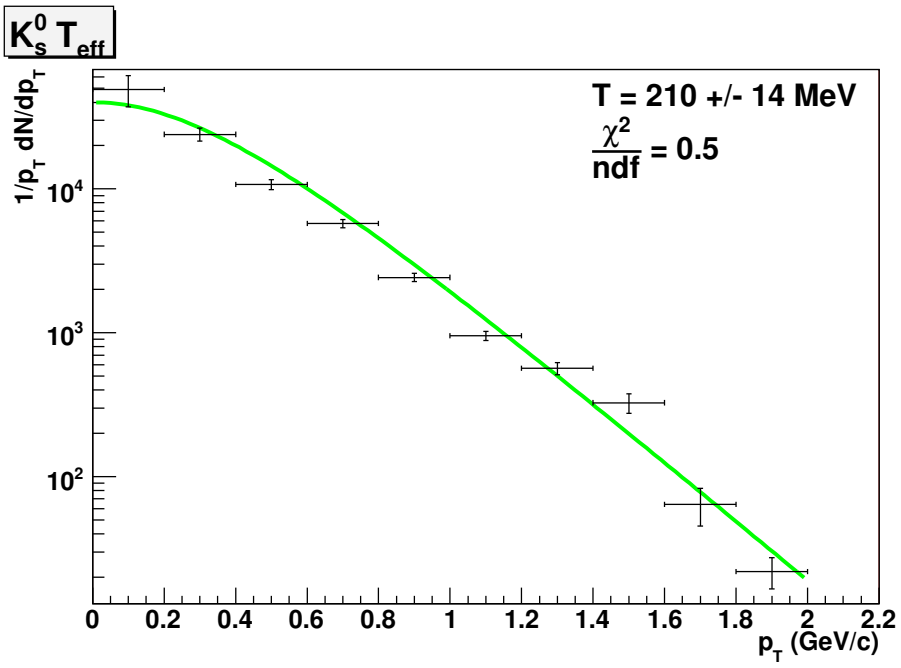


Figure 5.23: In-In K^0_S p_T distribution for the second set of cuts.

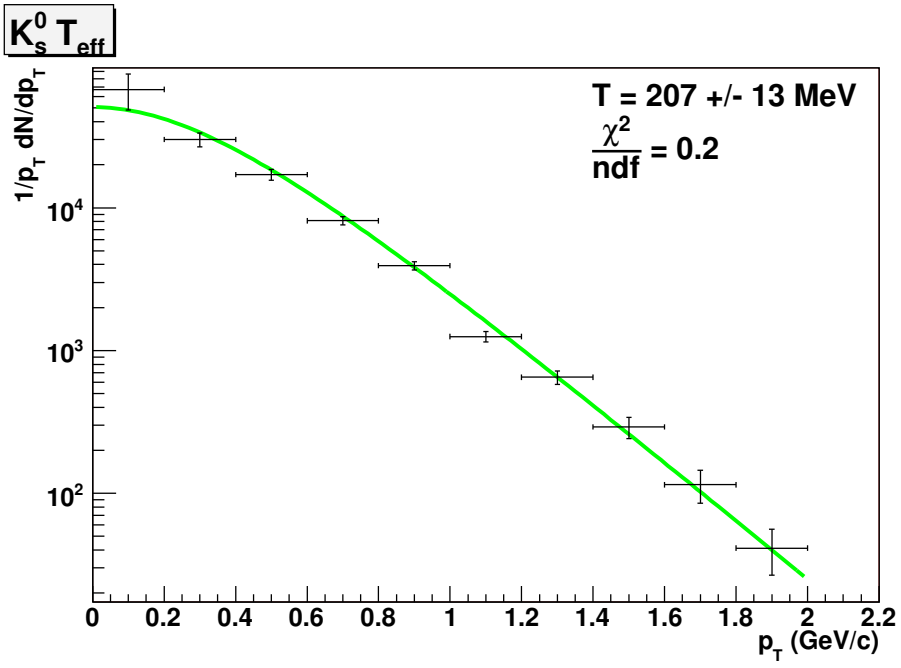


Figure 5.24: In-In K^0_S p_T distribution for the third set of cuts.

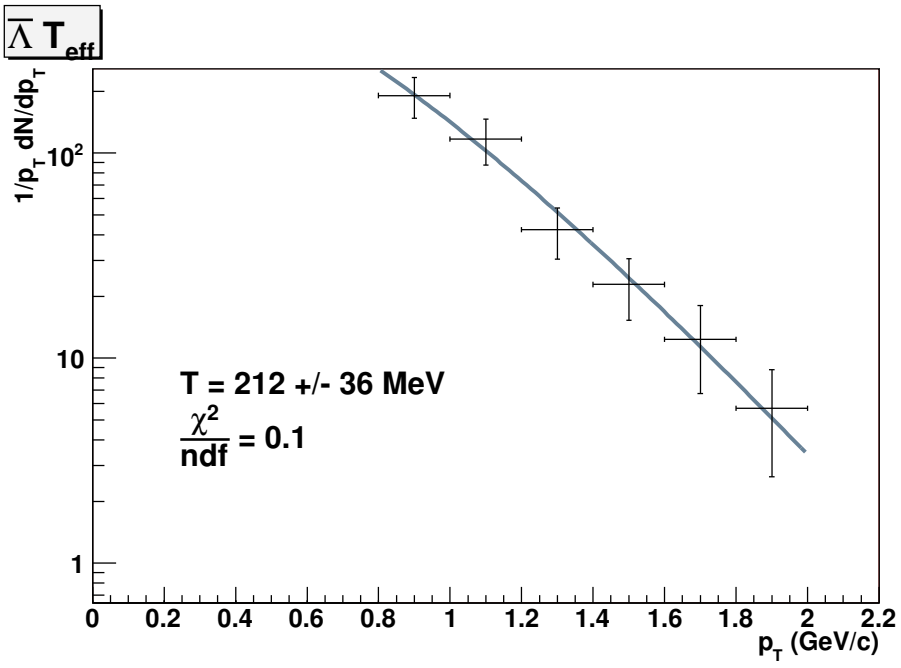


Figure 5.25: In-In $\bar{\Lambda}$ p_{T} distribution.

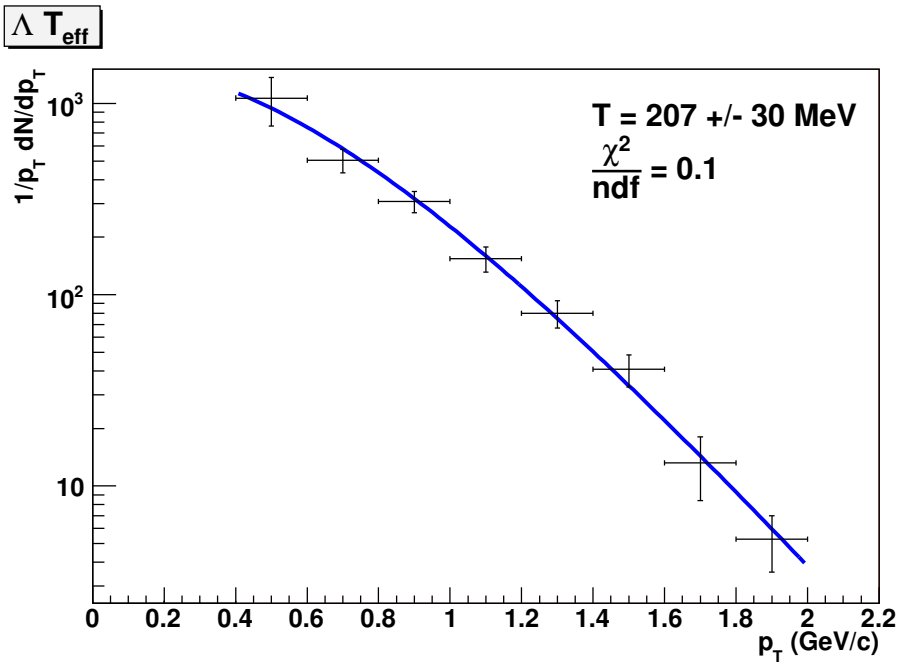


Figure 5.26: In-Si Λ p_{T} distribution.

Multiplicity Bin	VT Tracks
1	< 92
2	[92,159]
3	> 159

Table 5.2: Multiplicity bins.

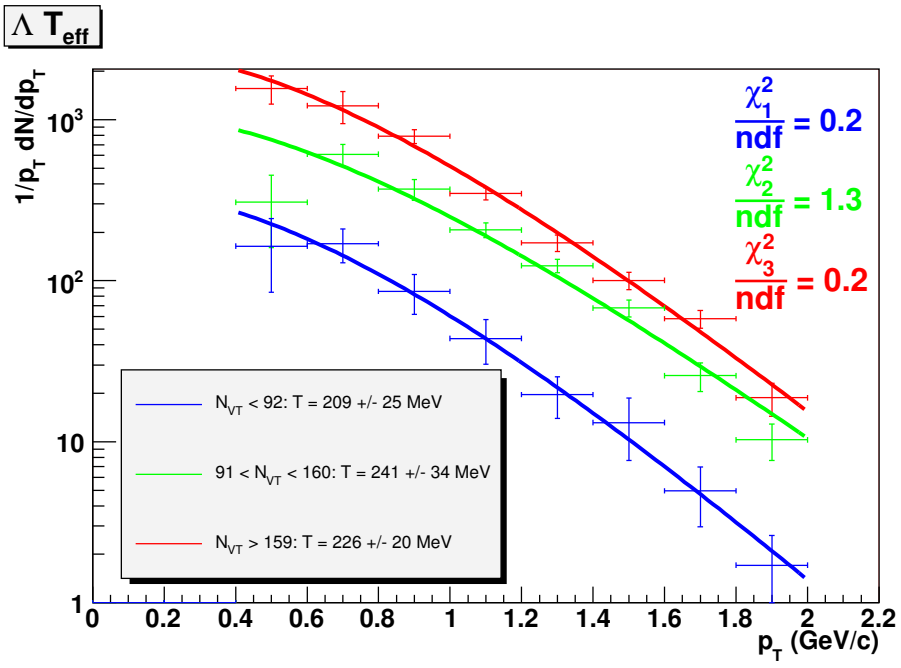


Figure 5.27: In-In Λ p_{T} distribution vs. VT multiplicity. Bin 1 is blue, bin 2 is green and bin 3 is red.

Particle	Bin 1	Bin 2	Bin 3
Λ	209 $\pm$ 25 MeV	241 $\pm$ 34 MeV	226 $\pm$ 20 MeV
$\mathrm{K}^0_{\mathrm{S}}$	210 $\pm$ 13 MeV	210 $\pm$ 13 MeV	216 $\pm$ 16 MeV

Table 5.3: T per multiplicity bin.

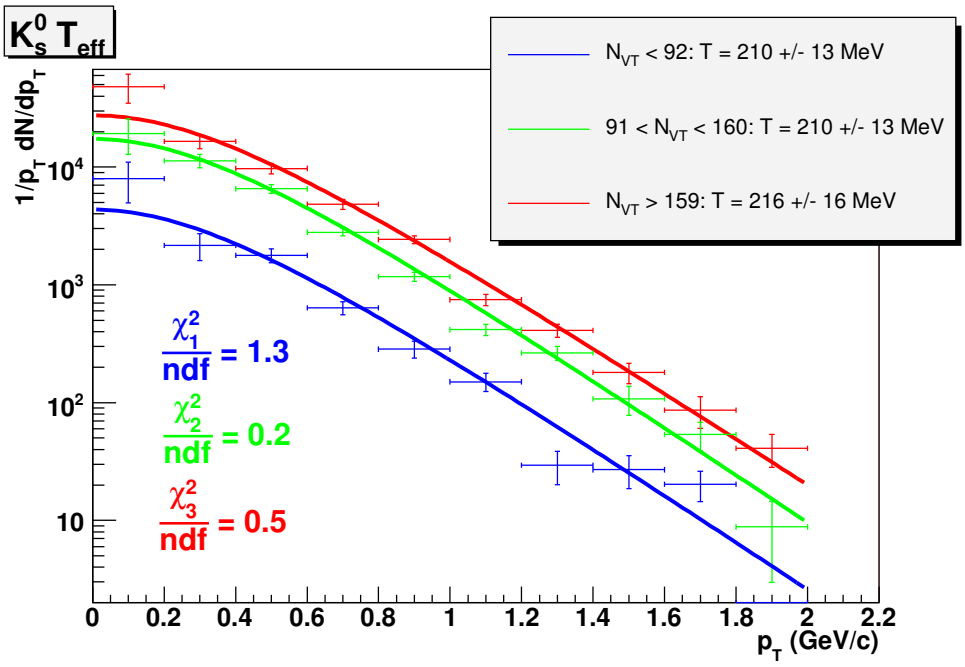


Figure 5.28: In-In K^0_S p_T distribution vs. VT multiplicity. Bin 1 is blue, bin 2 is green and bin 3 is red.

5.6 Rapidity Distributions

The mass spectra were also analysed in intervals of 0.2 units of rapidity in order to obtain the y spectra of the K^0_S and Λ in In-In collisions, and of the Λ in In-Si collisions. The final spectra, after acceptance and efficiency corrections, are shown in Figs. 5.29 to 5.33. The open points represent reflections of the closed, measured points around mid-rapidity, which is made meaningful by the symmetry of the In-In collision system. Only a small region of phase space is shown due to acceptance limitations. The spatial coordinates of the Si targets with respect to the Vertex Telescope means a larger interval in rapidity is probed. Target and beam rapidities lie outside the limits of the presented rapidity plots and are located at roughly 0 and 5.8 units of rapidity.

The K^0_S y spectrum is well described by a gaussian fit, which was found to have a σ of 0.80 $\pm$ 0.10 units of rapidity when using overlay Monte Carlo for acceptance/efficiency corrections and 0.60 $\pm$ 0.05 when using standard Monte Carlo.

The total K^0_S yield was calculated both by integrating the gaussian and by adding the yields in each rapidity bin, resulting in the values of 19.7 $\pm$ 2.6 and 20.0 $\pm$ 1.0 K^0_S/event (after accounting for the branching ratio), respectively, for the overlay Monte Carlo correction. These values are in excellent

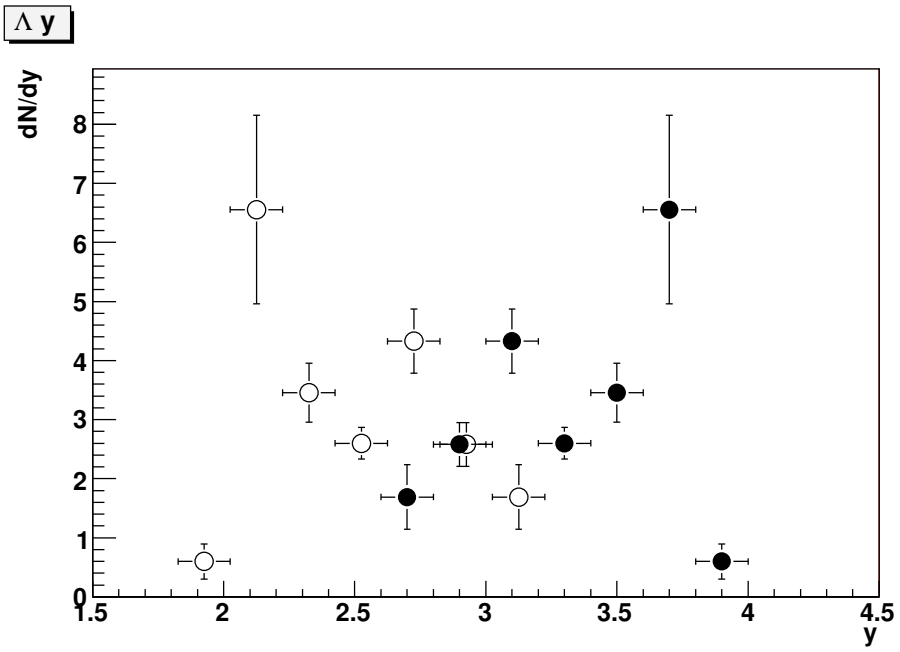


Figure 5.29: In-In Λ rapidity distribution for the second set of cuts. Open points represent the measured (closed) points reflected with respect to mid-rapidity.

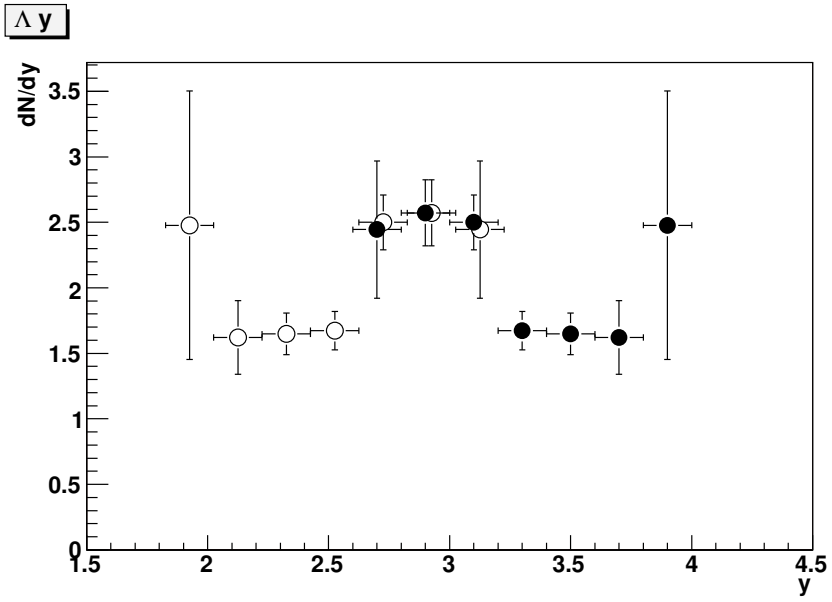


Figure 5.30: In-In Λ rapidity distribution for the third set of cuts. Open points represent the measured (closed) points reflected with respect to mid-rapidity

agreement with each other. The analogous standard MC acceptance correction gives 22.8 ± 1.0 and 18.1 ± 1.7, which agree with the previous values within 2 sigma.

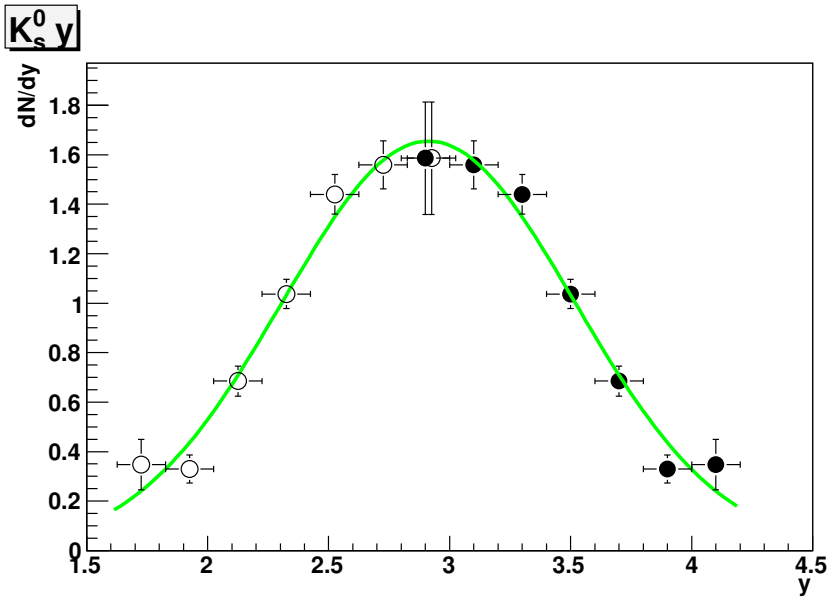


Figure 5.31: In-In K^0_S rapidity distribution corrected with standard MC. Open points represent the measured (closed) points reflected with respect to mid-rapidity. The green curve is a gaussian fit.

The total Λ yield was calculated only by adding the bin contents and dividing by the branching ratio. It was found to be 67.9 ± 5.6 Λ/event in In-In collisions and 47.6 ± 5.0 for In-Si collisions.

For both particles, when summing the individual yields the extrapolation to full phase space was done based on the VENUS generated rapidity spectra. Table 5.4 compiles the particle yield values found by showing weighted averages of the values just discussed.

Particle	**Total Yield**	$\frac{dN}{dy}\|_{\|y^*\|<0.4}$
In-In Λ	67.9 ± 5.6	14.4 ± 0.9
In-In K^0_S Ov	19.9 ± 2.0	6.1 ± 0.6
In-In K^0_S St	21.2 ± 2.6	8.9 ± 0.7
In-Si Λ	47.6 ± 5.0	13.5 ± 1.7

Table 5.4: Particle yields. The third column corresponds to the yield at mid-rapidity ($|y^*| < 0.4$). In the K^0_S yields Ov and St stand for overlay and standard (Monte Carlo), respectively.

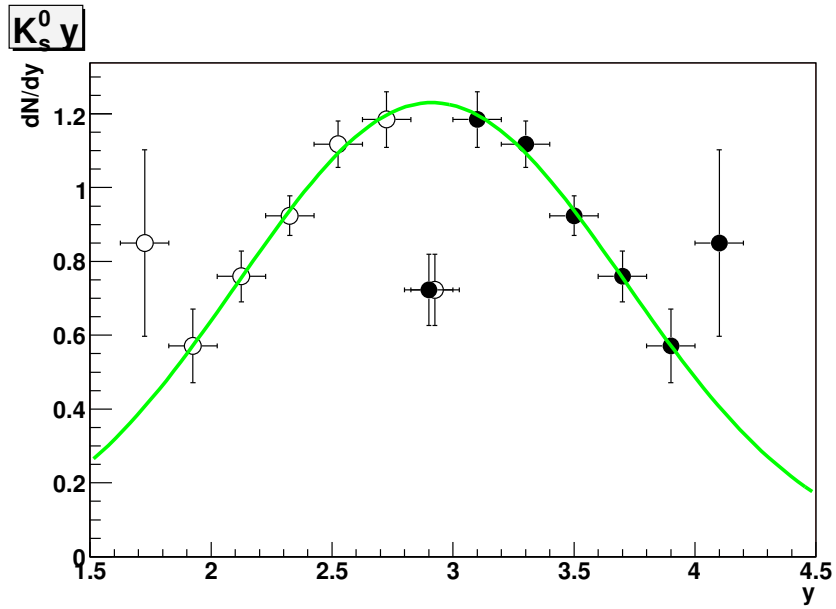


Figure 5.32: In-In K^0_S rapidity distribution corrected with overlay MC. Open points represent the measured (closed) points reflected with respect to mid-rapidity. The first and last data points were excluded from the gaussian fit (the green curve).

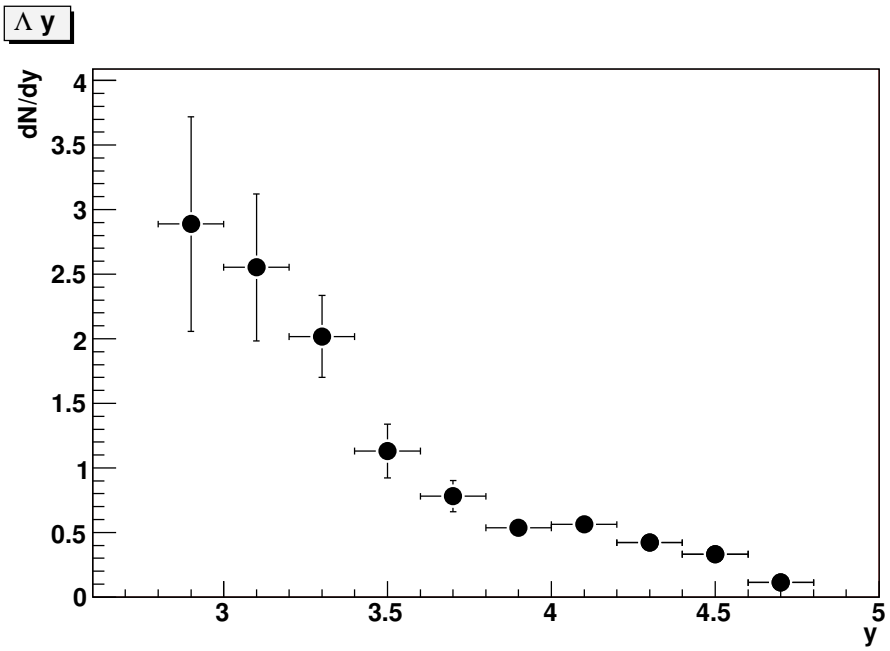


Figure 5.33: In-Si Λ rapidity distribution.

5.7 Particle Ratios

Due to a lack of statistics and a poor signal to background ratio, it was not possible to obtain a rapidity spectrum for the $\bar{\Lambda}$. Its total yield was calculated by subtracting the background in the rapidity interval ranging from 3 to 3.8 units of rapidity, correcting for acceptance and extrapolating to full phase space by using the VENUS generated rapidity spectrum. This results in a $\bar{\Lambda}$ yield $\langle\bar{\Lambda}\rangle = 7.7 \pm 1.2$, making the measured $\bar{\Lambda}/\Lambda$ ratio equal to 0.11 ± 0.02.

The K^0_S yield was taken to be the weighted average of the results from standard and overlay MC corrections, 20.5 ± 2.4. The K^0_S/Λ ratio is then 0.43 ± 0.07.

Chapter 6

Discussion and Conclusion

In this chapter the results presented so far are analysed, interpreted, and compared to previous measurements of strange particle production in heavy ion collisions at 158 GeV per nucleon.

6.1 Transverse Momentum spectra

The motivation behind the study of transverse momentum distributions in high energy collisions is that nearly all of the observed transverse motion is induced by the particle interactions and is thus sensitive to the dynamics of the system. A statistical method to study high energy hadron-hadron collisions was proposed [30] in which particle production is consistent with a thermal description. The inverse slope parameter T in the parametrisation used to fit the measured p_{T} spectra, $\exp(-m_{\mathrm{T}}/T)$, provides information about the temperature of the system and has two main components that are explained below.

It was found [31, 32] that the slope parameters in nucleus-nucleus collisions show marked differences to the ones measured for elementary p-p collisions. In the latter, the measured slopes are constant as a function of particle mass, whereas there is a monotonic increase in the former. Furthermore, for a given particle mass the slopes are higher for heavier collision systems (e.g. Pb-Pb slopes are higher than S-S slopes for each particle).

In hydrodynamics, matter flows, which is to say all particles travel at the same velocity regardless of mass. In classical terms, particles with higher mass have more kinetic energy. The experimental T slope measures not only this collective motion, but also random tranverse motion (thermal energy). The intrinsic freeze-out temperature, which is the temperature at which particles cease to interact with each other, is determined by the thermal motion. In proton-proton collisions rescattering is negligible and the slope parameter should reflect only the freeze-out temperature. As the size of the colliding nuclei gets larger, rescattering becomes more important and

collective motion develops. This interpretation is consistent with the observed data. The relationship between the measured T value, here called the *effective temperature* or T_{eff}, and the particle mass may be described by [32]

$$T_{\text{eff}} = T_0 + m\beta^2 \quad ,$$

where T_0 is the true freeze-out temperature, m is the mass of the particle, and β is the averaged collective flow velocity (in units of c). A similar relationship is discussed in Ref. [33] in the context of a hydrodynamical framework, where $\beta = R_G/t_0$, with R_G being the geometrical radius and t_0 the mean freeze-out time.

Figure 6.1 compares the values of the inverse p_{T} slopes found in this analysis with previous measurements at 158 GeV per nucleon for different collision systems. The solid lines represent the T_{eff} relationship described above using a common T_0 of 175 MeV. It must be noted that all results in Fig. 6.1 are from central collisions except the ones from the present analysis, which is mostly central but includes some peripheral collisions. The results presented here are consistent with the interpretation given above.

The $\bar{\Lambda}$ slope value agrees with the Λ one within errors. Its lower value is closer to what is measured for the K^0_{S} but the magnitude of the errors and closeness of the relevant values mean that it is not possible to state that the $\bar{\Lambda}$ is produced at a different temperature than its anti-particle.

6.1.1 Stability with respect to cuts

The p_{T} spectra and T values are consistent when using different cut values, except for the K^0_{S} in the first set of cuts. This set is the one with the largest systematic uncertainties, however, due to a poorer signal to background ratio. The final T values for both particles are then taken to be $\text{T}_\Lambda = 243 \pm 22$ MeV and $\text{T}_{\text{K}^0_{\text{S}}} = 208 \pm 14$ MeV (excluding the first set of cuts). The quoted errors include systematic components ranging from the background subtraction process up to the fit to the transverse momentum distributions, as described earlier.

In addition to investigating if the choice of track selection cuts introduced any unwanted systematic errors, the choice of MC correction was also studied. Overlay and standard Monte Carlo correction techniques for acceptance and efficiency corrections yielded values that were in remarkable agreement, differing by only 1 or 2 MeV. These differences are negligible in comparison to the errors already incurred in the determination of the temperature values. The only exception was the Λ produced from In-Si collisions where there was a difference of ~ 20 MeV. This was accounted for in the presented error.

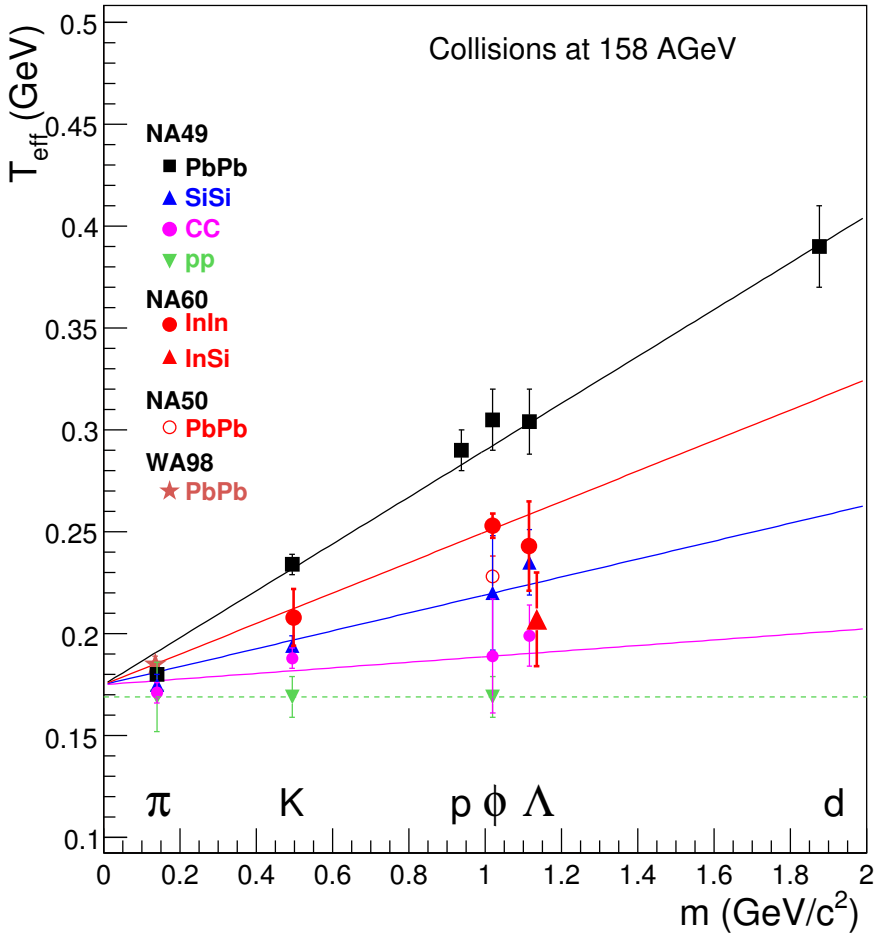


Figure 6.1: Comparison to previous T_{eff} results. The In-Si value has been moved to the right to increase visibility.

6.1.2 Multiplicity dependence of T

The T values measured in three multiplicity bins show no significant variation for the K^0_S with collision centrality. It can therefore be concluded that the same processes are responsible for K^0_S production in both peripheral and central In-In collisions.

The Λ T values show a noticeable deviation in the middle multiplicity bin. The T value calculated there has a much larger systematic error component due to the very first p_T bin. If the fit is done ignoring it, instead of 241 $\pm$ 34 MeV the value becomes 226 $\pm$ 25 MeV. Even when the higher value is used the variation is well within the errors and is therefore considered to not be statistically significant.

6.2 Rapidity spectra

The Λ baryon may be produced by two distinct mechanisms: baryon/anti-baryon pair production or associated production. The latter mechanism consists of swapping a u or d quark (depending on whether the nucleon is a proton or a neutron) for an s quark since the Λ shares a ud quark pair with either nucleon. It is then expected that the $\Lambda - \bar{\Lambda}$ rapidity spectrum reflects the net baryon number distribution.

Both proton and $p - \bar{p}$ rapidity distributions have been measured [32, 34, 35] for nucleus-nucleus collisions, showing a concave shape with a gentle slope, with a dip at mid-rapidity (shown in Fig. 6.2), while the p-p collisions spectra show two distinct peaks near beam and target rapidities. This indicates that nuclear stopping increases with increasing collision system size, inducing a rapidity shift towards mid-rapidity, with the distributions becoming flatter.

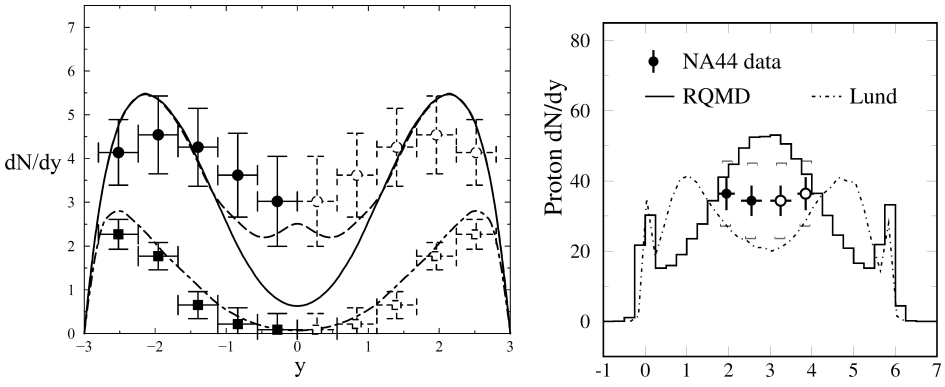


Figure 6.2: Net proton rapidity distribution taken from Ref. [34] (left), and proton rapidity spectrum from Ref. [32] (right). In the left plot the lower points correspond to peripheral collisions and the higher points to central ones.

The Λ's distribution in rapidity should then resemble the sum of the net proton rapidity spectrum with the contribution from the baryon/anti-baryon pair production (which is peaked at mid-rapidity). This would mean an almost flat distribution, with maybe a slightly higher yield at $y^* = 0$, depending on how large the baryon/anti-baryon pair contribution is. The data from the present analysis seems to support this hypothesis and is compared to previous results in Figs. 6.3 and 6.4.

The Λ rapidity spectrum obtained from In-Si collisions shown in Fig. 5.33 probes a different region in rapidity and lends further credence to the interpretation just described.

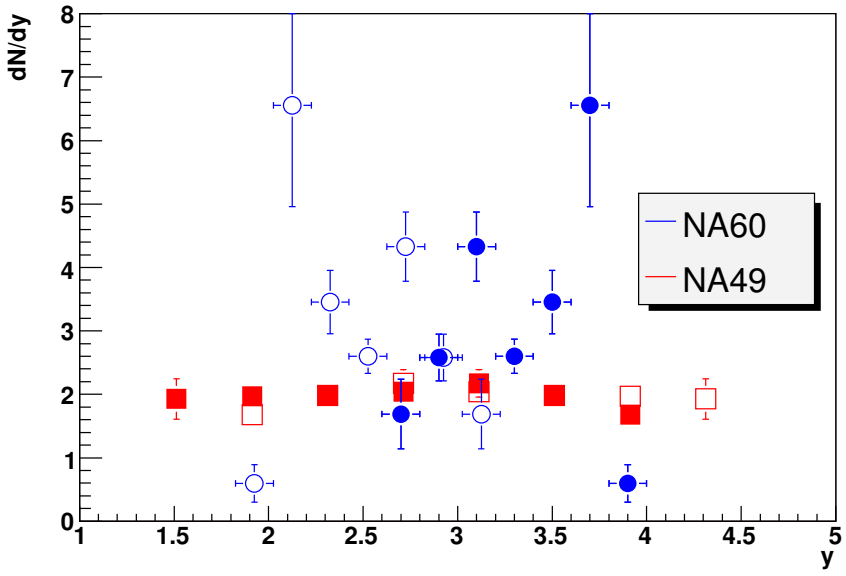


Figure 6.3: Λ rapidity spectrum (second set of cuts) compared to previous results. Open points represent the measured closed points reflected with respect to mid-rapidity. The NA49 have been scaled down to facilitate the comparison of both shapes

The K^0_S distribution, on the other hand, is expected to be peaked at mid-rapidity and is well represented by a gaussian curve. This has been confirmed in numerous measurements by several experiments, and seems to also be the case in the present analysis. There is a large systematic uncertainty due to the acceptance and efficiency corrections because the overlay and standard MC correction techniques result in different final rapidity spectra which have gaussian σs of 0.80 ± 0.10 and 0.60 ± 0.05 units of rapidity, respectively. This represents a systematic error of 0.09, making the measured value of $\sigma = 0.67 \pm 0.12$ rapidity units. This value is compared to previous measurements, made for Pb-Pb collisions, in Table 6.1.

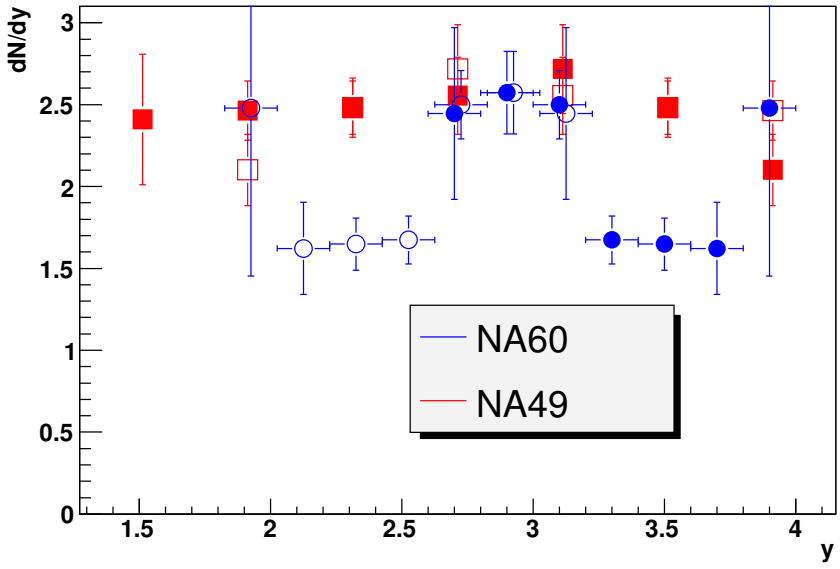


Figure 6.4: Λ rapidity spectrum (third set of cuts) compared to previous results. Open points represent the measured closed point with respect to mid-rapidity. The NA49 have been scaled down to facilitate the comparison of both shapes

Experiment	$\sigma_{K^0_S}$
NA60	0.67 $\pm$ 0.12
NA49	1.16 $\pm$ 0.11
NA57	0.65 $\pm$ 0.13

Table 6.1: K^0_S rapidity σ compared to previous measurements [27, 36]. The NA49 value is an average of two separate measurements [27]. The NA57 value is an average of the values cited for five centrality classes in Ref. [36].

The result presented here is in good agreement with what is measured by NA57 [36] and smaller than the NA49 [27] result.

6.3 Particle Ratios

The value obtained for the $\bar{\Lambda}/\Lambda$ ratio presented in Section 5.7 came at the cost of an undetermined systematic uncertainty in the calculation of the $\bar{\Lambda}$ yield. The Monte Carlo based extrapolations to full phase space could also have introduced distortions in the final number. The measured ratio of 0.11 ± 0.02 is compared to results from other experiments in Table 6.2 and, despite these caveats, seems to agree with the numbers from Pb-Pb collisions at the same energy.

Experiment	**Collision**	$\bar{\Lambda}/\Lambda$	**Reference**
NA60	(In-In)	0.11 ± 0.02	n/a
NA49	(Pb-Pb)	0.09 ± 0.02	[26]
NA49	(Pb-Pb)	0.105 ± 0.013	[27]
NA57	(Pb-Pb)	0.15 ± 0.01	[36]
WA97	(Pb-Pb)	0.133 ± 0.007	[10]
STAR	(Au-Au)	0.74 ± 0.04	[37]

Table 6.2: $\bar{\Lambda}/\Lambda$ ratio compared to previous measurements. All of the results are from collisions at 158 GeV/nucleon except for the STAR value, which is at $\sqrt{s_{NN}} = 130$ GeV.

The K^0_S/Λ ratio is smaller than previous Pb-Pb measurements. NA49 measures 1.00 ± 0.13 [27], while NA57 measures 1.51 ± 0.22 [36] at one unit of rapidity centered at $y^* = 0$, where this value is expected to be higher because of the peaked nature of the K^0_S rapidity distribution. These values mean that the presented value of 0.43 ± 0.07 is roughly a factor of 2 smaller than what has been measured in Pb-Pb collisions at 158 GeV per nucleon.

Due to the Λ hyperon's sole strange quark, it can be argued that in Pb-Pb collisions the K^0_S meson, with its contributions from both K^0 ($\bar{s}d$) and $\bar{K}^0$ ($s\bar{d}$), is more easily produced than in In-In collisions with respect to the Λ, explaining the lower number found here.

6.4 Conclusion

Measurements of strange particle production at the NA60 experiment in In-In and In-Si collisions at 158 GeV per nucleon have been presented. These measurements include the p_T spectra, which are well described by a $p_T \exp(-m_T/T)$ parametrisation and were measured to be $T_\Lambda = 245 \pm 25$ MeV, $T_{K^0_S} = 210 \pm 15$ MeV, $T_{\bar{\Lambda}} = 212 \pm 36$ MeV for In-In collisions and

$T_\Lambda = 207 \pm 23$ MeV for In-Si collisions. These results are consistent with a scenario in which collective transverse flow modifies the observed temperature in a mass dependent way. The Λ and K^0_S transverse momenta distributions were also analysed in three different centrality bins. Except for a systematic uncertainty in the determination of the Λ T value in the second VT multiplicity bin, the variations were small. They suggest a slight increase in temperature with collision centrality but the errors prevent any definite conclusion in that regard.

Rapidity spectra were also presented for the In-In (Λ and K^0_S) and In-Si (Λ) systems. The K^0_S distribution was found to be aptly described by a gaussian despite the systematic error induced by the acceptance and efficiency corrections. Further tuning of the Monte Carlo techniques used could result in more accurate measurements.

The Λ rapidity distributions seem to agree with the accepted picture of Λ production, being mostly flat in the mid-rapidity region.

Finally, the $\bar{\Lambda}/\Lambda$ ratio was calculated and found to be 0.11 ± 0.02, which is in very good agreement with the values found by other experiments for central Pb-Pb collisions at the same energy. This was not the case for the K^0_S/Λ ratio, which was found to be a factor of 2 lower than in Pb-Pb collisions.

Bibliography

[1] J. Cleymans *et al.*, Phys. Rep. 130 (1986) 217.

[2] F. Karsch, *Lattice QCD at high temperature and density*, Lect. Notes Phys. 583 (2002) 209; hep-lat/0106019.

[3] M.C. Abreu *et al.* (NA50 Coll.), *Evidence for Deconfinement of Quarks and Gluons from the J/ψ Suppression Pattern measured in Pb-Pb Collisions at the CERN-SPS*, Phys. Lett. B477 (2000) 28.

[4] M.C. Abreu *et al.* (NA38 and NA50 Coll.), *Dimuon and Charm Production in Nucleus-Nucleus Collisions at the CERN SPS*, Eur. Phys. J. C14 (2000) 443.

[5] R. Rapp and E. Shuryak, Phys. Lett. B473 (2000) 13.

[6] L. Capelli, *Etude des dimuons de la région des masses intermédiaires produits dans les collisions d'ions lourds auprès du SPS du CERN*, PhD thesis, 2001

[7] J.W. Qiu and X.F. Zhang, *Strong nuclear enhancement in intermediate mass Drell-Yan production*, Phys. Lett. B525 (2002) 265.

[8] G. Agakichiev *et al.* (CERES Coll.), *Low-mass e^+e^- Pair Production in 158 A GeV Pb-Au Collisions at the CERN SPS, its Dependence on Multiplicity and Transverse Momentum*, Phys. Lett. B422 (1998) 405.

[9] G. Agakichiev *et al.* (CERES Coll.), *Systematic Study of Low-Mass Electron Pair Production in p-Be and p-Au Collisions at 450 GeV/c*, Eur. Phys. J. C4 (1998) 231.

[10] R.A. Fini *et al.* (WA97 Coll.), J. Phys. G: Nucl. Phys. 27 (2001) 375.

[11] *Study of Dimuon and Charm Production with Proton and Heavy Ion Beams at the CERN SPS*, NA60 Letter of Intent, May 1999; CERN/SPSC 1999-015 and SPSC/M 679;
`http:/na60.cern.ch/www/experiment/spsc/loi/loi10.ps`

[12] *Study of Prompt Dimuon and Charm Production with Proton and Heavy Ion Beams at the CERN SPS*, NA60 proposal, March 2000, `http://na60.cern.ch/www/experiment/spsc/proposal/p.ps.gz`

[13] R. Veenhof and H.Wöhri, *The gas mixture of the NA60 muon chambers*, NA60 Internal Note, May 2001, `http://na60.cern.ch/www/experiment/notes /2001_1/gasnote.pdf`

[14] K. Wyllie et al., *A Pixel Readout Chip for Tracking at ALICE and Particle Identification at LHCb*, Proceedings of the Fifth Workshop on Electronics for LHC Experiments, Snowmass, Colorado, 1999

[15] J.J. van Hunen *et al.*, *Irradiation and SPS Beam Tests of the ALICE1LHCb Pixel Chip*, CERN-ALI-2001-015, 2001

[16] M. Keil *et al.*, *The NA60 Silicon Pixel Telescope*, Proceedings of the 12^{th} International Workshop on Vertex Detection (VERTEX 2003), Nucl. Instrum. Meth. A549 (2005) 20.

[17] J. Buytaert *et al.*, *The NA60 silicon tracking telescope for proton running*, NA60 Internal Note 2001-5, July 2001.

[18] K. Banicz, F. Bergsma, H. Wöhri, *The PT7 Magnet*, NA60 Internal Note 2001-2, `http://na60.cern.ch/www/experiment/notes/note_2001_2.pdf`

[19] *ALICE DATE User's Guide*, ALICE Internal Note/DAQ ALICE-INT-2000-31 v.2, January 2001

[20] R. Brun, F. Rademakers, *ROOT Reference Guide*, June 2000, `http://root.cern.ch/root/Reference.html`

[21] V. Bartsch et al., *Lorentz Angle Measurements in Silicon Detectors*, Nucl. Instr. Meth. A478 (2002) 330.

[22] A. David, *Prompt and displaced muon production in Indium-Indium collisions as measured by the NA60 experiment*, PhD Thesis, IST, Lisbon, February 2006.

[23] G. Agakichiev *et al.*, *A new robust fitting algorithm for vertex reconstruction in the CERES experiment*, Nucl. Instrum. Meth. A394 (1997) 225.

[24] R. Shahoyan, *Combinatorial Background in the NA60 Muon Spectrometer*, NA60 internal note, June 2004

[25] F. Antinori *et al.* (NA57 Coll.), *Study of the transverse mass spectra of strange particles in Pb-Pb collisions at 158 A GeV/c*, J. Phys. G: Nucl. Part. Phys. 30 (2004) 823.

[26] T. Anticic *et al.* (NA49 Coll.), Λ *and* $\bar{\Lambda}$ *Production in Central Pb-Pb Collisions at 40, 80, and 158 A·GeV*, Phys. Rev. Lett. 93 (2004) 022302.

[27] L. Barnby, *Measurements of* Λ, $\bar{\Lambda}$ *and* K^0_S *from Pb-Pb Collisions at 158 GeV per nucleon in a Large Acceptance Experiment*, PhD Thesis, June 1999

[28] R. Brun *et al.*, *GEANT User's Guide*

[29] S. Damjanovic, private communication.

[30] E. Fermi, *High Energy Nuclear Events*, Progress of Theoretical Physics, Vol. 5, No. 4, 570.

[31] I. Bearden et al. (NA44 Coll.), *Collective expansion in high energy heavy ion collisions*, Phys. Rev. Lett. 78 (1997) 2080.

[32] N. Xu *et al.* (NA44 Coll.), *Hadron Distributions - Recent Results from the CERN Experiment NA44*, Nucl. Phys. A 610 (1996) 175.

[33] T. Csörgő and B. Lörstad, *Bose-Einstein Correlations for Three-Dimensionally Expanding Cylindrically Symmetric Finite Systems*, Phys. Rev. C54 (1996) 1390.

[34] J.A. Casado, *New explanation of the strange baryon rapidity distributions in ultra-relativistic nucleus-nucleus collisions*, Nucl. Phys. A651 (1999) 93.

[35] T. Alber *et al.*, *Charged Particle Production in Proton-, Deuteron-, Oxygen- and Sulphur-Nucleus Collisions at 200 GeV per Nucleon*, Eur. Phys. J. C2 (1998) 643.

[36] F. Antinori *et al.* (NA57 Coll.), *Rapidity distributions around mid-rapidity of strange particles in Pb-Pb collisions at 158 A GeV/c*, J. Phys. G 31: Nucl. Part. Phys. (2005) 1345.

[37] C. Adler *et al.* (STAR Coll.), *Mid-rapidity* Λ *and* $\bar{\Lambda}$ *Production in Au+Au Collisions at* $\sqrt{s_{NN}} = 130$ *GeV*, Phys. Rev. Lett. 89 (2002) 092301.