TESIS DE DOCTORADO

SIMULATION AND RECONSTRUCTION ALGORITHMS FOR A COMMERCIAL MUON TOMOGRAPHY SYSTEM

José Javier Cuenca García

ESCUELA DE DOCTORADO INTERNACIONAL

PROGRAMA DE DOCTORADO EN FÍSICA NUCLEAR Y DE PARTÍCULAS

SANTIAGO DE COMPOSTELA

2018

DECLARACIÓN DEL AUTOR DE LA TESIS
Simulation and reconstruction algorithms for a commercial muon tomography system

D. José Javier Cuenca García.

Presento mi tesis, siguiendo el procedimiento adecuado al Reglamento, y declaro que:

1) *La tesis abarca los resultados de la elaboración de mi trabajo.*
2) *En su caso, en la tesis se hace referencia a las colaboraciones que tuvo este trabajo.*
3) *La tesis es la versión definitiva presentada para su defensa y coincide con la versión enviada en formato electrónico.*
4) *Confirmo que la tesis no incurre en ningún tipo de plagio de otros autores ni de trabajos presentados por mí para la obtención de otros títulos.*

En Santiago de Compostela, 15 de octubre de 2018

Fdo. José Javier
Cuenca García

AUTORIZACIÓN DEL DIRECTOR / TUTOR DE LA TESIS

Simulation and reconstruction algorithms for a commercial muon tomography system

D. Juan Antonio Garzón Heydt
D. Pablo Cabanelas Eiras

INFORMAN:

Que la presente tesis, corresponde con el trabajo realizado por D. José Javier Cuenca García, bajo mi dirección, y autorizo su presentación, considerando que reúne los requisitos exigidos en el Reglamento de Estudios de Doctorado de la USC, y que como director de ésta no incurre en las causas de abstención establecidas en Ley 40/2015.

En Santiago de Compostela, 15 de octubre de 2018

Fdo. Juan Antonio
Garzón Heydt

Fdo. Pablo Cabanelas Eiras

I would like to dedicate this thesis to G. Kornakov

Agradecimientos

Es de gente bien nacida
el agradecer el beneficio que recibe

Don Quijote de la Mancha
Miguel de Cervantes

Esta tesis no habría sido posible sin la dirección del Prof. Juan Antonio Garzón y el Dr. Pablo Cabanelas. A ellos van los más importantes agradecimientos. También mi agradecimiento a todo el equipo de la empresa HIDRONAV por haber financiado este trabajo y por haber estado siempre dispuestos a resolver dudas y preguntas.

Tampoco esta tesis hubiera sido posible sin Alberto Blanco y todo el equipo de la Universidad de Coimbra por el diseño y construcción de las RPCs que hemos tomado como detector de referencia. Gracias a las conversaciones con él he podido entender mucho mejor el funcionamiento de los detectores.

Agradecer al Dr. Georgy Kornakov todas las conversaciones, discusiones, consejos, lloros y quejas que ha tenido que soportar por mi parte. También por hacer de guía con Root y por ser la persona que me hizo continuar con esto en los momentos más duros.

Muchas gracias al Prof. Héctor Álvarez Pol por ser la pieza clave en las simulaciones y por haber hecho el trabajo mucho más llevadero. Bueno, y por ser como es.

Por supuesto agradecer a Marcos Seco todas las respuestas a las preguntas sobre programación, leaks de memoria, expresiones regulares, depurado de códigos y demás temas relacionados con la informática. Es un privilegio contar con alguien que habla lenguaje máquina y trata a esos seres de igual a igual.

Dar las gracias al Dr. Diego González-Díaz por haber sido de apoyo en un momento duro y por haberme dejado parasitar en su casa para así poder terminar de manera tranquila este trabajo.

Mil gracias también a Manuel Caamaño y Bea Fernández. Aunque bueno, lo de Manu no tenga mucho arreglo... así, en general.

Y por supuesto no olvidar a Marisol, Elena Van y Juanjo, Damián, Irma ni tampoco a nuestro técnico Antonio Pazos.

Muchas gracias a todas y todos por haber hecho esto posible.

Table of contents

List of figures

List of tables

Introduction

The cosmic radiation that arrives to the Earth surface is the result of multiple collisions of primary cosmic rays with the atmospheric components at a mean altitude of approximately 15 km. This primary radiation is mainly composed by protons and hydrogen nuclei, together with a very low fraction of heavy elements. The origin of these heavy elements is not well known, but they might be produced in astrophysical processes at the final stages of the stellar nucleosynthesis. After these collisions, the produced secondary cosmic radiation is composed, most importantly, by protons, kaons, pions, and electrons. The pions and kaons are not stable and they decay. Muons are produced as a product of these decay reactions. The muons travel along the atmosphere and can be detected at ground level.

The study of this secondary radiation could provide the key of understanding the structure of extragalactic objects and atmospheric phenomena. Observatories such as AUGER [3] are devoted to the study of the properties of cosmic rays for years. Since this secondary radiation is produced by collisions at the atmosphere, it seems to be reasonable to study if the muon flux is affected by atmospheric conditions such as temperature and pressure. These effects are currently being studied in several PhD. theses at the University of Santiago de Compostela [5, 6]. The conclusions of these studies could provide information about the layer structure of the terrestrial atmosphere.

The experiments of E.P. George in the 1950s can be considered as the starting point in the study of matter by the use of cosmic muons [35]. In that particular work, the cosmic ray muons were used to estimate the depth of overburden of a tunnel in an Australian mine. George measured the flux of cosmic rays inside and outside the tunnel and using the ratio of both signals inferred the amount of rock above the tunnel. With this experiment, the attenuation properties of materials were first use to perform a radiography of large objects.

After this experiment, in 1970, Luis Alvarez used the same technique as George and explored the Giza pyramids to search hidden chambers inside them [17]. These experiments became very important in understanding and develop the muon radiography techniques and in subsequent years were extended to other fields of science.

It is possible to obtain three dimensional images of objects by the use of cosmic ray muons. When these muons pass through a material medium they are affected by the Coulomb scattering. This process causes a deviation in the muon trajectory and depends on the nature of the material. The measurement of the deflection angle of muons and other particles can be used to obtain information about the properties of the material and even reconstruct their shapes. These images can be obtained using *muon tomography* or *muon radiography*.

Muon tomography and muon radiography have become great imaging techniques and they are applied in several research fields. A recent experiment tried to improve the first Alvarez investigation in the Egypt's pyramids [42]. In that work, using muon radiography, a secret chamber inside great pyramid was found. The muon tomography technique is also used to produce images of nuclear reactors, as the case of Fukushima's nuclear plant [38]. Also, it is used to determine the volcanic activity [39] or in homeland security applications [12].

The improvement in security on the borders against illegal traffic of radioactive materials has become a homeland security priority. This has motivated that many countries are interested on the development of new technologies that allow, with a high level of confidence, the detection of that kind of substances. It is believed that one of the most likely ways of entrance of those materials is by the use of containers transported by ships. This is one of the main reasons why most of the current research works are devoted to the development of facilities that are able to analyse those containers directly in the docks. Due to the huge number of containers that can be transported in a dock every day, they should be able to analyse a container in a short period of time. A little change in this processing time can cause a high delay time in traffic, and consequently the process can cause a non-desired chaotic situation.

A first approach to the problem would be to take a radiography of the container by the use of X-rays. Taking into account that not only the radioactive elements, but also their shielding material have high charge number, the electromagnetic radiation would show a high deviation inside the container. Therefore, this method could be a very good candidate.

However, the containers have typically a large volume and this method would be extremely inconvenient for several reasons. In one hand, it is necessary to develop a radiography system based on X-rays, high energy gammas, or even electrons by the use of an accelerator and cover a large surface. In the other hand, the price of that infrastructure would be very expensive.

An alternative to the X-rays would be the natural cosmic radiation. The cosmic rays are a permanent, free, and harmless source of high energy particles. The cosmic radiation at sea level is mostly composed by muons, electrons, and gammas. Due to the low intensity

of cosmic electrons and gammas, to perform a good quality radiography of the container would require high exposure times. This can cause a non-desired problem in the traffic of containers at the facility.

It seems to be obvious that the key of the problem is on the high energetic cosmic muons that are able to arrive to the Earth surface. The probability of these muons to be absorbed by a high Z material is almost zero. Therefore, the study of the Coulomb scattering properties of the muons could give information about the nature of the target material.

The other key part of the tomography system is the detection of particles. The Resistive Plate Chambers (RPCs) are gaseous detectors that are widely used for detecting muons. They provide good spatial and time resolutions and therefore they are well suited to perform tracking of muons. The RPCs are easy to set-up and have a good ratio price per channel, compared to other detectors.

The goal of this thesis is to develop the simulation software, the mathematical algorithms, and the analysis tools needed to construct a scanner formed by several RPC detectors. This system is based on the cosmic muon tomography to produce images of heavy materials inside a container.

The thesis is divided in three main blocks. The first block is a general description of the system. In Chapter 1 we make a brief introduction on the cosmic radiation. We will describe their composition, flux, and other properties. The cosmic rays are the source of radiation in our scanner.

In Chapter 2 we present the TimTrack algorithm [32]. As we will see, this mathematical tool is used to identify the trajectory of particles and perform the reconstruction of images.

The last chapter of the first block is devoted to the RPC detectors. In Chapter 3 we will describe a new model of RPC designed and constructed by the LIP group in Coimbra (Portugal). This new model, called muTT, is the detector that was used in the prototype of the scanner.

The second block of the thesis is fully devoted to the software simulations. First, in Chapter 4 we present the tools that we implemented to simulate a faithful representation of our experimental set-up. All simulations were done using the EnsarRoot framework, developed in the Universidad de Santiago de Compostela [26]. This software provides a Monte Carlo simulation and analysis tools, together with a easy implementation of the detector geometry. With EnsarRoot, we first simulated the general scattering properties of cosmic radiation in several materials. After that, in Chapter 5 we simulated physical observables that could help in the material identification.

The third block consists on the analysis of the real data provided by the scanner. These real information was used in Chapter 6 to perform a calibration of the RPC detectors and

make further corrections on data. After the calibrations were done, in Chapter 7 we present the analysis of the experimental data together with the images produced in the scanner.

A final summary of the thesis, with conclusions and remarks is done in Chapter 8.

Chapter 1

Cosmic Rays

1.1 Discovery of the cosmic radiation

It can be said that the first observations of the cosmic radiation were accidental. In the experiments performed by Coulomb and Faraday, it was observed an spontaneous discharge of their electroscopes. After the experiments carried out by Crookes, it was admitted that this discharge was a consequence of the ionisation of the surrounding air. But, what was the origin of this ionisation?

Before continuing with the chapter, we briefly quote a passage of J.J. Thompson about the works of C. Wilson and the discovery of the cosmic rays:

> During the period 1896-1900, 104 papers were published by workers in the Cavendish Laboratory. I have not space even to give their titles: one, however, I must refer to. It was the one in which C.T.R. Wilson showed that an electrified body lost its charge in dust-free-air, when he had arranged the experiment so that a defect in the insulation of the body would diminish the leak. This occurred when there were not Röntgen rays passing through the gas, and when it was shielded by thick metal from radiation from outside. It was the study of this "residual leak", so called because every known source had been eliminated, that led to the discovery of the "cosmic rays". It is one of the romances of science that the study of these very minute, and, what might seem trivial, effects should have led to results which threw much light on a subject of such great importance as the structure of the atom.

J.J. Thompson. Recollections and Reflections. G. Bell & Sons, Ltd. 1936

Becquerel, Pierre and Marie Curie shown that the radioactive substances emit charged particles; and those can be the cause of that discharge in the devices. It was shown that a

charged electroscope discharge when a radioactive element is present. So it was assumed for a long time that the ionisation of air was only caused by the radiation of substances present on the Earth surface, like radon. However, later experiments in tunnels shown that the level of radiation underground was the same as in the Earth surface. Later, Rutherford surrounded an electroscope with metal and shown that there was no significant reduction of ionisation. Therefore, there must be another explanation to the ionisation of the air beyond the *natural* radiation.

In 1909, Theodor Wulf invented a new model of a very sensitive electrometer. He used it to study the behaviour of the ionisation of air with altitude. If the radiation was produced by substances on the Earth surface, the radiation must decrease with height. To check this, he performed several measurements in the Eiffel tower [1]. He observed that the levels of radiation at the base of the tower are lower than in the top. This result is inconsistent with the hypothesis that the ionisation is produced by radiation on Earth surface. Despite of these results, no solid conclusion was accepted as valid.

After the experiments of Pacini, it was admitted the hypothesis that there is a source of radiation independent of the originated on the Earth surface. He concluded this after the study of the ionisation when the electroscopes are placed underwater. For the first time the hypothesis was that there is a source of radiation that might be from above and not from the soil.

In 1912, Victor Hess performs different experiments using several Wulf electrometers in balloon flights [2]. He observed that there is an increase of air ionisation above 1 km of altitude. One of the flights took place during a total solar eclipse. Hess observed the same increase in radiation rate as in previous flights. He then concluded that the source of this radiation is not the Sun and if the come from outer space they must have a great penetration power, because they are able to reach Earth surface.

The hypothesis of Hess was later confirmed by Kolhörster after flights beyond 9 km above sea level. For this contribution, Hess was awarded with the Nobel prize in Physics in 1936.

Cosmic rays have been studied deeper during the 20th century. Nowadays, there are many cosmic ray detectors in the world that provide information about the Universe by the study of this radiation. Large array detectors, such as in Auger observatory [3], are designed to cover large areas and collect information about high energy showers.

Other detectors, such as TRAGALDABAS [4] perform another approach to the study of cosmic rays. By the study of some observables it is possible to perform a deeper analysis on the relation between the cosmic ray flux and atmospheric properties [5, 6]. Although this radiation has been studied for more than one hundred years, there are still many unknowns

related with their origin and with all the processes affecting them before they arrive to Earth atmosphere.

This thesis studies how the cosmic rays can be used to develop a tomography system. We will briefly describe the physics behind the cosmic radiation and how it will be used in our studies. In particular, since our system it is located on Earth surface, it is important to understand the flux of the radiation that we expect, as well as its nature.

1.2   Sources of cosmic rays

It is common in literature to find the nomenclature *primary* and *secondary* cosmic rays. This is actually the notation used in the references [7, 8] and we will keep it along this chapter. However, as it is also mentioned in [7], this notation can be confusing because the same terms are also used (in a different sense) when referring to cosmic rays in atmosphere. For that reason, we believe it is convenient to use those terms only when describing the interactions between particles in atmosphere. Apart from that variation, we will adopt the same structure as in the cosmic ray chapter in [7]. For that reason, we will describe the cosmic radiation separating in different regions.

1.2.1   Extragalactic and Solar cosmic rays

Most of the cosmic radiation is mainly produced outside the Solar System. When the life of a star comes to an end, the materials produced during the nucleosynthesis processes can be ejected to the interstellar medium. For example, after a Supernova explosion the particles gain energy inside the remnant and they can scape. This radiation will be composed by carbon, iron, protons or helium; nuclei that are typically produced in stars.

The Supernovae are not the only source of cosmic rays. Quasars, galactic nuclei or gamma ray bursts are candidates to be cosmic rays sources. Further studies made by large observatories, such as Auger [3] or the data taken by the PAMELA satellite [9] could have the key of understanding the cosmic ray sources. New data from the IceCube observatory [10] propose star forming galaxies as a significant source of cosmic neutrinos [11]

This ejected material is affected by the presence of magnetic fields and also can interact with the interstellar medium. After those interactions, nuclei such as lithium or boron are obtained beside antimatter. The presence of magnetic fields in our Galaxy, the Earth or in the outer space affects significantly to the path of these particles. This fact makes very difficult the location of the cosmic ray sources.

The Sun is the closest source of cosmic radiation that we have. The solar flares and coronal mass ejections are produced when there is a reconfiguration of the magnetic field of the Sun or by a shock. The energy of this radiation is around 10^{10} eV. There is a correlation between the solar and galactic cosmic rays, known as Forbush decrease: when the ejections from the Sun displace a part of the galactic cosmic rays and therefore there is a decrease in the detection rate.

The intensity of these primary nucleons at top atmosphere level is given by the approximation [7]:

$$I_N(E) \approx 1.8 \cdot 10^4 \left(\frac{E}{1\,GeV} \right)^{-\alpha} \frac{\text{nucleons}}{m^2\,sr\,s\,GeV} \tag{1.1}$$

where E is the energy-per-nucleon, and $\alpha = 2.7$ is called the differential spectral index. This equation is valid for an energy range from several GeV until 100 TeV. Figure 1.1 shows the flux of primary cosmic radiation. As it can be seen in this figure, most of the particles have an energy greater than 10 GeV. This is a consequence of the interaction between the cosmic rays and the solar wind. The charged particles that are ejected from the Sun decelerate the primary radiation. Therefore, these primary particles are screened and this produces an anti-correlation between solar activity and intensity at low energy range.

1.2.2   Atmospheric cosmic rays

The cosmic radiation that is ejected from the Sun, the Milky Way or from other galaxies is constantly arriving to the Earth. During its travel, the radiation is exposed to magnetic fields, further interactions with other particles or by decay processes. When it arrives to our atmosphere this radiation is mostly composed by protons, electrons, positrons and some nuclei as shown in Fig.1.1. After these primary particles interact with the atmosphere, other particles are produced. These other products, that are the result of the interaction of the primary radiation with the atmosphere or by the decay of particles, are called *secondary* cosmic rays.

The primary cosmic rays interact with the nuclei and molecules present in the air, mainly oxygen and nitrogen. The probability of interaction depends on the air density and the energy of the primary cosmic rays. The most energetic primary particles will travel longer, and along their paths more secondary particles can be produced. The collisions between hadrons and the air molecules are mediated by the strong interaction. Most of the secondary particles are pions, and they also can decay to produce more secondary particles. As a numerical example, a charged pion with an energy of the order of 1 GeV travels 60 m and lives only 26 ns [12]. Therefore, most of the pions decay close to the altitude they were produced. In Fig.1.2 it is shown how the primary cosmic rays interact with the air molecules in the atmosphere.
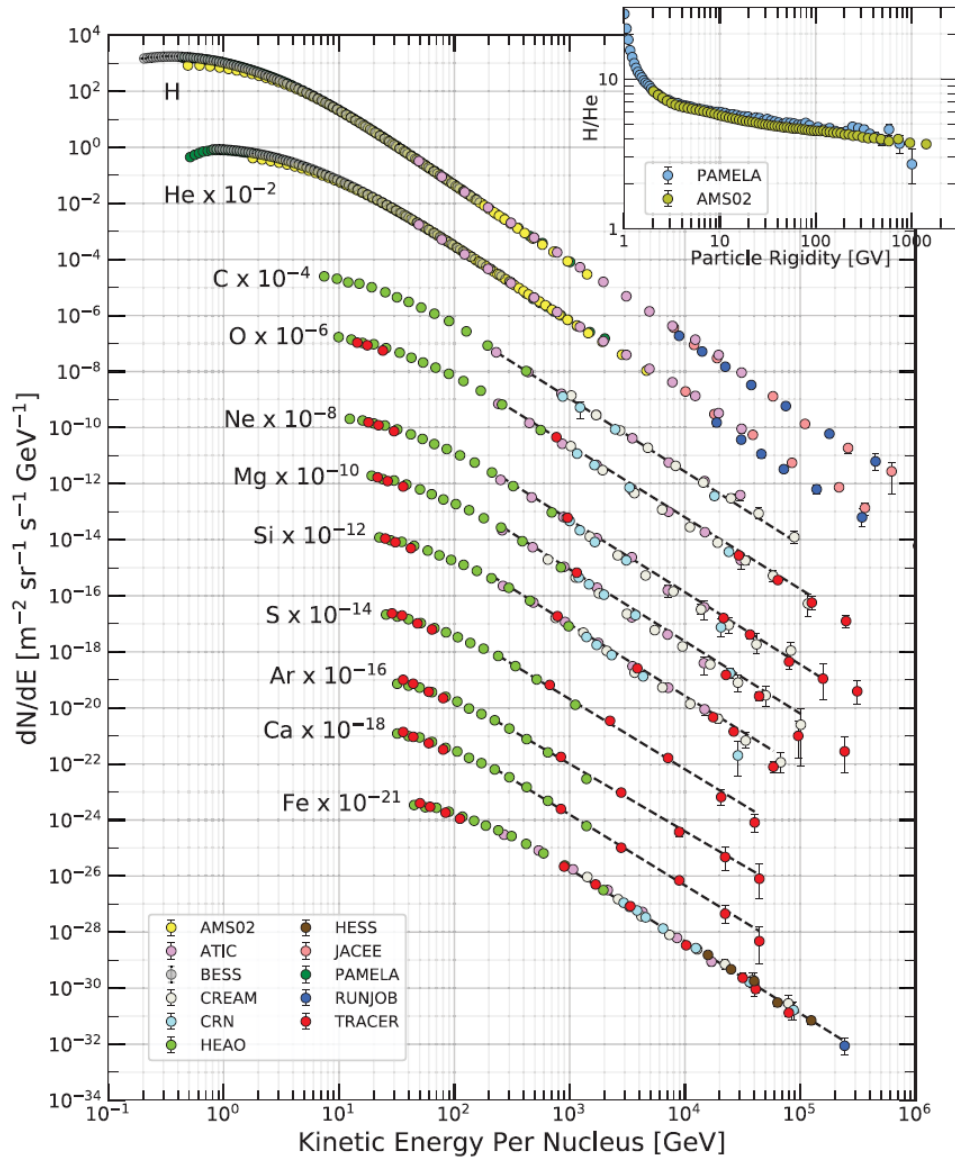
Fig. 1.1 Flux of primary cosmic rays as a function of energy. It is observed that below 10 GeV the flux is low. This is due to the fact that low energy cosmic rays are screened by the solar plasma.

The secondary particles (pions) also decay to produce muons and photons, depending on the charge of the pion:

$$\pi^{\pm} \rightarrow \mu^{\pm}\nu_{\mu}\,(\bar{\nu}_{\mu})$$
$$\pi^{0} \rightarrow 2\gamma$$

Secondary particle production in atmosphere and rock
After Gosse and Phillips, 2001

Top of atmosphere
(~ 35 km)

Air

Rock

μ	muons
e+	positron
e-	electron
λ	photons (gamma rays)
K	kaons
π	pions
p	protons
n	neutrons
v	neutrino

Electromagnetic component Hadronic component Mesonic component
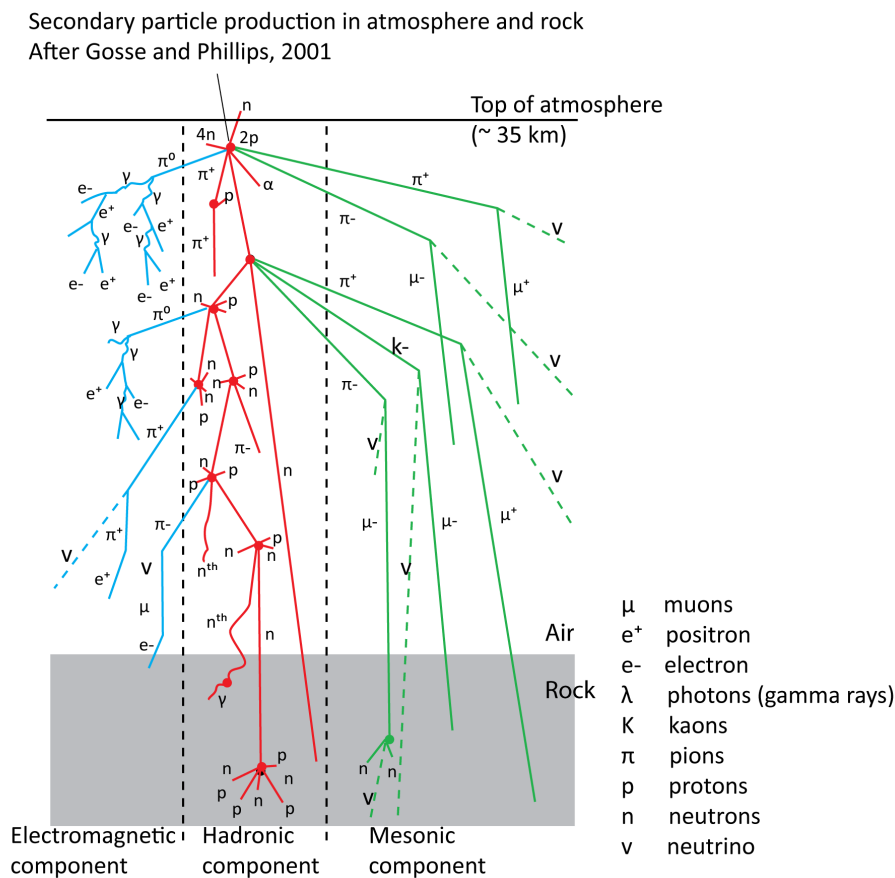
Fig. 1.2 Secondary cosmic rays produced at the Earth atmosphere. When a primary cosmic ray interacts at the top atmosphere several physical processes produce a cascade of particles that reach the sea level. Figure taken from www.antarticglaciers.com

On the other hand we have also the photons. If they have an energy above 1 MeV and are inside a nuclear medium, they can produce an electron-positron pair:

$$\gamma \rightarrow e^+ e^-$$

These electrons and positrons are losing their energy by bremsstrahlung, producing a new photon that can produce a new electron-positron pair.

We have then several physical processes that contribute to the production of secondary cosmic rays. We can find particle decays, collisions, pair productions, or bremsstrahlung. As a result, primary cosmic rays produce a cascade of particles when they interact with the Earth atmosphere and these products can reach the Earth surface.

1.2.3   Cosmic rays at Earth surface

In Fig.1.2 it is summarised schematically the composition of the cosmic rays showers that are produced in the atmosphere. Muons constitute the main component of our detected cosmic radiation. Beside the contribution of muons, there is an electromagnetic part that also arrives to ground. As it was mentioned, the photons produced in the π^0 decay can produce an electromagnetic cascade as the result of an electron-positron pair production combined with bremsstrahlung.

As a summary, we can assume that the cosmic radiation at sea level is formed by two main components: in one hand we have the muons and in the other hand the electromagnetic part, composed by electrons, positrons and photons. The Figure 1.3 summarises the information
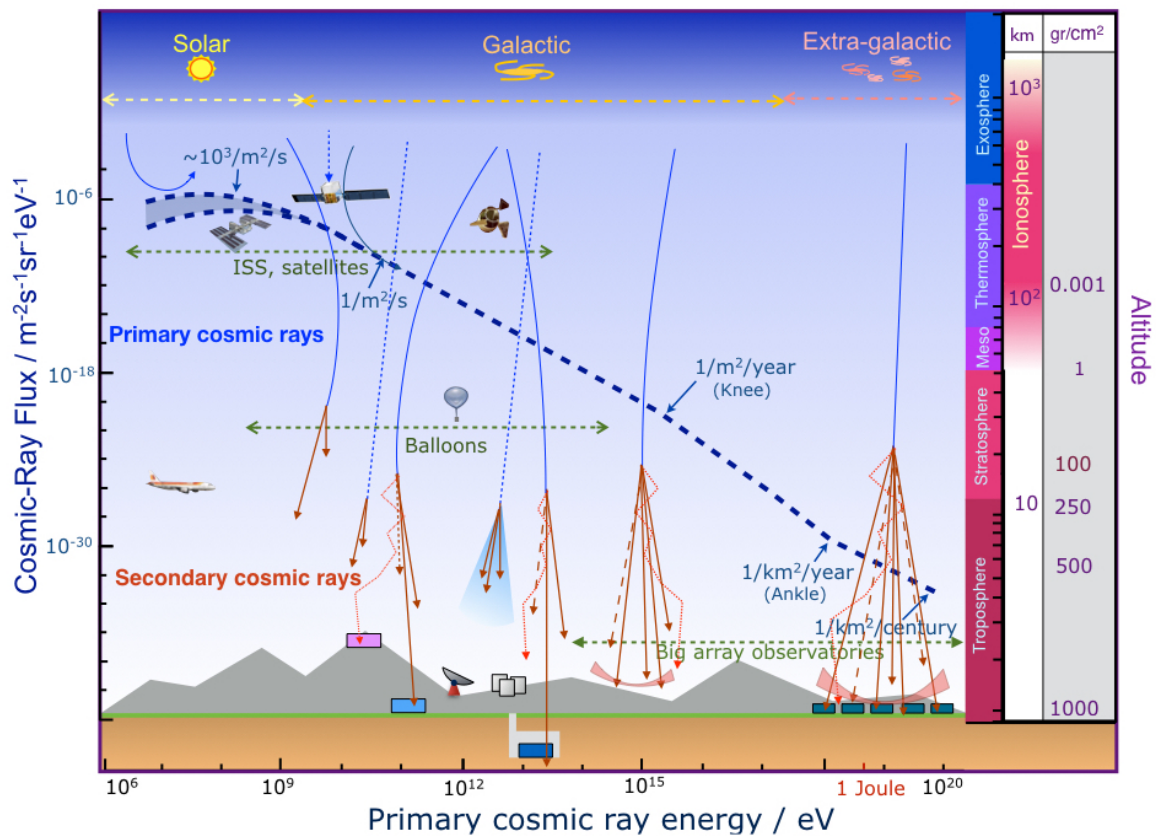
Fig. 1.3 Schematic representation of the energy spectrum of cosmic rays and most of the typical detection methods.

about the energy, sources, and detection methods of the cosmic rays.

In this figure, the blue line represents the cosmic ray flux as a function of the energy. It has a characteristic *leg* shape and it is common to distinguish between the knee and the ankle regions.

In the top part of the figure are represented different sources with their associated energies. We then have the solar cosmic rays, the radiation that comes from the galaxy, and the extra galactic sources such as supernovae, quasars, and others.

On the right part of the figure it is represented the altitude, the density, and the structure of the Earth atmosphere. The combination of all of these parameters can provide a good description of the cosmic radiation.

As we see, the Solar cosmic rays are the most abundant. However, they have affected by the magnetic fields and have low energy. Consequently, they extinguish at the top levels of the atmosphere. The most energetic solar cosmic rays (10 GeV) can produce a secondary cascade at around 35 km of altitude. The detection of this secondary cosmic rays is only possible by the use of balloons, satellites, or ground-based detectors at very high altitude.

The galactic cosmic rays have an energy range between $10^{10} - 10^{17}$ eV. They interact with the atmosphere around the troposphere limit and produce secondary cascades that can be detected at sea level. Several detectors, such as Sierra Negra (Mexico) [13] or Tibet [14] are located at around 4.000 m of altitude and are devoted to the study of these secondary cascades.

Very energetic cosmic rays coming from extra galactic sources produce large cascades on the Earth surface. The flux of these cosmic rays are 1 per km^2 and century, but they are the most energetic cosmic rays. Due to the huge amplitude of the cascades, their detection is only possible by the use of big arrays of detectors. Observatories such as Auger [3], cover large detection areas and study the possible origin of these cosmic rays.

In Fig.1.4 is presented a table that summarises the composition and flux of the cosmic radiation at sea level.

In the figure, are represented integrated fluxes of muons, photons, electrons, and neutrons as a function of the energy. As we see, muons are the main component for an energy up to 5 GeV.

The intensity of muons at sea level depends on the zenith angle. If we assume that the curvature of the Earth does not contribute to the angular dependency and we neglect it, the intensity follows the equation [15]:

$$I(\theta, X_h) = I(0°, X_h) \exp\left(\frac{X_h}{\Lambda} (1 - \sec\theta) \right) \tag{1.2}$$

where in this equation X_h is the vertical depth (expressed in g/cm^2) and Λ is the attenuation length (also in g/cm^2). The general dependency of the intensity with angle and energy is given by:

$$I(\theta, X_h, E) = I(0°) \cos^{n(E, X_h)}(\theta) \tag{1.3}$$

Mean Flux of Cosmic Rays at the Sea Level (m⁻²s⁻¹sr⁻¹) (Adapted from: Kaye&Laby. Tables of Physical and Chemical constants, www.kayelaby.npl.co.uk)								
E(GeV)	Muons				Electrons	Photons	Protons	Neutrons
	Flux	IntFx./ %	DifFx./%	Lead Range/cm				
0.1	99	1	1	4.8	6.0	8	1.9	10*
0.2	97	3	2	12	3.0	3.5	1.5	-
0.5	86	14	11	34	1.0	1.1	0.9	1.5
1	69	31	17	69	0.4	0.4	0.5	0.7
2	46	54	23	1.3E2	0.1	0.1	0.25	*) Estimated theoretical values
5	20	80	26	3.1E2	0.02	0.02	0.1	
10	9	91	11	5.8E2			0.03	
20	3	97	6	1.1E3				

Fig. 1.4 Cosmic ray flux at sea level. The fluxes of the different components are shown. As seen, muons are the dominant particles at sea level.

Experimentally, it is known that for typical energies of $3 - 4$ GeV, the value of n is inside the interval

$$n \in [1.7, 2.3] \tag{1.4}$$

and as an approximation it is commonly accepted that the angular distribution follows a $\cos^2 \theta$ law.

If the muon decay is negligible, it holds the empirical formula for muons [7]:

$$\frac{dN_\mu}{dE_\mu d\Omega} \approx \frac{0.14 E_\mu^{-2.7}}{cm^2 \ s \ sr \ GeV} \left(\frac{1}{1 + \frac{1.1 E_\mu \cos\theta}{115 \ GeV}} + \frac{0.054}{1 + \frac{1.1 E_\mu \cos\theta}{850 \ GeV}} \right) \tag{1.5}$$

An overview of the spectrum of muons at sea level is shown in Fig.1.5. In this figure are represented muons for two different angles.
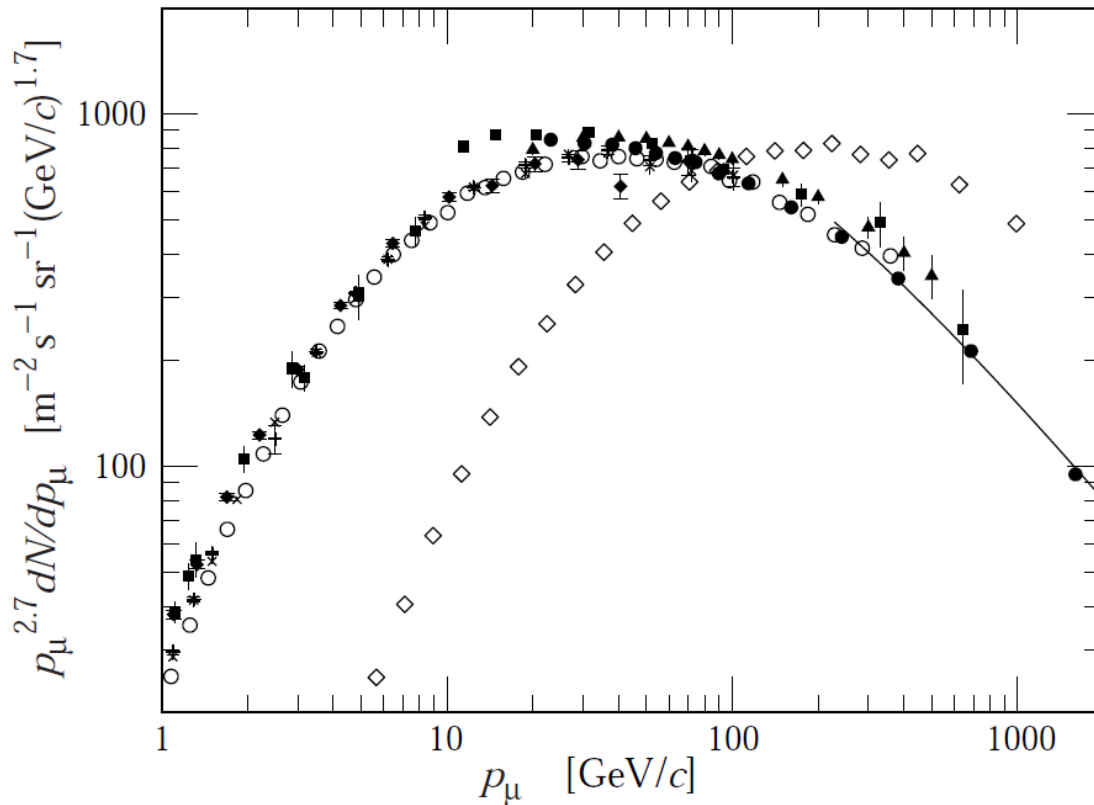
Fig. 1.5 Spectrum of muons at two angles. The $\diamond$ corresponds to an angle of $\theta = 75°$. The other points correspond to different measurements for $\theta = 0°$. Figure taken from [7]

The $\diamond$ symbol represents the muons with an angle of $\theta = 75°$. The rest of points are for muons at $\theta = 0°$.

For the purpose of this thesis, we focus on the study of the cosmic muons at sea level. The values presented in this chapter will define the structure of the simulations of the experimental set-up. The role of the electrons and gammas is not as important as for muons, and we will study them in a different manner.

Chapter 2

The TimTrack algorithm

2.1 Introduction and motivation

One of the key parts in data analysis is the reconstruction of the trajectories of the particles associated to a particular event across the experimental set-up. When a charged particle passes through a material medium, it loses part of its energy by ionisation of the medium. Therefore, it might be possible to get a signal confirming that the particle was detected. This is the basic way that a particle detector works. For each event we have a collection of all registered interactions of the particles with our detectors. This information is crucial to describe the trajectory of the particles. Knowing these trajectories is important not only to estimate the physical features of our detectors (calibration, efficiency, acceptance) but also to identify the nature of particles and then perform a more detailed physics analysis.

Not all detectors are designed to provide the same type of information. For example, we can have detectors in our experimental set-up that provide a good information about the velocity of the particle. Also, there are detectors that provide information about the position where the interaction took place. In a typical experimental set-up it is common to find a set of several types of detectors collecting different type of information. This is the starting point of the data analysis, and this information needs to be processed in order to obtain a good description of the trajectories or other variables.

There are many techniques and algorithms that provide information of the trajectory of particles. Traditionally, these tracking analysis are made in two steps: first, the spatial information is taken to determine all possible trajectories that the particle could have in the detector. After this first discrimination, the time information is used to cut the events that are not compatible in time. This two-step process, in which time and space variables are treated separately, has been used since decades. One of the main reasons to do this is because first detectors had a very poor (or even null) temporal resolution. In that case, it is not worth

performing a combined analysis of the variables. The lack of information about the time resolution makes not possible an analysis of correlations between space and time variables.

Modern RPC detectors are able to provide space and time information with good accuracy. Therefore, it seems to be reasonable to propose new tracking methods that abandon the previous two-step strategy and perform this analysis in one step.

This chapter is devoted to the description of the TimTrack algorithm. The method was developed by J. Garzón and P. Cabanelas [31, 32] and it has been adopted as the reference reconstruction method for the Trasgo family of detectors. Unlike the above mentioned methods, TimTrack performs the reconstruction of trajectories using both spatial and temporal variables together, all in one step. This point is very important since this means that the method is able to manage information from different detectors, independently of the variable that they measure. If, for example, our set-up consists on a combination of RPCs and plastic scintillators, the method is capable of unifying all information and provide a summarised description of the system.

From the mathematical point of view, one of the main advantages of the algorithm is that TimTrack is based on the least squares method. This fact makes possible a more comfortable formulation of the equations because a full matrix description of the method is possible.

This chapter is divided in three main parts. First, we present the general formulation of the TimTrack method. Second, there is a description of the models used in real physics environments. Third, numerical examples of the method.

2.2   General description of the method

The TimTrack algorithm is based on the least squares method. The work of G. Audi [18] shows a full matrix description of this method, and is taken as the starting point of the mathematical formulation of TimTrack.

The goal of TimTrack is to determine all the parameters that are related to the propagation of the particle taking as an input a collection of experimental data. These parameters are stored in an object called **saeta**[1], and therefore a saeta is a vector that contains all the kinematical information of the particle. The minimum set of parameters are contained in a saeta of the kind:

$$\mathbf{s} = \left(X_0, X', Y_0, Y', T_0, S\right) \tag{2.1}$$

where in this equation

---

[1]The name *saeta* stands for Set of All paramETers Available
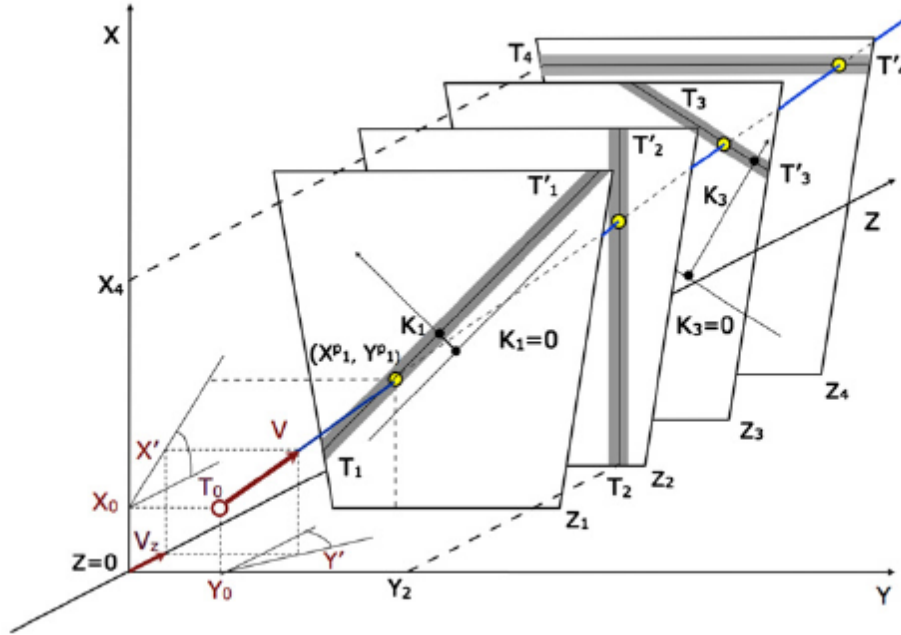
Fig. 2.1 Geometrical representation of a saeta in a generic strip detector. A saeta (blue line) contains different kinematic parameters related to the propagation of the particle trough the components of our experimental set-up. The figure was taken from [32].

- X_0, Y_0 are the x and y coordinates of the origin.

- X', Y' are the projections of the propagation along the z-axis, $\Delta x / \Delta z$ and $\Delta y / \Delta z$.

- T_0 is the arrival time referred to an external counter.

- S is the *slowness*, that is, the inverse of the velocity of the particle, $S = 1/v$.

With these six parameters the trajectory of a particle can be fully described. The equation (2.1) shows the simplest form of a saeta that describes the propagation of a particle. Depending on the situation the parameters can be different and other parametrisations of the problem can be done.

As an example, figure 2.1 shows an experimental set-up consisting in four plane parallel detectors. A particle is crossing the set-up and with each detector we obtain spatial and temporal coordinates. After the reconstruction is made with TimTrack, the propagation of the particle is given by the parameters presented in equation (2.1) as a saeta vector. As shown in the figure, the X' and Y' parameters represent the slope of the trajectory along the propagation direction, z.

Essentially, the TimTrack method provides a connection between two different spaces: data parameters. This connection is done assuming that our measured data follow a certain

model that depends on our parameters. As a summary, to apply the TimTrack method we need:

1. A set of experimental data. They are grouped in a vector called **d**.

2. A set of parameters. They are presented as a saeta **s**.

3. A model that connects both spaces: $\mathbf{d} = \mathbf{m}(\mathbf{s})$.

From the above it is obvious that the dimension of **m** equals the dimension of **d**. We called n_d the dimension of both spaces. It is important to note that n_d is not the *size* of the space, but the number of coordinates that form each element. In general, we have a total number of parameters n_s that should fulfil $n_d \geq n_s$.

The parameters grouped in a saeta are obtained by the minimisation of a mathematical functional, *S*. This operator, in the least squares method, is related to the statistical distance between the measured data and the model. Therefore, the values of the parameters should satisfy that the distance between the model and the experiment, which is called *residual*, is minimal.

The mathematical form of the functional that we propose is:

$$S = \sum_i^{n_d} \left(\frac{d_i - m_i(\mathbf{s})}{\sigma_i} \right)^2 \tag{2.2}$$

where σ_i is the uncertainty associated to the i variable. The goal of the method is to calculate the values of the parameters that make this observable reach a minimum.

It is convenient, from a mathematical point of view, to write this functional in a matrix form [18, 58] as

$$S = (\mathbf{d} - \mathbf{m}(\mathbf{s}))' \cdot W \cdot (\mathbf{d} - \mathbf{m}(\mathbf{s})) \tag{2.3}$$

where W is the matrix that contains the statistical weights of the variables (σ_i) and the apostrophe represents the transpose operation over a vector or matrix.

The model that follow the data can be written as a linear expansion:

$$\mathbf{m}(\mathbf{s}) = G \cdot \mathbf{s} + \mathbf{g}_0(s) \tag{2.4}$$

where G is the Jacobian matrix:

$$G = \frac{\partial \mathbf{m}(\mathbf{s})}{\partial \mathbf{s}} = \begin{pmatrix} \frac{\partial m_1(s)}{\partial s_1} & \frac{\partial m_1(s)}{\partial s_2} & \cdots & \frac{\partial m_1(s)}{\partial s_{n_s}} \\ \frac{\partial m_2(s)}{\partial s_1} & \frac{\partial m_2(s)}{\partial s_2} & \cdots & \frac{\partial m_2(s)}{\partial s_{n_s}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial m_{n_d}(s)}{\partial s_1} & \frac{\partial m_{n_d}(s)}{\partial s_2} & \cdots & \frac{\partial m_{n_d}(s)}{\partial s_{n_s}} \end{pmatrix}$$

From the previous equation it follows that

$$\mathbf{g}_0(s) = \mathbf{m}(\mathbf{s}) - G \cdot \mathbf{s}$$

and with these objects, the functional S takes the form:

$$S = (G\mathbf{s})' \cdot W \cdot (G\mathbf{s}) - 2(G\mathbf{s})' \cdot W \cdot (\mathbf{d} - \mathbf{g}_0) + (\mathbf{d} - \mathbf{g}_0)' \cdot W \cdot (\mathbf{d} - \mathbf{g}_0) \tag{2.5}$$

At this point, it is convenient to define new objects that are characteristic and that allow us to expressed the observable S in a more comfortable way:

$$K = G' \cdot W \cdot G \tag{2.6}$$
$$\mathbf{a} = G' \cdot W \cdot (\mathbf{d} - \mathbf{g_0}) \tag{2.7}$$
$$s_0 = (\mathbf{d} - \mathbf{g}_0)' \cdot W \cdot (\mathbf{d} - \mathbf{g}_0) \tag{2.8}$$

With these definitions the functional takes the form:

$$S = \mathbf{s}' \cdot K \cdot \mathbf{s} - 2 \cdot \mathbf{s}' \cdot \mathbf{a} + s_0 \tag{2.9}$$

So far, we only have rewritten the functional S, that is nothing else than the χ^2 variable. The set of parameters that best describes the model is given by the minimisation of the chi-square functional S:

$$\frac{\partial S}{\partial \mathbf{s}} = 0 \tag{2.10}$$

This condition leads to the equation

$$\mathbf{a} = K \cdot \mathbf{s} \tag{2.11}$$

and therefore

$$\mathbf{s} = K^{-1} \cdot \mathbf{a} \tag{2.12}$$

If we call $\mathscr{E}$ to the inverse of the K matrix, we arrive to the called *sea* equation:

$$\mathbf{s} = \mathscr{E}\mathbf{a} \tag{2.13}$$

At this point it is important to do some remarks:

1. The K matrix is called the *configuration matrix*. It is always a squared symmetric matrix of the same dimension that the set of parameters, n_s. If the model is linear it depends only on the intrinsic characteristics of the detectors and the uncertainties of the experimental data.

2. The inverse of K is the matrix that contains the uncertainties of the parameters $\mathbf{s}$. For that reason it is called *error matrix*:

$$\mathscr{E} = \left(\frac{1}{2} \frac{\partial^2 S}{\partial \mathbf{s}^2} \right)^{-1} = K^{-1}$$

3. Since $n_d \geq n_s$ it is always possible to invert the K matrix and therefore the minimisation problem can be solved. If $n_d = n_s$ the problem has an unique and exact solution.

4. The $\mathbf{a}$ vector is called *reduced data vector*. It translates the n_d data in a vector of dimension n_s.

5. The quantity s_0 is always a scalar.

After the collection of all experimental data is reduced to the vector $\mathbf{a}$, the connection between the data and parameter spaces is done with the K matrix.

One of main the advantages of the TimTrack method is that it can be used even if we have a collection of different detectors at the same time. Every experiment consists on a group of detectors that measure different variables. For example, we can have Cherenkov detectors that are able to measure velocity with good accuracy together with a Time-Of-Flight detector measuring time. Each of these detectors are described by one model m_i.

The TimTrack algorithm allows us to describe the whole array of detectors by solving a more general *sea* equation. There are some considerations to be taken into account:

1. We suppose that we have n detectors in our experimental setup. Each detector is described by an individual model m_i.

2. The models can be different, because they describe different experimental data. In a general case, each detector has its own K_i matrix and $\mathbf{a}_i$ vector.

3. The total K matrix and the total **a** vector that describe the whole set-up are the sum of the individual vectors and matrices.

4. The general *sea* equation is therefore:

$$\mathbf{s} = \left(\sum_i^n K_i\right)^{-1} \cdot \left(\sum_i^n \mathbf{a}_i\right) \tag{2.14}$$

An schematic overview of the TimTrack applied to several detectors can be seen in Fig.2.2 Each detector of the Figure is measuring different types of data. Detector number 1 measures

models:	m_1	m_2	m_3	m_4	m_5	....
data (f.e.):	(x,y)	(x,t)	(x,y,t)	(β)	(γ)	....
K matrices:	K_1	K_2	K_3	K_4	K_5	....
a vectors:	a_1	a_2	a_3	a_4	a_5	
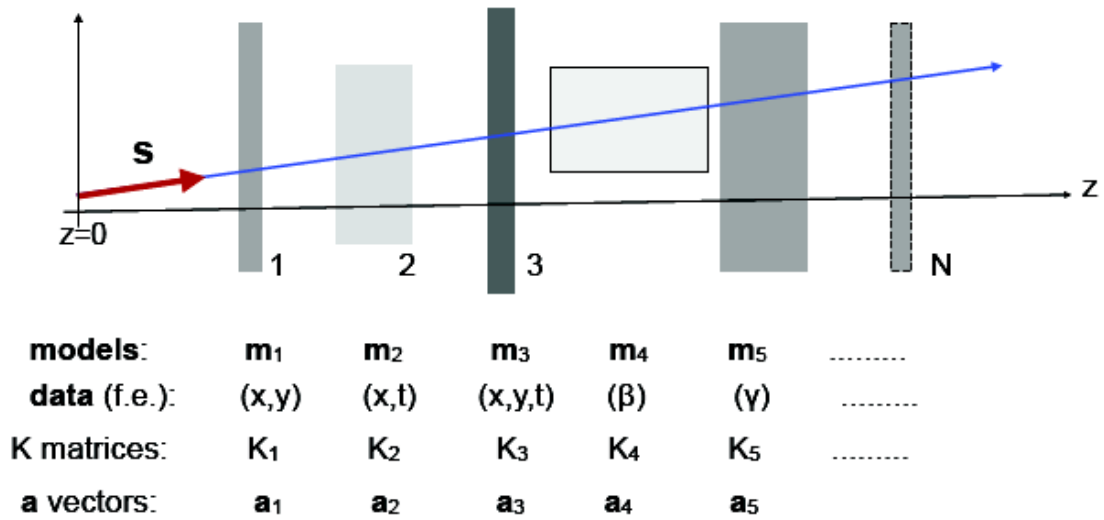
Fig. 2.2 Example of TimTrack method applied to a detector array. In this particular experimental set-up we have N different detectors each of them following a particular model for their measurements.

position (x, y), the detector number 2 (x, t), and so on. As a result, each model generates a matrix K and a vector **a** for each set of data. The saeta that represents the trajectory through all the detectors is the solution of the general *sea* equation (2.14).

2.3 Non-linear TimTrack

If we consider a non-linear model that describes the data, the solution to the problem should be found by an iterative method. In the linear case, the vector $\mathbf{g}_0$ has no dependency on the parameters. This vector, roughly speaking, is quantifying how different are our data

from linearity. The derivatives of the model, since it is linear, are constant and therefore the elements of the Jacobian matrix are constant numbers. As a consequence, these mathematical objects do not depend on the solution.

In contrast, in the non-linear case this dependency exists. We have to find the global minimum by iteration until a convergence is found. Figure 2.3 shows a simple illustration of this iterative process, where a solution is proposed at each step.
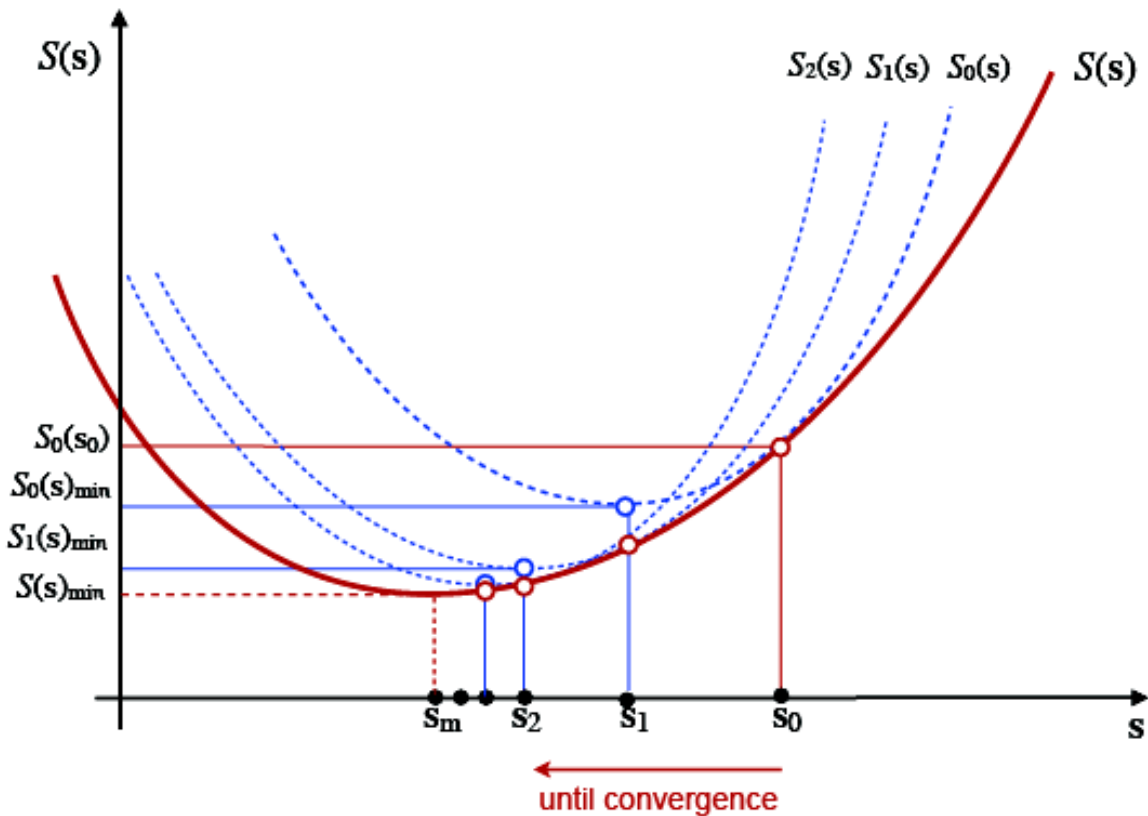
Fig. 2.3 Schematic representation of the search of a solution for a non-linear model with the TimTrack method. An initial solution $\mathbf{s}_0$ is proposed. This initial vector is modified at each step until a converged vector $\mathbf{s}_m$. For those values of the parameters, a global minimum of the functional $S(\mathbf{s})_{min}$ is reached.

The strategy is the same as the used to find the global minimum of a function with several variables. At this point we follow the description and notation of the non-linear case given by G. Kornakov [54]. The form of the functional that has to be minimised is almost the same, the difference is found in the dependency of the objects:

$$S(\mathbf{s}) = \mathbf{s}' \cdot K_0 \cdot \mathbf{s} - 2 \cdot \mathbf{s}' \cdot \mathbf{a}_0 + k_0(\mathbf{s}) \tag{2.15}$$

where now the objects have the dependency in $\mathbf{s}$:

$$
\begin{align}
K_0 &= G_0' \cdot W \cdot G_0 \tag{2.16} \\
\mathbf{a}_0(\mathbf{s}) &= G_0' \cdot W \cdot (\mathbf{d} - \mathbf{g}_0(s)) \tag{2.17} \\
k_0(\mathbf{s}) &= (\mathbf{d} - \mathbf{g}_0(s))' \cdot W \cdot (\mathbf{d} - \mathbf{g}_0(s)) \tag{2.18}
\end{align}
$$

The iterative process begins when an initial set of parameters is proposed as solution. Let's assume that the set $\mathbf{s}_0$ is close to the global minimum that solves our problem. Hence, a Taylor expansion around $\mathbf{s}_0$ can be done:

$$
S(\mathbf{s}) \approx S(\mathbf{s}_0) + \left.\frac{\partial S(\mathbf{s})'}{\partial \mathbf{s}}\right|_{\mathbf{s}_0} \delta\mathbf{s} + \frac{1}{2}\delta\mathbf{s}' \left.\frac{\partial^2 S(\mathbf{s})}{\partial \mathbf{s}^2}\right|_{\mathbf{s}_0} \delta\mathbf{s} + \ldots \tag{2.19}
$$

with $\mathbf{s} = \mathbf{s}_0 + \delta\mathbf{s}$. The condition that the minimum must satisfy is that $S(\delta\mathbf{s}) = 0$, so therefore:

$$
K_0\delta\mathbf{s} + (K_0 \cdot \mathbf{s}_0 - \mathbf{a}_0) = 0 \tag{2.20}
$$

and then

$$
\delta\mathbf{s} = K_0^{-1} \cdot \mathbf{a}_0 - \mathbf{s}_0 \tag{2.21}
$$

The new set of parameters are:

$$
\mathbf{s}_1 = \mathbf{s}_0 + \delta\mathbf{s} = K_0^{-1}\mathbf{a}_0 \tag{2.22}
$$

and we can see that this equation has the same form as (2.13). With these new values of the parameters we can recalculate the new K matrix, the new $\mathbf{a}$ vector, and so on. The new values are closer to the minimum and then the iterative process begins until a convergence is achieved. This process can be summarised in a more general equation that accounts for the recursive method:

$$
\mathbf{s}_i = K_{i-1}^{-1} \cdot \mathbf{a}_{i-1} \tag{2.23}
$$

After a few iterations a convergence is reached, and the difference $\delta\mathbf{s}$ should decrease at each iteration.

As a summary, to find a solution of a non-linear TimTrack we need to follow the next steps:

1. First, we propose a set of parameters $\mathbf{s}_0$ as initial vector. In principle, this vector must be close to the global minimum of the functional S.

2. We calculate the Jacobian matrix G_0. With it, we calculate the vector $\mathbf{g}_0(s)$.

3. With these two objects, we calculate the values of K_0 and $\mathbf{a}_0$ as the value of K and $\mathbf{a}$ evaluated at $\mathbf{s}_0$.

4. We apply the *sea* equation to find a new set of parameters $\mathbf{s}_1$.

5. The process is repeated iteratively until a convergence is reached.

2.4   Non-linear TimTrack with constraints

There are many physical situations in which the set of parameters $\mathbf{s}$ are constrained. They might satisfy some geometrical or physical conditions, as for example, energy-momentum conservation.

These boundary conditions among the parameters can be included in different ways. One possible way is to superimpose the restrictions to the parameters before the calculations are made. This redefinition could lead to a complicate calculation of the Jacobian matrix, specially if the constraints are not lineal. Using this method, the covariances between the individual parameters are lost.

Another possible way to introduce the constraints consists on the extension of the model. The boundary conditions among the parameters are defined in additional equations. This method needs an arbitrary weight factor in the W matrix that is related to the uncertainties of the physical problem.

The third possibility consists on a redefinition of the functional S including a set of n_c functions that contain the constraints of of the parameters. The minimisation of the functional is done by the use of the Lagrange multipliers. The constraints are always presented in the form:

$$f_i(\mathbf{s}) = 0 \qquad (2.24)$$

where the index i runs to n_c. If we apply the Lagrange multipliers method, we first need to construct the functional:

$$L(\mathbf{s}) = S(\mathbf{s}) + 2\sum_i^{n_c} \lambda_i f_i(\mathbf{s}) \qquad (2.25)$$

where λ_i are the set of Lagrange multipliers and the 2 is introduced for later convenience. Since the multipliers are arbitrary, an additional will not affect the final solution and in this case will simplify the final equations.

The minimum of the functional is found by the condition:

$$\frac{\partial S}{\partial \mathbf{s}} = \frac{\partial}{\partial \mathbf{s}} \left[(\mathbf{d} - \mathbf{m}(\mathbf{s}))' \cdot W \cdot (\mathbf{d} - \mathbf{m}(\mathbf{s})) + 2\sum_i^{n_c} \lambda_i f_i(\mathbf{s}) \right] = 0. \qquad (2.26)$$

This equation can be solved using the general non-linear method explained in the previous section. We can assume that the model and the constriction functions can be expanded as Taylor series of first order:

$$
\begin{aligned}
\mathbf{m}(\mathbf{s}) &\approx \mathbf{m}(\mathbf{s}_0) + \left.\frac{\partial \mathbf{m}(\mathbf{s})}{\partial \mathbf{s}}\right|_{\mathbf{s}_0} d\mathbf{s} = \mathbf{m}(\mathbf{s}_0) + G_0 \cdot d\mathbf{s} \\
\mathbf{f}(\mathbf{s}) &\approx \mathbf{f}(\mathbf{s}_0) + \left.\frac{\partial \mathbf{f}(\mathbf{s})}{\partial \mathbf{s}}\right|_{\mathbf{s}_0} d\mathbf{s} = \mathbf{f}(\mathbf{s}_0) + R_0 \cdot d\mathbf{s}
\end{aligned}
\tag{2.27}
$$

where $\mathbf{f}$ is the array of functions with the boundary conditions, and R_0 is the Jacobian matrix of the constraints evaluated at $\mathbf{s}_0$.

Using the definitions of $\mathbf{a}_0$ and $\mathbf{s}_1$ we can rewrite the condition of minimum (2.26) as

$$
K_0 \mathbf{s}_1 + R_0 \lambda = \delta \mathbf{a}_0
\tag{2.28}
$$

where $\delta \mathbf{a}_0 = \mathbf{a}_0 - K_0 \mathbf{s}_0$. If we follow the same procedure as in the non-linear case we arrive to a *sea* equation of the form

$$
\mathbf{s} = K_{i-1}^{-1} \mathbf{h}_{i-1}
\tag{2.29}
$$

where $\mathbf{h}_i = \delta \mathbf{a}_i - R_i \lambda$. As we can see, the solution of the parameters depends on the Lagrange multipliers. Therefore, we need to obtain their values. Otherwise the solution is useless.

The definition of the constraint functions and the form of the Lagrangian at minimum allow us to propose the set of equations:

$$
\begin{aligned}
\delta \mathbf{a}_0 &= K_0 \cdot d\mathbf{s} + R_0 \lambda \\
-\mathbf{f}_0 &\approx R_0 \cdot d\mathbf{s}
\end{aligned}
$$

If we rewrite these equations in a matrix form:

$$
\underbrace{\begin{pmatrix} K_0 & R_0' \\ R_0 & 0 \end{pmatrix}}_{\hat{K}_0} \cdot \underbrace{\begin{pmatrix} d\mathbf{s} \\ \lambda \end{pmatrix}}_{d\hat{\mathbf{s}}} = \underbrace{\begin{pmatrix} \delta \mathbf{a}_0 \\ -\mathbf{f}_0 \end{pmatrix}}_{\delta \hat{\mathbf{a}}_0}
\tag{2.30}
$$

we have an extension of the objects that includes the Lagrange multipliers. Again, if we apply the non-linear method, we arrive to the *sea* equation

$$
\delta \hat{\mathbf{s}}_i = \hat{K}_{i-1}^{-1} \cdot \delta \hat{\mathbf{a}}_{i-1}
\tag{2.31}
$$

with $\hat{\mathbf{s}}_i = \hat{\mathbf{s}}_{i-1} + \delta \hat{\mathbf{s}}_i$.

The errors associated to the parameters are contained, as in the other cases, in the inverse of the *K* matrix.

In the figure 2.4 it is shown how the iterative process works. An initial solution of the
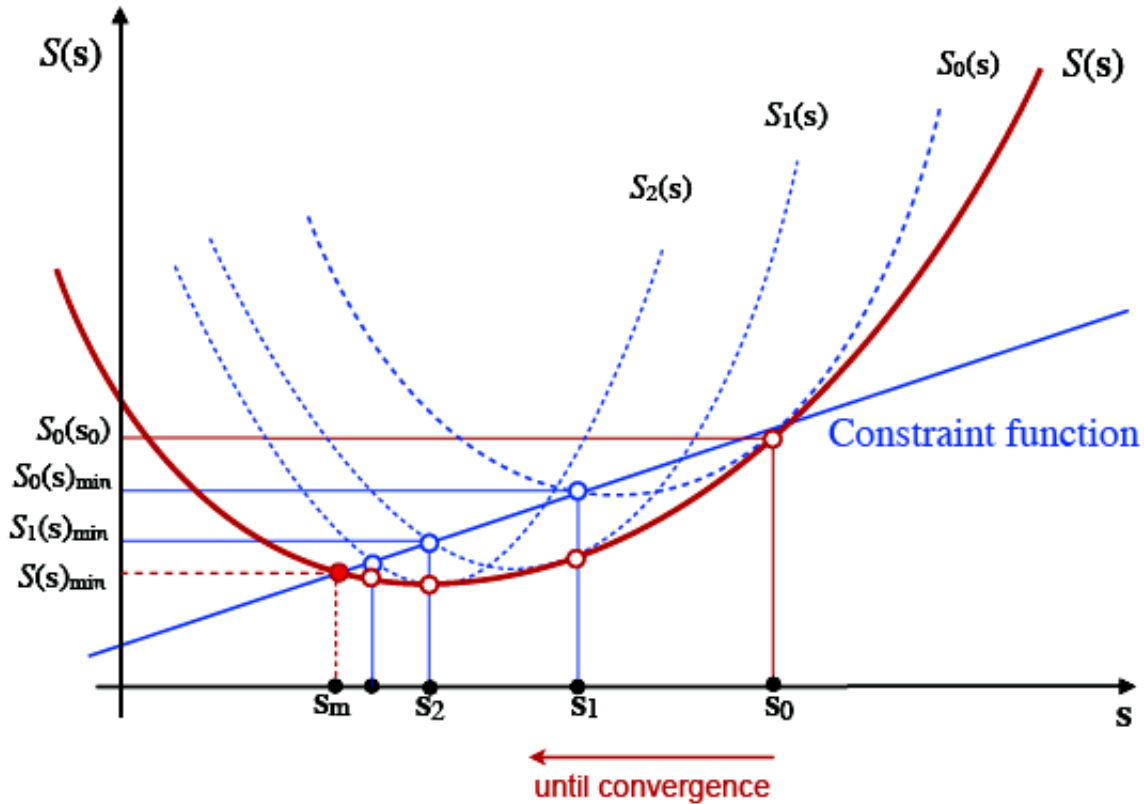
Fig. 2.4 Search of a solution of a non-linear model with constraints in the parameters. The problem is solved iteratively as the standard non-linear case, but the final values of the parameters $\mathbf{s}_m$ must fulfil the conditions of a constraint function (blue-line).

parameters is proposed and the new set of values is obtained after the restricted *sea* equation is applied. The solution must satisfy the restrictions contained in the constraint functions.

As a final remark, the non-linear TimTrack with restrictions can be summarised in the algorithm:

1. We add to the our model a set of n_c equations that describe the constraints of our parameters. Each constraint has the form $f_i = 0$.

2. The Lagrangian to be minimised is constructed as the equation (2.25).

3. We calculate the Jacobian matrix $R = \partial_{\mathbf{s}} \mathbf{f}(\mathbf{s})$.

4. An initial solution $\mathbf{s}_0$ is proposed. We evaluate the objects with these parameters.

5. The iterative process begins. We solve

$$\begin{pmatrix} \delta\mathbf{s}_1 \\ \lambda \end{pmatrix} = \begin{pmatrix} K_0 & R'_0 \\ R_0 & 0 \end{pmatrix}' \cdot \begin{pmatrix} \mathbf{a}_0 - K_0\mathbf{s}_0 \\ -\mathbf{f}_0 \end{pmatrix}$$

6. The new solution has the form $\mathbf{s}_1 = \mathbf{s}_0 + \delta\mathbf{s}_1$.

7. The iteration process continues until convergence.

2.5   TimTrack as a calibration method

The detectors present in an experimental set-up are not perfect. There are many external factors that can affect to the measurements. It is usual to apply further corrections to the experimental data because of these imperfections.

   The TimTrack algorithm can be used to determine these calibration constants of our detectors. We assume that the parameters related to the calibration are present in the mathematical model together with the kinematic parameters. In our formulation we denote these calibration constants by the vector α.

   These parameters can represent a spatial displacement that has to be applied to our detector, or a temporal shift due to the length of the cables, among other quantities. In terms of the TimTrack method, the α values are treated exactly the same way that the other parameters. Therefore, all the objects, as the Jacobian matrix, must include the derivatives respect to them and the *sea* equation provides their values.

   Mathematically, the model that describes the set of experimental data that are influenced by calibration parameters is :

$$m(\mathbf{s};\alpha) = G_A \cdot \mathbf{s}_\alpha + \mathbf{g}(\mathbf{s};\alpha) \tag{2.32}$$

where

$$G_A = \begin{pmatrix} \frac{\partial m}{\partial s} & \frac{\partial m}{\partial \alpha} \end{pmatrix}$$

and

$$\mathbf{s}_\alpha = \begin{pmatrix} \mathbf{s} \\ \alpha \end{pmatrix}$$

Note that this equation is the same as (2.4). The only difference is that we added the parameters as new set of variables. We follow the same procedure as in the previous cases to arrive to an equation that has to be solved.

The forms of the configuration matrix and the reduced data vector are:

$$K_A = G_A' \cdot W \cdot G_A \tag{2.33}$$

$$\mathbf{a}_\alpha = G_A' \cdot W \cdot (\mathbf{d} - \mathbf{g}(\mathbf{s}; \alpha)) \tag{2.34}$$

where it can be seen that they include the dependency on the parameters.

Finally, we have to solve the *sea* equation with calibration parameters:

$$\mathbf{s}_\alpha = K_A^{-1} \cdot \mathbf{a}_\alpha. \tag{2.35}$$

One of the main advantages of the TimTrack method, and in particular when is applied as a calibration tool, is that the objects (K matrix, Jacobian or the reduced data vector) can be decomposed in blocks. This makes the mathematical description more clear and compacted, and it is particularly useful when several parameters are used. For example, we can have four RPC detectors and each detector has their own calibration parameters. The global description of the array is done with a general *sea* equation as (2.14). In this case, the algebraic sum would mix spaces and provide a wrong global description. To avoid this problem we work with sparse matrices and treat each parameter individually.

2.6 Example of models and their applications

In this section we define the models that are used along the thesis for calibration and reconstruction tasks. The models are based on an experimental set-up that consists on four parallel planes, each of them providing (x, y, t) coordinates, as the schematic set-up shown in Fig.2.1. Each plane is located at z_i. Since the particle is propagating along the positive z coordinate, we considered z as a free independent variable.

The measured quantities have associated an intrinsic uncertainty, and their values are grouped in the W matrix as statistical weights. In this particular case, the uncertainties are σ_x, σ_y and σ_t. Therefore, the W matrix has the form:

$$W = \begin{pmatrix} w_x & 0 & 0 \\ 0 & w_y & 0 \\ 0 & 0 & w_t \end{pmatrix}$$

with $w_i = 1/\sigma_i^2$. This matrix will be the same in all the models presented in this section.

2.6.1 Linear model

The linear model is the most simple representation of a particle that propagates following a straight line. The equations that define the model, $\mathbf{m}(\mathbf{s})$, are:

$$\begin{aligned} m_x &= X_0 + X' \cdot z \\ m_y &= Y_0 + Y' \cdot z \\ m_t &= T_0 + S \cdot z \end{aligned}$$

As it can be seen, the x, y coordinates are the parametric representation of a straight line, where the free parameter is the z coordinate. We also assumed a linear dependency in time. This is an approximation and only true if we have vertical particles.

The unknown parameters are grouped in the saeta:

$$\mathbf{s} = (X_0, X', Y_0, Y', T_0, S).$$

as defined in Fig.2.1. To calculate the values of the parameters we need to solve the *sea* equation (2.13). For this, we need the configuration matrix (K) and the reduced data vector (**a**). Following the definition of the objects given in the array of equations (2.6), the K matrix has the form, for each plane:

$$K_i = \begin{pmatrix} w_x & w_x \cdot z_i & 0 & 0 & 0 & 0 \\ w_x \cdot z_i & w_x \cdot z_i^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & w_y & w_y \cdot z_i & 0 & 0 \\ 0 & 0 & w_y \cdot z_i & w_y \cdot z_i^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & w_t & w_t \cdot z_i \\ 0 & 0 & 0 & 0 & w_t \cdot z_i & w_t \cdot z_i^2 \end{pmatrix}$$

As we can see, the configuration matrix does not depend on the parameters **s**. It only depends on the position of the plane z_i and the uncertainties of the experimental data w_i.

The follow the same definition to construct the **a** vector:

$$
\mathbf{a}_i =
\begin{pmatrix}
w_x \cdot x_i \\
w_x \cdot x_i \cdot z_i \\
w_y \cdot y_i \\
w_y \cdot y_i \cdot z_i \\
w_t \cdot t_i \\
w_t \cdot t_i \cdot z_i
\end{pmatrix}
$$

where (x_i, y_i, t_i) is the experimental measurement at plane i. As for the K matrix, this vector is not depending on the parameters.

The number of unknowns (i.e.,the parameters to be solved) is six. Since each plane contributes with three data this problem has an exact solution for two planes (we have $n_d = 3 \times 2 = 6$ data for 6 parameters). The fact that we have two extra data vectors is contributing to get a better solution, since they are providing more information to the system.

The total K matrix and the total **a** vector are the sums of the four individual K_i and $\mathbf{a}_i$. Since these objects are not dependent on the parameters, the solution is found without an iterative process. The parameters are found by the direct application of the *sea* equation (2.13).

2.6.2 Non-linear model

The linear case is an idealisation of the trajectory of the particle across our detector. A more realistic description can be made by the use of a non-linear model:

$$
\begin{aligned}
m_x &= X_0 + X' \cdot z \\
m_y &= Y_0 + Y' \cdot z \\
m_t &= T_0 + S \cdot k \cdot z
\end{aligned}
$$

where

$$
k = \sqrt{1 + X'^2 + Y'^2}
$$

and the kinematical parameters are the same as in the linear case:

$$
\mathbf{s} = (X_0, X', Y_0, Y', T_0, S).
$$

This non-linear model takes into account the non-vertical displacement. The quantity k represents the module of the direction vector, i.e., the actual distance travelled by the particle in a vertical step of Δz. The x, y components follow also the equation of a straight line, as in the linear case.

In contrast to the linear model case, the different objects will depend on the values of the unknown parameters. The form of each K_i matrices and the $\mathbf{a}_i$ vectors are:

$$
K_i = \begin{pmatrix}
w_x & w_x z_i & 0 & 0 & 0 & 0 \\
w_x z_i & w_x z_i^2 + S^2 w_t X'^2 z_i^2/k^2 & 0 & S^2 w_t X'Y' z_i^2/k^2 & S w_t X' z_i/k & S w_t X' z_i^2 \\
0 & 0 & w_y & w_y z_i & 0 & 0 \\
0 & S^2 w_t X'Y' z_i^2/k^2 & w_y z_i & w_y z_i^2 + S^2 w_t Y'^2 z_i^2/k^2 & S w_t Y' z_i/k & S w_t Y' z_i^2 \\
0 & S w_t X' z_i/k & 0 & S w_t Y' z_i/k & w_t & w_t k z_i \\
0 & S w_t X' z_i^2 & 0 & S w_t Y' z_i^2 & w_t k z_i & w_t k^2 z_i^2
\end{pmatrix}
$$

and

$$
\mathbf{a}_i = \begin{pmatrix}
w_x x_i \\
w_x x_i z_i + S w_t X' z_i T/k \\
w_y y_i \\
w_y y_i z_i + S w_t Y' z_i T/k \\
w_t T \\
w_t k z_i T
\end{pmatrix}
$$

with

$$
T = t + S X'^2 z_i/k + S Y'^2 z_i/k
$$

As we can see these objects depend on the parameters S, X', and Y'. We also can see that in the K matrix appear correlations between the variables that are not present in the linear model case.

The total K matrix and the total $\mathbf{a}$ vector are found as in the linear case. The *sea* equation must to be solved iteratively. For that, we propose an arbitrary initial solution [2] $\mathbf{s}_0$ and we modify it solving iteratively the *sea* equation. The process stops when a convergence criteria is reached. In this thesis, we used the difference in Euclidean modules of the solutions:

$$
(|\mathbf{s}_{i+1}| - |\mathbf{s}_i|) < \varepsilon
$$

to stop the iterative process. Here, ε is arbitrary. In our case, we used $\varepsilon = 10^{-3}$ as a convergence value in this difference of modules criteria.

---

[2]It is convenient to propose an initial vector as close as possible to the global minimum.

2.6.3 Supersaetas

When the muons interact with materials they scatter due to the Coulomb multiple scattering. The shape of the material can be formed by the reconstruction of the interaction vertices. We define now a new object named *supersaeta*. A supersaeta consists on the union of two saetas that share a common point: the vertex.

We consider two models that describe the data: one for the two upper planes and another for the two lower planes. Each model generates a saeta and we impose the condition that both trajectories cross at the interaction vertex. For the calculations, we take the vertex (X_0, Y_0, Z_0) as the reference point.

The two models are:

$$
\begin{aligned}
m_{xu} &= X_0 + X_1' \cdot (z - Z_0) \\
m_{yu} &= Y_0 + Y_1' \cdot (z - Z_0) \\
m_{tu} &= T_0 + S \cdot (z - Z_0)
\end{aligned}
$$

$$
\begin{aligned}
m_{xd} &= X_0 + X_2' \cdot (z - Z_0) \\
m_{yd} &= Y_0 + Y_2' \cdot (z - Z_0) \\
m_{td} &= T_0 + S \cdot (z - Z_0)
\end{aligned}
$$

where we used the notation u and d to denote up and down. The parameters of the supersaeta are:

$$
\mathbf{s} = (X_0, X_1', X_2', Y_0, Y_1', Y_2', T_0, S, Z_0).
$$

In this supersaeta are included the parameters of both trajectories, since X_1', Y_1' are the slopes of the reconstructed trajectory in the upper planes and in a similar manner X_2', Y_2' for the lower planes. The point (X_0, Y_0, Z_0) is the vertex, T_0 is the time related to an external reference and S is the slowness. In this model we assume that the velocity of the particle is a free parameter, i.e., not constant.

The non-zero elements of the configuration matrix K_u are shown in the newt array of equations. To make the notation easier we defined the quantities

$$
k = \sqrt{1 + X_1'^2 + Y_1'^2} \qquad \text{and} \qquad dZ = z_i - Z_0
$$

with $dZ^2 = (z_i - Z_0)^2$. Since we have nine parameters, the configuration matrices are 9×9 squared symmetric matrices $k_{uij} = k_{uji}$. These are the elements of K_u:

$$
\begin{aligned}
k_{u11} &= w_x \\
k_{u12} &= w_x \cdot dZ \\
k_{u19} &= -w_x \cdot X_1' \\
k_{u22} &= w_x \cdot dZ^2 + \frac{w_t \cdot S^2 \cdot X_1'^2 \cdot dZ^2}{k^2} \\
k_{u25} &= \frac{w_t \cdot S^2 \cdot X_1'^2 \cdot dZ^2}{k^2} \\
k_{u27} &= \frac{S \cdot w_t \cdot X_1' \cdot dZ}{k} \\
k_{u28} &= S \cdot w_t \cdot X_1' \cdot dZ^2 \\
k_{u29} &= -S^2 \cdot w_t \cdot X_1' \cdot dZ - w_x \cdot X_1' \cdot dZ \\
k_{u44} &= w_y \\
k_{u45} &= w_y \cdot dZ \\
k_{u49} &= -w_y \cdot Y_1' \\
k_{u55} &= \frac{w_t \cdot S^2 \cdot Y_1'^2 \cdot dZ^2}{k^2} + w_y \cdot dZ^2 \\
k_{u57} &= \frac{w_t \cdot S \cdot Y_1' \cdot dZ}{k} \\
k_{u58} &= S \cdot w_t \cdot Y_1' \cdot dZ^2 \\
k_{u59} &= -S^2 \cdot w_t \cdot Y_1' \cdot dZ - w_y \cdot Y_1' \cdot dZ \\
k_{u77} &= w_t \\
k_{u78} &= w_t \cdot k \cdot dZ \\
k_{u79} &= -S \cdot w_t \cdot k \\
k_{u88} &= w_t \cdot k^2 \cdot dZ^2 \\
k_{u89} &= -S \cdot w_t \cdot k^2 \cdot dZ \\
k_{u99} &= w_x \cdot X_1'^2 + w_y \cdot Y_1'^2 + w_t \cdot S^2 \cdot k^2
\end{aligned}
$$

The reduced data vector $\mathbf{a}_u$ for a measurement (x_i, y_i, t_i) at the plane located at z_i has the form:

$$
\mathbf{a}_u =
\begin{pmatrix}
w_x \cdot \Delta x_i \\
-Z \cdot \dfrac{w_x \cdot k^2 \cdot \Delta x_i + S \cdot w_t \cdot X_1' \cdot [-t_i \cdot k + S \cdot (Z_0 \cdot k' - z_i \cdot q')]}{k^2} \\
0 \\
w_y \cdot \Delta y_i \\
-Z \cdot \dfrac{w_y \cdot k^2 \cdot \Delta y_i + S \cdot w_t \cdot Y_1' \cdot [-t_i \cdot k + S \cdot (Z_0 k' - z_i \cdot q')]}{k^2} \\
0 \\
w_t \cdot t_i \cdot k \cdot S \cdot \dfrac{-Z_0 \cdot k' + z_i \cdot q'}{k} \\
w_t \cdot Z \cdot t_i \cdot k \cdot S (Z_0 \cdot k' - z_i \cdot q') \\
-S \cdot t_i \cdot w_t \cdot k + w_x \cdot X_1' \cdot \Delta x_i + w_y \cdot Y_1' \cdot \Delta y_i + S^2 \cdot w_t (k' \cdot Z_0 - q' \cdot z_i)
\end{pmatrix}
$$

where in this vector

$$
\Delta x_i = x_i - X_1' \cdot Z_0, \quad \Delta y_i = y_i - Y_1' \cdot Z_0, \quad k' = 1 + 2X_1'^2 + 2Y_1'^2, \quad q' = X_1'^2 + Y_1'^2
$$

The same procedure is made for the model describing the data of the two lower planes. The non-zero values of the K_d matrix have the same form as in K_u, but as a function of X_2', Y_2'. The definition of the variables k and dZ are equivalent as in the upper case:

$$
k = \sqrt{1 + X_2'^2 + Y_2'^2} \qquad \text{and} \qquad dZ = z_i - Z_0
$$

$$k_{d11} = w_x$$

$$k_{d12} = w_x \cdot dZ$$

$$k_{d19} = -w_x \cdot X_2'$$

$$k_{d33} = w_x \cdot dZ^2 + \frac{w_t \cdot S^2 \cdot X_2'^2 \cdot dZ^2}{k^2}$$

$$k_{d36} = \frac{w_t \cdot S^2 \cdot X_2' \cdot Y_2' \cdot dZ^2}{k^2}$$

$$k_{d37} = \frac{S \cdot w_t \cdot X_2' \cdot dZ}{k}$$

$$k_{d38} = S \cdot w_t \cdot X_2' \cdot dZ^2$$

$$k_{d39} = -S^2 \cdot w_t \cdot X_2' \cdot dZ - w_x \cdot X_2' \cdot dZ$$

$$k_{d44} = w_y$$

$$k_{d45} = w_y \cdot dZ$$

$$k_{d49} = -w_y \cdot Y_2'$$

$$k_{d66} = \frac{w_t \cdot S^2 \cdot Y_2'^2 \cdot dZ^2}{k^2} + w_y \cdot dZ^2$$

$$k_{d67} = \frac{w_t \cdot S \cdot Y_2' \cdot dZ}{k}$$

$$k_{d68} = S \cdot w_t \cdot Y_2' \cdot dZ^2$$

$$k_{d69} = -S^2 \cdot w_t \cdot Y_2' \cdot dZ - w_y \cdot Y_2' \cdot dZ$$

$$k_{d77} = w_t$$

$$k_{d78} = w_t \cdot k \cdot dZ$$

$$k_{d79} = -S \cdot w_t \cdot k$$

$$k_{d88} = w_t \cdot k^2 \cdot dZ^2$$

$$k_{d89} = -S \cdot w_t \cdot k^2 \cdot dZ$$

$$k_{d99} = w_x \cdot X_2'^2 + w_y \cdot Y_2'^2 + w_t \cdot S^2 \cdot k^2$$

The same is done for the $\mathbf{a}_d$ vector:

$$
\mathbf{a}_d = \begin{pmatrix}
w_x \cdot \Delta x_i \\
0 \\
-Z \cdot \frac{w_x \cdot k^2 \cdot \Delta x_i + S \cdot w_t \cdot X_2' \cdot [-t_i \cdot k + S \cdot (Z_0 \cdot k' - z_i \cdot q')]}{k^2} \\
w_y \cdot \Delta y_i \\
0 \\
-Z \cdot \frac{w_y \cdot k^2 \cdot \Delta y_i + S \cdot w_t \cdot Y_2' \cdot [-t_i \cdot k + S \cdot (Z_0 k' - z_i \cdot q')]}{k^2} \\
w_t \cdot t_i \cdot k \cdot S \cdot \frac{-Z_0 \cdot k' + z_i \cdot q'}{k} \\
w_t \cdot Z \cdot t_i \cdot k \cdot S (Z_0 \cdot k' - z_i \cdot q') \\
-S \cdot t_i \cdot w_t \cdot k + w_x \cdot X_2' \cdot \Delta x_i + w_y \cdot Y_2' \cdot \Delta y_i + S^2 \cdot w_t (k' \cdot Z_0 - q' \cdot z_i)
\end{pmatrix}
$$

where we introduced the same variables as in the upper case:

$$
\Delta x_i = x_i - X_2' \cdot Z_0, \quad \Delta y_i = y_i - Y_2' \cdot Z_0, \quad k' = 1 + 2X_2'^2 + 2Y_2'^2, \quad q' = X_2'^2 + Y_2'^2
$$

The total K matrix is the sum of the contributions of both models, as for the $\mathbf{a}$ vector:

$$
K = \sum_{i=1}^{2} K_{ui} + \sum_{j=1}^{2} K_{dj}
$$

$$
\mathbf{a} = \sum_{i=1}^{2} \mathbf{a}_{ui} + \sum_{j=1}^{2} \mathbf{a}_{dj}
$$

and therefore the values of the parameters are obtained as the solution of the *sea* equation (2.13).

A variation of these models can be done by assuming that the velocity of the particle is constant. In that case, the total number of parameters grouped in the supersaeta is eight instead of nine:

$$
\mathbf{s} = (X_0, X_1', X_2', Y_0, Y_1', Y_2', T_0, Z_0).
$$

Therefore, the K matrices are 8×8 and the objects have a different form.

2.6.4 Time calibration parameters

The TimTrack method can be used to determine the relevant calibration constants in our detector. In this case, we consider an example of TimTrack as a time calibration tool for an RPC with a detection area made by strips. In the figure 2.5 we show the time measurement in a RPC plane. Each plane is divided in two parts, that we called *left* and *right*. When a
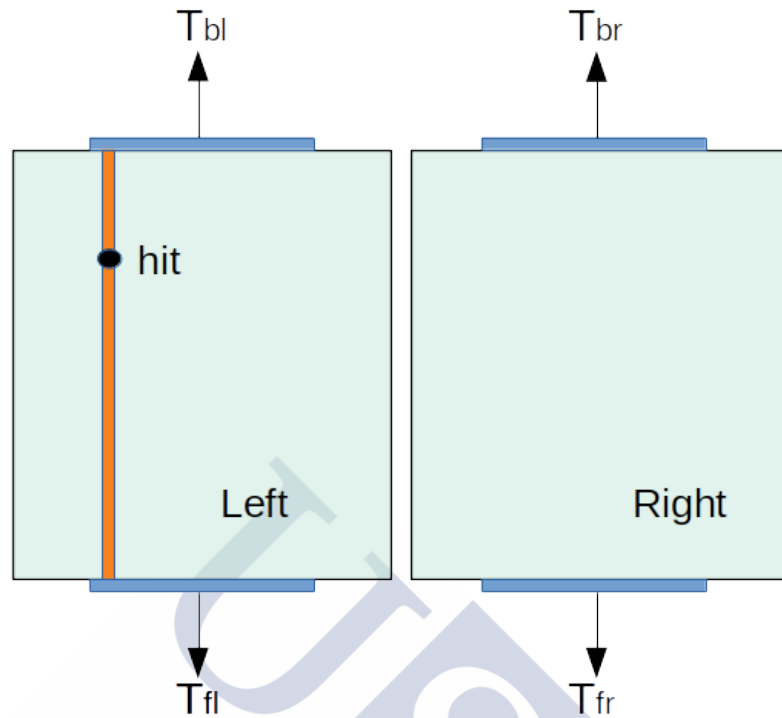
Fig. 2.5 Example of an RPC detector based on strips (orange band) and divided in two regions: left and right. Each region has an individual time readout electronics and therefore there is a shift in time depending on the position of the hit.

particle leaves a signal in a particular strip (red rectangular region) the signal travels through the strip until is collected in both extremes, providing a t_{up} and t_{down}. The measurement of time is obtained as the mean value of them:

$$t = \frac{t_{up} + t_{down}}{2}. \tag{2.36}$$

Each left and right parts have their own cables and detection systems independent to each other. This means that the measurement of time will be different depending on the fact that particle interacts on the left or right side. Thus, there is a shift in time that needs to be introduced in order to compensate both sides.

The whole calibration process is explained in more detail in Chapter 3. Here we present only the mathematical structure of the models.

In this model, we assume that the experimental set-up consists on four parallel RPCs detectors. The shift in time is done only for the first RPC. The mathematical model is:

$$
\begin{aligned}
m_{1x} &= X_0 + X' \cdot z_1 \\
m_{1y} &= Y_0 + Y' \cdot z_1 \\
m_{1t} &= T_0 + S \cdot k \cdot z_1 + \tau
\end{aligned}
$$

$$
\begin{aligned}
m_{2x} &= X_0 + X' \cdot z_2 \\
m_{2y} &= Y_0 + Y' \cdot z_2 \\
m_{2t} &= T_0 + S \cdot k \cdot z_2
\end{aligned}
$$

$$
\begin{aligned}
m_{3x} &= X_0 + X' \cdot z_3 \\
m_{3y} &= Y_0 + Y' \cdot z_3 \\
m_{3t} &= T_0 + S \cdot k \cdot z_3
\end{aligned}
$$

$$
\begin{aligned}
m_{4x} &= X_0 + X' \cdot z_4 \\
m_{4y} &= Y_0 + Y' \cdot z_4 \\
m_{4t} &= T_0 + S \cdot k \cdot z_4
\end{aligned}
$$

with $k = \sqrt{1 + X'^2 + Y'^2}$ and the shift im time is represented with τ. As mentioned, the planes 2,3, and 4 are not affected by the right-left time shift.

The parameters contained in the saeta are then:

$$
\mathbf{s} = (X_0, X', Y_0, Y', T_0, \tau)
$$

where we consider that the velocity of the particle is fixed.

As described in Equation (2.32) the Jacobian matrix, G_A, is formed as a the union of the derivatives of the model respect to the parameters and the calibration variables. We write

both matrices individually and then we construct the final Jacobian:

$$
\frac{\partial m}{\partial s} =
\begin{pmatrix}
1 & z_1 & 0 & 0 & 0 \\
0 & 0 & 1 & z_1 & 0 \\
0 & \gamma_x \cdot z_1 & 0 & \gamma_y \cdot z_1 & 1 \\
1 & z_2 & 0 & 0 & 0 \\
0 & 0 & 1 & z_2 & 0 \\
0 & \gamma_x \cdot z_2 & 0 & \gamma_y \cdot z_2 & 1 \\
1 & z_3 & 0 & 0 & 0 \\
0 & 0 & 1 & z_3 & 0 \\
0 & \gamma_x \cdot z_3 & 0 & \gamma_y \cdot z_3 & 1 \\
1 & z_4 & 0 & 0 & 0 \\
0 & 0 & 1 & z_4 & 0 \\
0 & \gamma_x \cdot z_4 & 0 & \gamma_y \cdot z_4 & 1
\end{pmatrix}
\quad
\frac{\partial m}{\partial \alpha} =
\begin{pmatrix}
0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0
\end{pmatrix}
\rightarrow G_A =
\begin{pmatrix}
1 & z_1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & z_1 & 0 & 0 \\
0 & \gamma_x \cdot z_1 & 0 & \gamma_y \cdot z_1 & 1 & 1 \\
1 & z_2 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & z_2 & 0 & 0 \\
0 & \gamma_x \cdot z_2 & 0 & \gamma_y \cdot z_2 & 1 & 0 \\
1 & z_3 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & z_3 & 0 & 0 \\
0 & \gamma_x \cdot z_3 & 0 & \gamma_y \cdot z_3 & 1 & 0 \\
1 & z_4 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & z_4 & 0 & 0 \\
0 & \gamma_x \cdot z_4 & 0 & \gamma_y \cdot z_4 & 1 & 0
\end{pmatrix}
\tag{2.37}
$$

where we defined the quantities

$$
\gamma_x = \frac{S \cdot X'}{k} \quad \text{and} \quad \gamma_y = \frac{S \cdot Y'}{k}
\tag{2.38}
$$

using the previously defined k. The weight matrix, W, is now a 12×12 diagonal matrix containing the w_x, w_y, w_t of each plane. In this description we assumed that the uncertainties are the same for all planes.

The K matrix of this model is:

$$
K =
\begin{pmatrix}
4 \cdot w_x & w_x \cdot Z_t & 0 & 0 & 0 & 0 \\
k_{12} & \frac{(S^2 \cdot w_t \cdot X'^2 + w_x \cdot k^2) \cdot Z_t^2}{k^2} & 0 & \frac{S^2 \cdot w_t \cdot X' \cdot Y' \cdot Z_t^2}{k^2} & \gamma_x \cdot w_t \cdot Z_t & \gamma_x \cdot w_t \cdot z_1 \\
k_{13} & k_{23} & 4 \cdot w_y & w_y \cdot Z_t & 0 & 0 \\
k_{14} & k_{24} & k_{34} & \frac{(S^2 \cdot w_t \cdot Y'^2 + w_y \cdot k^2) \cdot Z_t^2}{k^2} & \gamma_y \cdot w_t \cdot Z_t & \gamma_y \cdot w_t \cdot z_1 \\
k_{15} & k_{25} & k_{35} & k_{45} & 4 \cdot w_t & w_t \\
k_{16} & k_{26} & k_{36} & k_{46} & k_{56} & w_t
\end{pmatrix}
$$

where we defined

$$
Z_t = z_1 + z_2 + z_3 + z_4 \quad \text{with} \quad Z_t^2 = z_1^2 + z_2^2 + z_3^2 + z_4^2
$$

the γ_x, γ_y defined as before and since the matrix is symmetric we only wrote half of its elements.

With this elements we can build the reduced data vector for a set of measurements (x_i, y_i, t_i) (where i denotes the plane number):

$$
\mathbf{a} = \begin{pmatrix} w_x \cdot X_t \\ w_x \cdot (x_1 z_1 + x_2 z_2 + x_3 z_3 + x_4 z_4) + S \cdot w_t \cdot X' \cdot \frac{t_1 k z_1 + k \cdot (t_2 z_2 + t_3 z_3 + t_4 z_4 - S \cdot Z_t^2)}{k^2} \\ w_y \cdot Y_t \\ w_y \cdot (y_1 z_1 + y_2 z_2 + y_3 z_3 + y_4 z_4) + S \cdot w_t \cdot Y' \cdot \frac{t_1 k z_1 + k \cdot (t_2 z_2 + t_3 z_3 + t_4 z_4 - S \cdot Z_t^2)}{k^2} \\ w_t \cdot \left(T_t - S \cdot \frac{Z_t}{k}\right) \\ w_t \cdot \left(t_1 - S \cdot \frac{z_1}{k}\right) \end{pmatrix}
$$

where in this equation

$$
X_t = x_1 + x_2 + x_3 + x_4 \qquad Y_t = y_1 + y_2 + y_3 + y_4 \qquad T_t = t_1 + t_2 + t_3 + t_4
$$

and the parameters are found solving the *sea* equation.

In this model, the total matrices are not obtained like in the other models as the sum of the individual contributions of the planes. We used this particular model to illustrate another way to find the parameters by the use of matrices build in blocks. This way of proceed is particularly useful when the calibration parameters are not global and only are present in some parts of the model.

Similar to this model, we can consider the case in which the first RPC is already calibrated in time but not the others. Again, it is convenient to express the matrices in block form.

This new model can be written as:

$$
\begin{aligned}
m_{1x} &= X_0 + X' \cdot z_1 \\
m_{1y} &= Y_0 + Y' \cdot z_1 \\
m_{1t} &= T_0 + S \cdot k \cdot z_1
\end{aligned}
$$

$$
\begin{aligned}
m_{2x} &= X_0 + X' \cdot z_2 \\
m_{2y} &= Y_0 + Y' \cdot z_2 \\
m_{2t} &= T_0 + S \cdot k \cdot z_2 + \tau_2
\end{aligned}
$$

$$
\begin{aligned}
m_{3x} &= X_0 + X' \cdot z_3 \\
m_{3y} &= Y_0 + Y' \cdot z_3 \\
m_{3t} &= T_0 + S \cdot k \cdot z_3 + \tau_3
\end{aligned}
$$

$$
\begin{aligned}
m_{4x} &= X_0 + X' \cdot z_4 \\
m_{4y} &= Y_0 + Y' \cdot z_4 \\
m_{4t} &= T_0 + S \cdot k \cdot z_4 + \tau_4
\end{aligned}
$$

where we introduced a shift τ_i for each plane. Therefore, the parameters are grouped in the saeta:

$$
\mathbf{s} = (X_0, X', Y_0, Y', T_0, \tau_2, \tau_3, \tau_4)
$$

As in the previous case, we considered that the velocity of the particle is constant.

The Jacobian matrix is calculated as in equation (2.37). The first part, $\partial m/\partial s$ is exactly the same. As we have three calibration parameters, the derivatives respect to them are:

$$
\frac{\partial m}{\partial \alpha} =
\begin{pmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
\hline
0 & 0 & 0 \\
0 & 0 & 0 \\
1 & 0 & 0 \\
\hline
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 1 & 0 \\
\hline
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 1
\end{pmatrix}
$$

where the horizontal lines are drawn just for visual orientation. The lines separate the matrix in four blocks, one per plane.

The procedure to obtain the K matrix and the **a** vector is the same as in the previous model.

$$
K =
\begin{pmatrix}
4 \cdot w_x & w_x \cdot Z_t & 0 & 0 & 0 & 0 & 0 & 0 \\
k_{12} & \frac{(S^2 \cdot w_t \cdot X'^2 + w_x \cdot k^2) \cdot Z_t^2}{k^2} & 0 & \frac{S^2 \cdot w_t \cdot X' \cdot Y' \cdot Z_t^2}{k^2} & \eta_x \cdot Z_t & \eta_x \cdot z_2 & \eta_x \cdot z_3 & \eta_x \cdot z_4 \\
k_{13} & k_{23} & 4 \cdot w_y & w_y \cdot Y_t & 0 & 0 & 0 & 0 \\
k_{14} & k_{24} & k_{34} & \frac{(S^2 \cdot w_t \cdot Y'^2 + w_y \cdot k^2) \cdot Z_t^2}{k^2} & \eta_y \cdot Z_t & \eta_y \cdot z_2 & \eta_y \cdot z_3 & \eta_y \cdot z_4 \\
k_{15} & k_{25} & k_{35} & k_{45} & 4 \cdot w_t & w_t & w_t & w_t \\
k_{16} & k_{26} & k_{36} & k_{46} & k_{56} & w_t & 0 & 0 \\
k_{17} & k_{27} & k_{37} & k_{47} & k_{57} & k_{67} & w_t & 0 \\
k_{18} & k_{28} & k_{38} & k_{48} & k_{58} & k_{68} & k_{78} & w_t
\end{pmatrix}
$$

where we introduced the variables

$$
\eta_x = \frac{S \cdot w_t \cdot X'}{k} \qquad \text{and} \qquad \eta_y = \frac{S \cdot w_t \cdot Y'}{k}.
$$

The **a** vector has the form:

$$\mathbf{a} = \begin{pmatrix} w_x \cdot X_t \\ w_x \cdot (x_1 z_1 + x_2 z_2 + x_3 z_3 + x_4 z_4) + S \cdot w_t \cdot X' \cdot \frac{t_1 k z_1 + k \cdot (t_2 z_2 + t_3 z_3 + t_4 z_4 - S \cdot Z_t^2)}{k^2} \\ w_y \cdot Y_t \\ w_y \cdot (y_1 z_1 + y_2 z_2 + y_3 z_3 + y_4 z_4) + S \cdot w_t \cdot Y' \cdot \frac{t_1 k z_1 + k \cdot (t_2 z_2 + t_3 z_3 + t_4 z_4 - S \cdot Z_t^2)}{k^2} \\ w_t \cdot \left(T_t - S \cdot \frac{Z_t}{k} \right) \\ w_t \cdot \left(t_2 - S \cdot \frac{z_2}{k} \right) \\ w_t \cdot \left(t_3 - S \cdot \frac{z_3}{k} \right) \\ w_t \cdot \left(t_4 - S \cdot \frac{z_4}{k} \right) \end{pmatrix}$$

with the values of X_t, Y_t, T_t, Z_t, and Z_t^2 as defined before. The parameters are then obtained solving the *sea* equation.

A variation on this model can be done. In this case, the velocity is fixed, but it is possible to leave it as a free parameter. The sizes of vectors and matrices are obviously different. The number of degrees of freedom makes possible a solution of the problem. We would have nine parameters and twelve equations.

2.6.5   TimTrack with drift velocity inside a pad

We treat now the example of several RPC planes whose detection area is divided in rectangular pads. This is the case, as we will see in Chapter 3, of the TRAGALDABAS detector [4]. In the figure 2.6 is shown an example of a particle crossing two RPC planes. Each RPC plane is divided in rectangular detection areas or pads. When a particle interacts with the detector at some pad, the signal propagates inside the detection area until it is read at the centre of the cell. It is known that the velocity of the signal inside the pad (drift velocity) is around 60% of the speed of light.

Therefore, a model that describes the propagation of a particle including the drift velocity is:

$$\begin{aligned} m_x &= X_0 + X' \cdot z \\ m_y &= Y_0 + Y' \cdot z \\ m_t &= T_0 + S \cdot k \cdot z + S_s \cdot r \end{aligned}$$

where in this equation S_s is the inverse of the velocity of the signal, $k = \sqrt{1 + X'^2 + Y'^2}$, and r is the distance that the signal covers inside the path from the interaction point until the
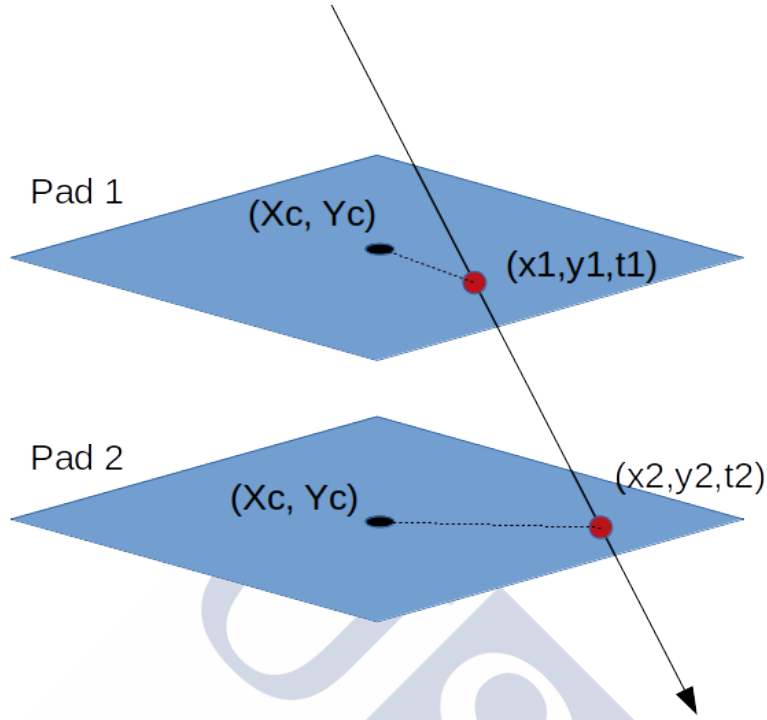
Fig. 2.6 Schematic representation of a particle detected in two pads. The detection areas of the planes are divided in rectangular regions. The signal readout is located at the centre of the pad (X_c, Y_c) and must travel from the hit (x_i, y_i, t_i) to the centre of the pad. Therefore, the model of the data includes the drift velocity of the signal inside the pad.

detection point at the centre of the pad:

$$r = \sqrt{(m_x - X_c)^2 + (m_y - Y_c)^2}$$

This is a non-linear model and has to be solved iteratively. The parameters are grouped in the saeta:

$$\mathbf{s} = (X_0, X', Y_0, Y', T_0, S)$$

where we considered that the drift velocity is constant for all planes.

The uncertainties of the measurements σ_x, σ_y are calculated from the uniform distribution of probability. Since one pad has dimensions l_x, l_y and the signal is read out at the centre, x and y are random variables distributed uniformly. Therefore:

$$\sigma_x = \frac{l_x}{\sqrt{12}} \qquad \text{and} \qquad \sigma_x = \frac{l_y}{\sqrt{12}}$$

The elements of the K matrix are:

$$k_{11} = w_x + \frac{S_s^2 \cdot w_t \cdot \Delta_x^2}{r^2}$$

$$k_{12} = z_i \cdot \left(w_x + \frac{\Delta_x \cdot S_s \cdot w_t (\Delta_x \cdot k \cdot S_s + r \cdot S \cdot X')}{k \cdot r^2} \right)$$

$$k_{13} = \frac{S_s^2 \cdot w_t \cdot \Delta_x \cdot \Delta_y}{r^2}$$

$$k_{14} = z_i \cdot \frac{\Delta_x \cdot S_s \cdot w_t (\Delta_y \cdot k \cdot S_s + r \cdot S \cdot Y')}{k \cdot r^2}$$

$$k_{15} = \frac{\Delta_x \cdot S_s \cdot w_t}{r}$$

$$k_{16} = \frac{\Delta_x \cdot S_s \cdot w_t \cdot k \cdot z_i}{r}$$

$$k_{22} = w_x \cdot z_i^2 + w_t \cdot \left(\frac{\Delta_x \cdot S_s \cdot z_i}{r} + \frac{S \cdot X' \cdot z_i}{k} \right)^2$$

$$k_{23} = z_i \cdot \frac{\Delta_y \cdot S_s \cdot w_t (\Delta_x \cdot k \cdot S_s + r \cdot S \cdot X')}{k \cdot r^2}$$

$$k_{24} = w_t \cdot \frac{(\Delta_x \cdot k \cdot S_s + r \cdot S \cdot X')(\Delta_y \cdot k \cdot S_s + r \cdot S \cdot Y')}{k^2 \cdot r^2}$$

$$k_{25} = w_t \cdot \left(\frac{\Delta_x \cdot S_s \cdot z_i}{r} + \frac{S \cdot X' \cdot z_i}{k} \right)$$

$$k_{26} = w_t \cdot z_i^2 \cdot \frac{\Delta_x \cdot k \cdot S_s + r \cdot S \cdot X'}{r}$$

$$k_{33} = w_y + \frac{S_s^2 \cdot w_t \cdot \Delta_y^2}{r^2}$$

$$k_{34} = z_i \cdot \left(w_y + \frac{\Delta_y \cdot S_s \cdot w_t (\Delta_y \cdot k \cdot S_s + r \cdot S \cdot Y')}{k \cdot r^2} \right)$$

$$k_{35} = \frac{\Delta_y \cdot S_s \cdot w_t}{r}$$

$$k_{36} = \frac{\Delta_y \cdot S_s \cdot w_t \cdot k \cdot z_i}{r}$$

$$k_{44} = w_y \cdot z_i^2 + w_t \cdot \left(\frac{\Delta_y \cdot S_s \cdot z_i}{r} + \frac{S \cdot Y' \cdot z_i}{k} \right)^2$$

$$k_{45} = w_t \cdot \left(\frac{\Delta_y \cdot S_s \cdot z_i}{r} + \frac{S \cdot Y' \cdot z_i}{k} \right)$$

$$k_{46} = w_t \cdot z_i^2 \cdot \frac{\Delta_y \cdot k \cdot S_s + r \cdot S \cdot Y'}{r}$$

$$k_{55} = w_t$$

$$k_{56} = w_t \cdot k \cdot z_i$$

$$k_{66} = w_t \cdot k^2 \cdot z_i^2$$

where we defined the quantities

$$\Delta_x = m_x - X_c = X_0 + X' \cdot z_i - X_c \qquad \text{and} \qquad \Delta_y = m_y - Y_c = Y_0 + Y' \cdot z_i - Y_c.$$

The reduced data vector, **a** is:

$$\mathbf{a} = \begin{pmatrix} w_x \cdot x_i + \Delta_x \cdot S_s \cdot w_t \cdot \frac{r \cdot S(X'^2 + Y'^2)z_i + k\left[-r^2 S_s + r \cdot t_i + S_s(\Delta_x \cdot m_x + \Delta_y \cdot m_y)\right]}{k \cdot r^2} \\ w_x \cdot x_i \cdot z_i + w_t(\Delta_x \cdot k \cdot S_s + r \cdot S \cdot X')z_i \cdot \frac{r \cdot s(X'^2 + Y'^2)z_i + k\left[r(-S_s \cdot r + t_i) + S_s(\Delta_x \cdot m_x + \Delta_y \cdot m_y)\right]}{k^2 \cdot r^2} \\ w_y \cdot y_i + \Delta_y \cdot S_s \cdot w_t \cdot \frac{r \cdot S(X'^2 + Y'^2)z_i + k\left[-r^2 S_s + r \cdot t_i + S_s(\Delta_x \cdot m_x + \Delta_y \cdot m_y)\right]}{k \cdot r^2} \\ w_y \cdot y_i \cdot z_i + w_t(\Delta_y \cdot k \cdot S_s + r \cdot S \cdot Y')z_i \cdot \frac{r \cdot s(X'^2 + Y'^2)z_i + k\left[r(-S_s \cdot r + t_i) + S_s(\Delta_x \cdot m_x + \Delta_y \cdot m_y)\right]}{k^2 \cdot r^2} \\ w_t \cdot \frac{r \cdot S(X'^2 + Y'^2)z_i + k\left[-r^2 \cdot S_s + r \cdot t_i + S_s(\Delta_x \cdot m_x + \Delta_y \cdot m_y)\right]}{k \cdot r} \\ w_t \cdot z_i \cdot \frac{r \cdot S(X'^2 + Y'^2)z_i + k\left[-r^2 \cdot S_s + r \cdot t_i + S_s(\Delta_x \cdot m_x + \Delta_y \cdot m_y)\right]}{r} \end{pmatrix}$$

used the definitions as before. The parameters are obtained from the iterative solution of the *sea* equation.

Despite this model seems to be quite complete, it has some limitations related to the calculation method. If the particle comes vertically, the final solution tends to provide the centre of the pad, as if the drift velocity would be zero. One possible solution is to make a perturbation on the saetas at each iteration. With this, the process will avoid local stable solutions and will look for the global minimum.

2.7 Numerical examples of the models

In this section we present numerical examples of some models defined before. This will help the reader to understand in more detail how the TimTrack method works.

For these examples we used a fictitious set-up consisting on four RPC planes as in Fig.2.7. Each RPC measures (x, y, t). The origin of the spatial components is taken at the central point of each RPC. We adopted the units of *cm* and *ns* for the space and time magnitudes.

We assumed that the uncertainties in x, y, and t are the same for each plane. The values that we assigned to them are:

$$\sigma_x = \sigma_y = 0.5 \, cm \qquad \text{and} \qquad \sigma_t = 0.2 \, ns$$
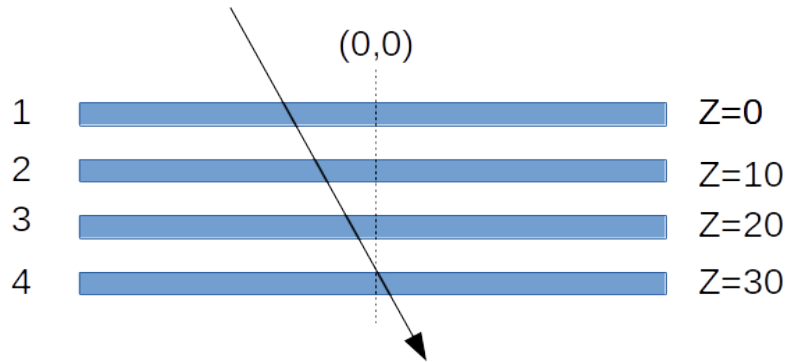
Fig. 2.7 Fictitious experimental set-up used to illustrate numerical examples of the TimTrack method. The distances are in cm. The origin of x, y is at the central point of each plane. The numbers on the left part indicate the plane number.

Therefore, the weight matrix is always constant:

$$W = \begin{pmatrix} 1/0.5^2 & 0 & 0 \\ 0 & 1/0.5^2 & 0 \\ 0 & 0 & 1/0.2^2 \end{pmatrix}$$

and even if it is build as a block matrix it will contain the same values.

2.7.1 Example of linear model

We will evaluate a numerical example of the linear model described in subsection 2.6.1. We generate a collection of hits:

$$
\begin{aligned}
(x_1, y_1, t_1) &= (-10, 20, 1.1) & \text{at} \quad z = 0 \\
(x_2, y_2, t_2) &= (0, 17, 1.45) & \text{at} \quad z = 10 \\
(x_3, y_3, t_3) &= (10, 14, 1.8) & \text{at} \quad z = 20 \\
(x_4, y_4, t_4) &= (20, 11, 2.15) & \text{at} \quad z = 30
\end{aligned}
$$

These numbers are based on a particle crossing the detector at the speed of light. The x coordinates are increasing as the y decrease.

Using the definition of the objects, we calculate the K matrix for each plane:

$$K_1 = \begin{pmatrix} 4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 25 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad k_2 = \begin{pmatrix} 4 & 40 & 0 & 0 & 0 & 0 \\ 40 & 400 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 40 & 0 & 0 \\ 0 & 0 & 40 & 400 & 0 & 0 \\ 0 & 0 & 0 & 0 & 25 & 250 \\ 0 & 0 & 0 & 0 & 250 & 2500 \end{pmatrix}$$

$$K_3 = \begin{pmatrix} 4 & 80 & 0 & 0 & 0 & 0 \\ 80 & 1600 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 80 & 0 & 0 \\ 0 & 0 & 80 & 1600 & 0 & 0 \\ 0 & 0 & 0 & 0 & 25 & 500 \\ 0 & 0 & 0 & 0 & 500 & 10000 \end{pmatrix} \quad k_4 = \begin{pmatrix} 4 & 120 & 0 & 0 & 0 & 0 \\ 120 & 3600 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 120 & 0 & 0 \\ 0 & 0 & 120 & 3600 & 0 & 0 \\ 0 & 0 & 0 & 0 & 25 & 750 \\ 0 & 0 & 0 & 0 & 750 & 22500 \end{pmatrix}$$

and then the total K matrix is:

$$K = \sum_{i=1}^{4} K_i = \begin{pmatrix} 0.175 & -0.0075 & 0 & 0 & 0 & 0 \\ -0.0075 & 0.0005 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.175 & -0.0075 & 0 & 0 \\ 0 & 0 & -0.0075 & 0.0005 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.028 & -0.0012 \\ 0 & 0 & 0 & 0 & -0.0012 & 0.00008 \end{pmatrix}$$

The same procedure is made with the **a** vector:

$$\mathbf{a}_1 = \begin{pmatrix} -40 \\ 0 \\ 80 \\ 0 \\ 27.5 \\ 0 \end{pmatrix} \quad \mathbf{a}_2 = \begin{pmatrix} 0 \\ 0 \\ 68 \\ 680 \\ 36.25 \\ 362.5 \end{pmatrix} \quad \mathbf{a}_3 = \begin{pmatrix} 40 \\ 800 \\ 56 \\ 1120 \\ 45 \\ 900 \end{pmatrix} \quad \mathbf{a}_4 = \begin{pmatrix} 80 \\ 2400 \\ 44 \\ 1320 \\ 53.75 \\ 1612.5 \end{pmatrix}$$

and the total vector:

$$\mathbf{a} = \sum_{i=1}^{4} \mathbf{a}_i = \begin{pmatrix} 80 \\ 3200 \\ 248 \\ 3120 \\ 162.5 \\ 2875 \end{pmatrix}$$

The values of the parameters are obtained by solving the *sea* equation. Then, we invert the total K matrix and multiply it by the total reduced data vector:

$$\mathbf{s} = K^{-1} \cdot \mathbf{a} = \begin{pmatrix} -10 \\ 1 \\ 20 \\ -0.3 \\ 1.1 \\ 0.035 \end{pmatrix}$$

So the parameters are:

$$X_0 = -10 \quad X' = 1 \quad Y_0 = 20 \quad Y' = -0.3 \quad T_0 = 1.1 \quad S = 0.035$$

The interpretation of these parameters are:

1. The initial point, at plane 1, is $(-10, 20)$. This is the point where the particle begins its propagation.

2. The slopes are $X' = 1$ and $Y' = -0.3$. This means that the x variable changes 1 cm per z unit. The negative value in Y' means that the variable is decreasing in z.

3. The initial time is 1.1 ns, the same as the measured in plane 1.

4. The velocity of the particle is $v = 1/S = 28.6\,cm/ns$. The speed of light is $c = 30\,cm/ns$, so the velocity of the particle is compatible with the speed of light.

The other important object is the inverse of the *K* matrix. This is what we called *error matrix*:

$$\mathscr{E} = K^{-1} = \begin{pmatrix} 0.175 & -0.0075 & 0 & 0 & 0 & 0 \\ -0.0075 & 0.0005 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.175 & -0.0075 & 0 & 0 \\ 0 & 0 & -0.0075 & 0.175 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.028 & -0.0012 \\ 0 & 0 & 0 & 0 & -0.0012 & 0.00008 \end{pmatrix}$$

The diagonal elements give information about the uncertainties associated to the parameters of the saeta:

$$\sigma_i = \frac{1}{\sqrt{K_{ii}^{-1}}}$$

The non diagonal terms contain the covariances and therefore provide information about the correlations between the parameters. In this matrix, there is a negligible correlation between X_0 and X' given by the -0.0075 coefficient.

2.8 Track finding and reconstruction algorithm

Using the models presented in previous sections it is possible to implement a global reconstruction algorithm. The cosmic rays will arrive to our experimental set-up and will produce several interactions in it. When the event finishes, it is possible to describe and sketch the cascade that they produced.

We propose here an algorithm to describe and classify the particles that produced a cascade in our set-up. This algorithm is based on the structure of the TRAGALDABAS detector, but can be extended to other experimental configurations. This algorithm is a first approach to the reconstruction of cascades with production of secondary particles.

In Fig.2.8 we present an example of a hypothetical cascade produced during a particular event. As we can see in the figure, there are different objects present. It is possible that some of the tracks are a consequence of false information produced by noise inside the detector or by some other effects.

The goal of the algorithm is to construct a representation as indicated in Fig.2.8 taking as input all the hits registered in one event.

We divided the algorithm in several steps. The strategy is to remove at each step the information that has no physical meaning. For that purpose, we present some discrimination criteria for the hits.
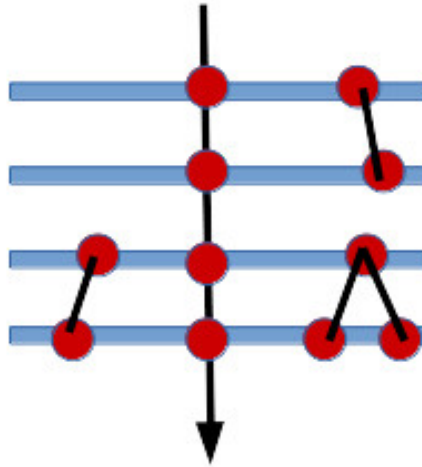
Fig. 2.8 Example of a geometrical distribution of hits registered in one event. We can find different objects depending on the nature of the particles. We propose an algorithm of identification and classification of cascades.

The algorithm is presented using an experimental set-up consisting in four parallel planes. The generalisation to *n* planes is straightforward.

Step1: compatibility between two planes

This first step of the algorithm tries to remove all trajectories that are not compatible with the speed of light. Since we are affected by the experimental uncertainties it is not convenient to impose the exact value of the speed of light. Therefore, the boundary condition can be written as:

$$v \leq c + \delta \tag{2.39}$$

where δ is fixed and can be related to the temporal resolution of our detector.

We start considering a distribution of hits as shown in Fig.2.9. To study if a couple of hits fulfil our boundary condition (2.39), we first take all the possible combinations of hits between two planes. With those two hits we construct a saeta between them that describes a virtual trajectory. This saeta can be build using whether a linear or non-linear model.

The velocity of the particle is obtained from the analysis of the virtual trajectory between those hits, since this information is given by the saeta. Therefore, the virtual trajectory between two hits is only possible if the condition (2.39) is fulfilled. Otherwise, we discard it and conclude that the particle did not follow this trajectory.

In Fig.2.9 it is shown schematically this first selection process. The red dots in the figure represent all the hits registered in one event. The algorithm takes the first hit in the first plane
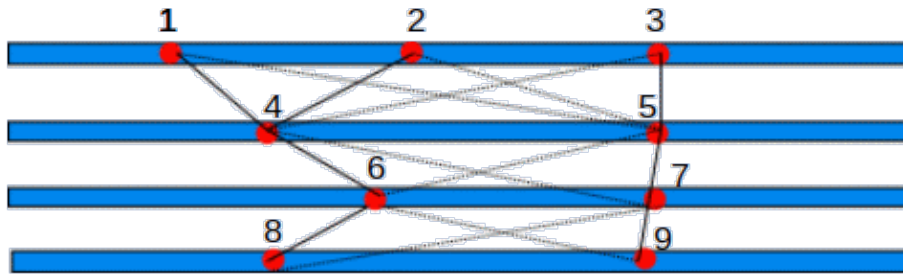
Fig. 2.9 The saetas between two planes are used to discard trajectories that are not compatible with the speed of light. The solid line represents a possible trajectory and the dashed lines are not compatible trajectories. We numbered the hits to make the description of the process easier.

and connects it with other hit in the next plane. A saeta between them is build. The solid line indicates that the trajectory is possible according to the speed of light. On the contrary, dashed lines indicate that the trajectory is not possible.

After all the combinations between plane number one and plane number two are evaluated, we proceed in the same manner between the second and third, the third and the fourth, and so on.

When all impossible trajectories are removed, we proceed to the next step.

Step 2: compatibility between three planes

We store all the information related to the virtual trajectory of the particle. This is very important from the computational point of view. When the algorithm was coded, we created a C++ class that indexes all the hits contained in a saeta. This makes possible to get access to this information at any point. Thus, we exclude already evaluated combinations or other objects.

Using the information obtained in the previous step between two planes, we extend the trajectory and propose saetas at three planes. To do that, we consider only the *saeta in, saeta out* hits. That is to say, the hits that connect and are connected by a saeta. Obviously this is not possible for the first and last planes: the hits in the first plane only can connect, and the hits in the last plane only can be connected.

When the candidate hits are selected, we construct a saeta of size three (with three hits) between them. For example, if we see Fig.2.9, these saetas will include de hits (1,4,6), (4,6,8), (2,4,6), (3,5,7), and (5,7,9). Since there is no saeta connecting (4,7), the hypothetical (1,4,7) is not considered.

One possible criteria to evaluate the trajectory consists on comparing the behaviour of the slopes of the saetas. With this, we can see if the trajectory is a straight line or if it presents a zig-zag pattern. As an example, we can compare the slope of the saeta formed by (1,4,6) with the slope of (4,6,8). These trajectories, according to the slopes, the particle clearly followed a zig-zag path. On the contrary, if we compare (3,5,7) to (5,7,9) we conclude that the trajectory is a straight line. The slope can be a good first discriminator. As in the previous step, the hits
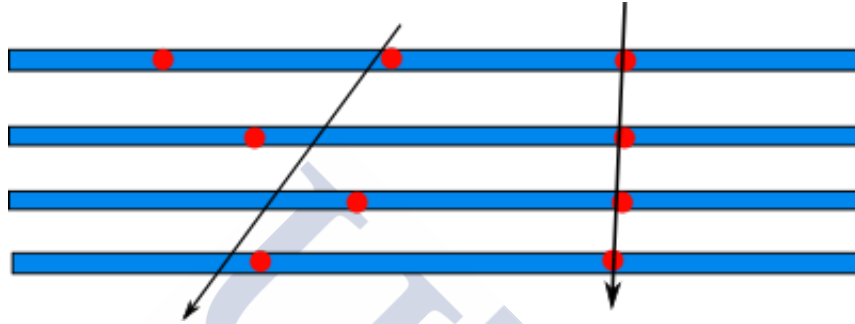
Fig. 2.10 By the study of the slopes of the saetas between three planes it can determined whether the trajectory of the particle is a straight line or not. On the right side of the figure, the different saetas between three planes provide similar slopes. However, on the left side the trajectory is not clear because the three-plane saetas have very different slopes.

that form a saeta are stored and identified by the use of an index.

If necessary, further combinations can be done to make a better discrimination. For instance, a saeta (2,6,8) could be of interest and can be compared to (2,4,8). This depends on the selection criteria that is followed.

Step3: compatibility between four planes

In our example set-up this would be the last step. After all the connections between hits are stored, we build a final object: a saeta of size four, i.e., formed by four hits.

Following the example of Fig.2.9, the possible saetas are (1,4,6,8), (2,4,6,8) and (3,5,7,9). The evaluation of the trajectory is made by a quantity related to the quality of the fit: the χ^2 value. A trajectory with a good χ^2 represents a straight line, and therefore that trajectory could correspond to a muon. On the contrary, if this value is not so good it is possible that represents an electron. In Fig.2.11 we show an schematic representation of the trajectories and their possible physical meaning. After the process is finished, we can have some trajectories that are physically possible, but that do not fit in the muon-electron pattern. For this reason, we define some objects that will help us to classify and label the cascade produced by one event.
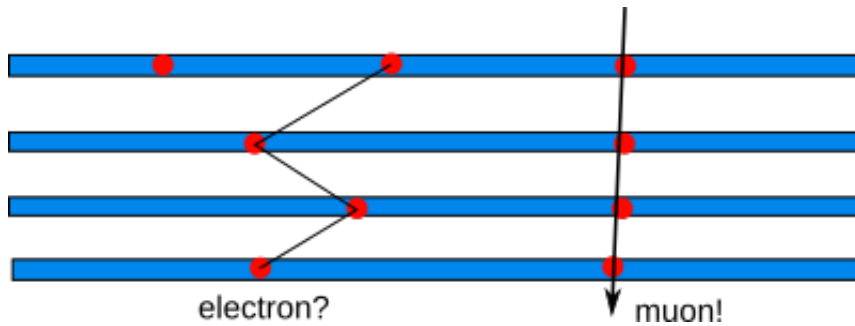
Fig. 2.11 After the formation of a saeta between the four planes, the quality of the fit can be a clue to determine the nature of the particle. On the right side, the trajectory is compatible with a straight line, and this indicates that the particle is a muon. On the left side, the trajectory is erratic and it might be an electron.

Components of a cascade

We present here some objects that are physically possible and that can be found in a cascade. As a final remark, we also proposed a nomenclature for the topology of a cascade.

In a cascade, we distinguish between a *kink*, a *vertex* and a *saeta*.

1. Kinks. A kink is defined as a trajectory that is straight but in some point of its path a new trajectory appears. Examples are shown in Fig.2.12. We denote the kinks by the
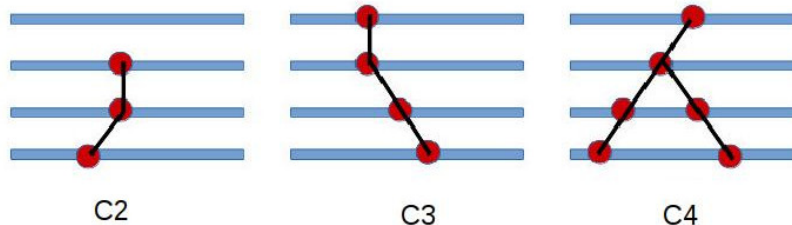
Fig. 2.12 Charged vertex or kink. In a kink, two straight trajectories share a common point.The number indicates the size of the longest trajectory.

letter C and a number that indicates the amount of hits in the longest trajectory.

2. Vertex. A vertex is the trajectory that represents a typical "V" shape. As for the kinks, the number indicates the amount of hits in the longest trajectory. It is denoted by the letter V as shown in Fig.2.13.

3. Saeta. A saeta is the standard object that is obtained by the TimTrack method. It is denoted by the letter S and the number indicates the number of planes and hits used to construct them. The saeta objects are shown in Fig.2.14.
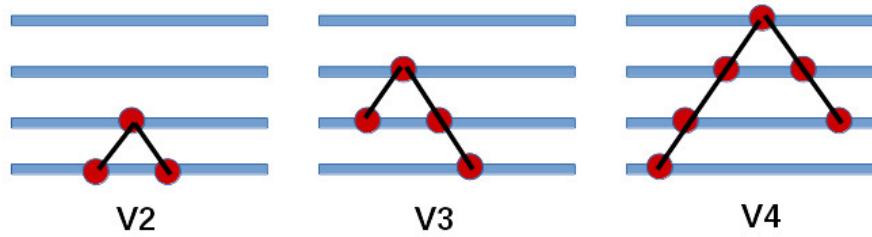
Fig. 2.13 Neutral vertex. The number indicates the size of the longest trajectory.
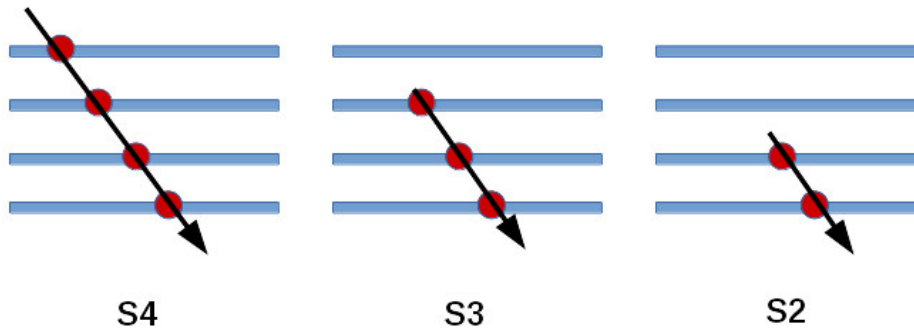
Fig. 2.14 Saetas. The number indicates the size of the object.

Cascade codification

To describe the composition of a cascade, we propose a system based on the elements that are present during one event. It is possible, for every cascade, to form a mathematical word that denotes and describes all the objects that form it. Therefore, this word contains the number of saetas, kinks and vertex that were formed. This word has the following composition:

$$
\begin{array}{ccc}
s_4 & s_3 & s_2 \\
v_4 & v_3 & v_2 \\
c_4 & c_3 & c_2
\end{array}
$$

when we denote the number of saetas, vertices and kinks in a particular order. The total number of each component is limited to 99. Since we have nine elements in total, it is possible to form a 18-bits chain with the defined structure.

There are several cascades that can be described by the same word. We say that these cascades have the same *topology* or that are topologically equivalent. It is not possible with this method to sketch a cascade from its mathematical word. Maybe a future modification of the method could make possible this distinction.

Chapter 3

Resistive Plate Chambers detectors

3.1 Introduction

One of the most used detectors nowadays in particle physics experiments are the Resistive Plate Chambers (RPCs). These detectors were first developed in the early '80s [27, 45] and they are suitable for covering great detection areas with a relative low cost per information channel. Traditionally, RPCs have been used as counter trigger detectors in experiments such as ATLAS [28]. But a redefinition in the RPC technology permits nowadays that they can be used as timing detectors. Therefore, it is possible to use RPCs to perform particle tracking.

Muon tomography requires the use of a detector fulfilling some special features. For example, the time resolution must be in the order of a nanosecond. With this resolution we will be able to identify and assign an individual track to each muon and reject possible random coincidences [19].

In this chapter it will be described the basics of the RPC detectors, their main features and why they are the perfect candidates to be the standard detector in cosmic muon tomography. There are many works that perform an exhaustive description of the physics behind RPC detectors, as for example the thesis of Lippmann [55]. Here we only present general concepts that will be used in simulation, calibration, and analysis of the detector.

In section 3.3 we will introduce the timing RPC detectors (tRPC). A part of this thesis was devoted to the development of two different tRPC: TRAGALDABAS [49, 4] and *muTT*. As it will be mentioned, these detectors are part to the *Trasgo* family. Many of the simulation and reconstruction algorithms were performed in parallel for all detectors of the *Trasgo* family.

The new *muTT* model of RPC was chosen to be the reference detector in the cosmic ray tomography system. It was designed by the LIP-Coimbra group (Portugal) and is

an improvement of the RPC technology. This detector permits to obtain better detection techniques reducing the monetary cost.

The study of the *muTT* detector is divided in two chapters. Here we make a description of the system and in Chapter 6 we present the calibration parameters, such as spatial displacement, time correction, and resolutions.

3.2   Basics of the RPC detectors

The RPCs belong to the family of the gaseous detectors. This means that the interaction of the particles with our detector occurs in a gaseous medium. After this interaction takes place, a signal is produced.

An RPC detector consists on two parallel plates filled with a gas that is ionised when a particle passes trough the detector. Between the two parallel plates there is an uniform electric field created by two parallel electrodes. To avoid discharges or perturbations in ionisation of the gas, at least one electrode must be covered with a high resistivity material.

When the gas is ionised, the electrons and ions are accelerated by a high voltage between the anode and cathode. When these particles are travelling inside the gas, more secondary particles can be created. This is called *avalanche*. A signal is then induced to the electrodes and is sent to the readout system.

The high resistivity material (of the order of $10^{10}\Omega \cdot cm$) also avoids the creation of sparks and the avalanche is not spreading all over the gas. Therefore, it helps to clean the gas after the avalanche finishes.

The main limitation of the RPC detectors, specially in the first models of Cardarelli [27], is the time needed by the high resistivity material to recover its properties. When a particle interacts with the gas and produces a signal, the detection area needs some time to clean the avalanche. This time varies depending on the material, but is on the order of milliseconds. During this time the effective electric field is weaker due to the avalanche. As a consequence, the detector has some dead time that is affecting to the time resolution. In fact, if the flux of particles is high compared to the dead time of the RPC some interactions would be lost and this affects to the efficiency of the detector as well. Modern RPCs use multiple gas gaps to increase the probability of interaction particle-gas and then improve the efficiency and time resolution. The increase in efficiency is expected to be [51, 54]

$$\varepsilon = 1 - (1 - \varepsilon_n)^n \tag{3.1}$$
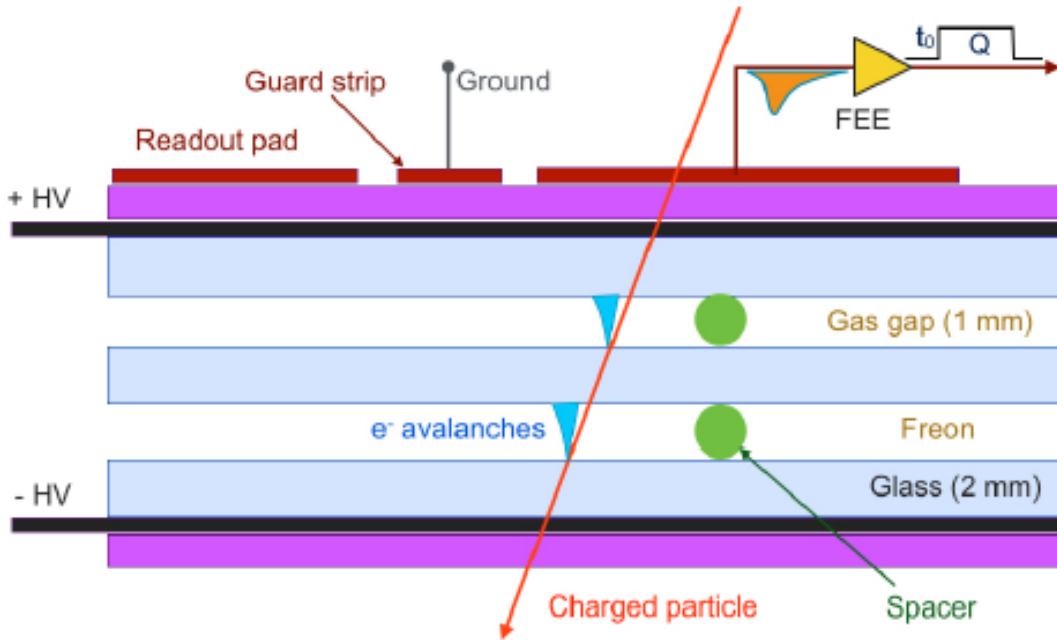
Fig. 3.1 Schematic overview of an RPC detector. In this example, we have two gaps of gas, 1 mm each. Electronic avalanches are produced by ionisation of the gas and they produce a signal that is sent to the read-out electronics.

where n is the number of gaps and ε_n is the efficiency of one gap. Typical values of the single efficiency, ε_n are of the order of 75% as shown in [53].

On the other hand, the time resolution can be estimated by [53]

$$\sigma t = \frac{\sigma_{t,n}}{\sqrt{n}} \tag{3.2}$$

where it is assumed that the behaviour of the contributions to the total fluctuations are infinitesimal and therefore the time response has a Gaussian behaviour.

The first RPC models used different gases as active medium, like Argon mixtures. Nowadays, the most used gas in the modern RPC detectors is the tetrafluorethane $(C_2H_2F_4)$[1] mixed with another gases as isobutane or sulphur hexafluoride. The proportions of the mixture is important to control the avalanches and the efficiency of the detector. As an example, the RPC detectors used at the ATLAS and CMS experiments have a gas mixture of 95.2% $C_2H_2F_4$, 4.5% i-C_4H_{10}, and 0.3% SF_6 [29]. One of the major disadvantages of this gas mixture is the presence of not environmentally friendly gases, such as SF_6 and $C_2H_2F_4$. Recent works have presented alternative gas mixtures based on Helium [16]. Another possible

---

[1]Freon is a trademark and sometimes we will use it as a generic name to any fluorocarbon. The Freon-134a (R-134a) The chemical formula of the Freon-134a is $C_2H_2F_2$.

solution is the use of a closed-circuit gas system. With this mechanism, the impurities of the gas are removed and it is not emitted to the atmosphere [44].

3.2.1   Operation modes

The RPC detectors are able to work in two different modes: *avalanche* or *streamer*. The difference between both modes stays in how intense the avalanche is.

1. **Streamer mode**: the voltages are very high and they will produce a strong avalanche that distorts the electric field. Since the induce signals are strong, no additional amplification systems are needed. This simplifies the readout electronics. This mode of operation is traditionally used when the RPC works as a trigger in a experiment.

2. **Avalanche mode**: the charge released in one avalanche is small compared to the streamer mode. In this case, the readout electronics is different because the signal must be amplified. This is the mode used for timing purposes.

The modern RPC detectors operate in the avalanche mode with a multigap configuration. With these detectors it is possible to reach time resolutions of the order of 90 ps and below. This time resolution makes the RPC detectors be a perfect candidate to perform particle tracking.

As an example, the Fig.3.1 shows an schematic overview of an RPC detector. This particular case corresponds to a two-gap configuration operating in avalanche mode. Each gas gap has 1 mm width and they are filled with R-134a. The gas gaps are isolated with 2 mm of glass. The two gaps are inside a high voltage created by two parallel planes. When a charged particle crosses the RPC it induces an ionisation inside the gas gaps. This produces an avalanche. If the size of the avalanche is large enough, it produces an influence on the electric field in the gas gap. The electrons inside the gas will move to the anode and the ions to the cathode. As a result, a perturbation in the electric field is produced around the zone where the avalanche was developed. This induces a signal that is read-out. In this case, the induction is produced in a rectangular region. Other detection techniques divide the detection region in layers of strips. This signal is processed by the electronic devices and its transformed to relevant physical information.

3.3   Timing RPC detectors (tRPC)

The use of the RPCs as a timing and tracking detector was possible after a reconfiguration of the features of the traditional RPCs. First, to work in avalanche mode it is necessary

to integrate to the electronics an amplification method for the signal. Second, the use of multi-gap structures allow to achieve time resolutions of the order of $90 - 100$ ps. It is common to find RPCs with gaps of 0.3 mm in avalanche mode.

Another modifications in the mechanical structure of the RPCs are related to the high resistivity material. Originally it was used bakelite electrodes to avoid sparks inside the gas. Nowadays it is preferred to use window glass as a standard (its resistivity is of the order of $10^{12} - 10^{13} \Omega \cdot cm$)

A high voltage must to be applied to the parallel plates. It must be strong enough to obtain a good efficiency but not as high to produce streamers and break the avalanche mode. There are tRPCs that work with voltages of $100 \, kV/cm$ in the avalanche mode with low probability of streamers.

3.4 The *Trasgo* family of detectors

The detection of high energetic cosmic rays is made indirectly by the use of large detector arrays at the Earth surface. Only a small fraction of these high energy cosmic rays arrive to the ground. Therefore, the study of the primary radiation makes the event-by-event analysis very complicated.

The *Trasgos*(acronym of TRAcksreconStructinGmOdule) were proposed after the observations of cosmic rays in the HADES spectrometer at GSI [20]. The idea was to build a high performance affordable cosmic ray detector, sensitive to the muons and electromagnetic showers, and based on RPC detectors. The Trasgos are designed to perform studies of the cosmic rays properties because the provide a good time and spatial resolutions. Therefore, they are ideal detectors for particle tracking purposes and allow a better reconstruction of the secondary cascades.

In general, the *Trasgo* family is formed by the multi-gap timing RPCs. In this thesis we worked in parallel with two different detectors of this family. First, we have the TRAGALD-ABAS detector [4], that is used to study cosmic ray properties. In the other hand, we have the new *muTT* model, that is used in the muon tomography system.

3.4.1 The TRAGALDABAS detector

The TRAGALDABAS detector is located at the University of Santiago de Compostela [49, 4]. The detector is taken data since 2013 and it is devoted to the analysis of the structure of the cosmic rays, the solar activity, and atmospheric properties. In Fig.3.2 is presented an schematic overview of the TRAGALDABAS detector. It consists on four RPC planes with
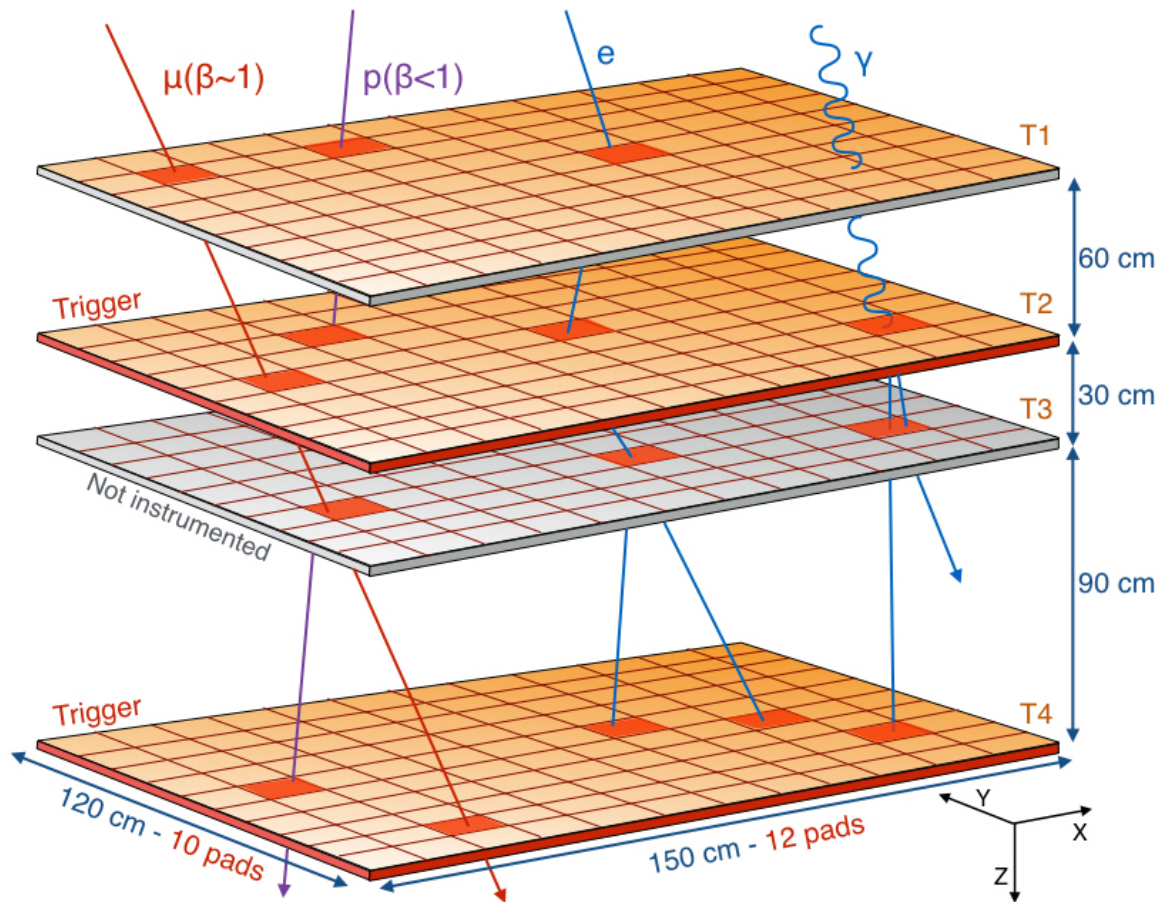
Fig. 3.2 Schematic representation of the TRAGALDABAS detector. It consists on four RPC planes divided in rectangular detectiton pads. Nowadays, the detector has three operative RPCs.

an exteranal size of $1.285 \times 1.650 \ m^2$ divided in rectangular pads. Nowadays, only three of the four detectors are fully instrumented and operative.

The RPCs of TRAGALDABAS were built at the LIP-Coimbra with the same technology as the used in the P. Auger observatory. The internal structure of the RPC are similar to the described in Fig.3.1. They consist on two gaps of 1 mm each filled with R-134a gas.

Between the external sides of the RPC there is a high voltage of $\pm 5600 \ V$ at each plate, equivalent to $5600 \ V/mm$ in each gap. This voltage induces an electric field to work in the avalanche mode. The induced signals are read-out by copper pads.

The front end electronics of TRAGALDABAS was developed at GSI for the HADES experiment[52]. In consists on the Mambo-Dumbo (MB-DB, mother board-daughter board) amplification system. The signals induced in the pads are first amplified, integrated, and digitalised in 4-channel daughter boards (DB). The DB are then connected to a mother board

(MB) that provide the output of the signal and triggers. The signals are then sent to a read-out board (TRB.V2) [30] and then to a computer to be registered.

Most of the software tools developed for the TRAGALDABAS detector were use for the develop of the new *muTT* model. As we will explain in more detail in Chapter 4 the new libraries for the simulation of the *muTT* features are a modification of the already made work in TRAGALDABAS.

3.4.2 The *muTT* detector

A new model of RPC, the *muTT*, was designed and built by the LIP-Coimbra group. This new detector belongs to the *Trasgo* family. It was designed to be the reference detector in the muon tomography system.

The new *muTT* is an improvement of the planes used in TRAGALDABAS. First, the detection region is divided in strips, instead of rectangular pads. Second, the electronic design makes the price per channel more affordable compared to previous RPC models.

The prototype of the muon scanner was set-up inside an industrial hangar located at O Porriño (Pontevedra). Several tests of the *muTT* were performed before install them in the final structure of the scanner. As we will see in Chapter 6, this previous tests were used to estimate the calibration constants of the detector.

In this chapter we provide a description of the general properties of the new *muTT* detector. We present the structure of the planes, their main features, and a brief description of the electronics.

3.5 Description of the *muTT* detector

3.5.1 Disposition of the planes and nomenclature

Before the *muTT* detector was installed in the final scanner structure, it was tested in a simplified disposition.

In Fig.3.3 it can be seen the initial disposition of the *muTT* planes. This structure was used to perform tests on the integrity and other features of the RPC planes. After the calibration tests were done, they were placed at the final scanner structure.

The planes were numbered according to their relative position in space. We divide the detector in two different zones: up and down. As it can be seen in Fig.3.3 there is an aluminium plate located at the centre of the detector that divides up and down zones. Each plane has a name that provides information about the region and the relative position inside
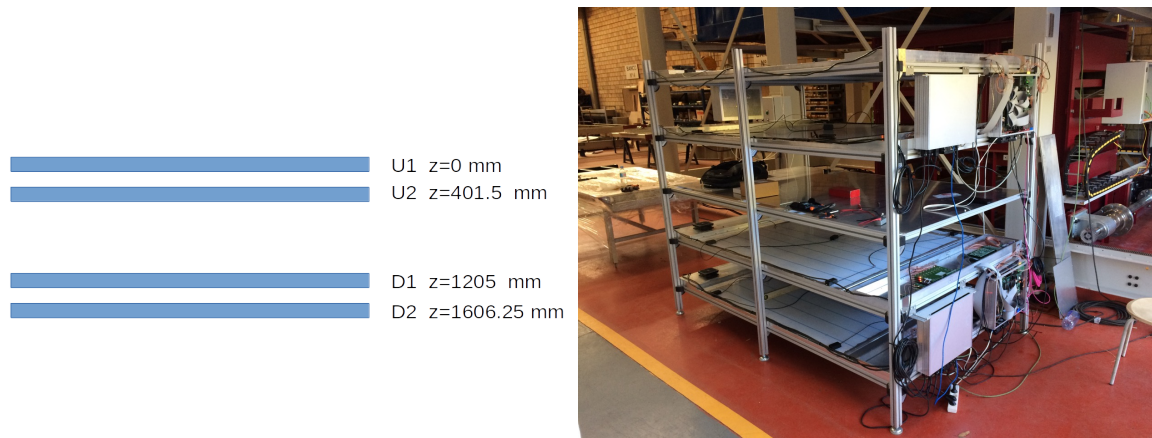
Fig. 3.3 Schematic representation of the *muTT* detector. On the right side it can be seen a picture of the structure. On the left, a summary of the distances and names of the planes.

that region. For example, U1 means *the first plane in the upper region*. This system can be useful in case other planes are added in future.

The distances between planes are taken relative to the U1 plane, i.e., the origin of the vertical *z*-component is set at U1 and we defined it as positive with the propagation of particles. The relevant information is carried by the relative distance between two planes. As we will see in other chapters, the origin of the *z*-component is not fixed in U1 and sometimes we will refer to the centre of the detector.

As a convention, it was decided that the standard units for the *muTT* detector are: distances given in mm and time in ns.

3.5.2 Description of the RPC planes

The new *muTT* model has significant differences compared to previous tRPC models, such as the used in TRAGALDABAS. The design of this detector is somehow similar to another RPC designed also by LIP-Coimbra [23].

Each *muTT* plane has an effective area of 1500×1200 *mm*^{2}. This detection area is covered by two layers of strips[2], one at the top and the other on the bottom part of the plane. Both sets of strips are perpendicular to each other.

In the Fig.3.4 there is a representation of the internal structure of a *muTT* plane. This RPC presents a multi-gap configuration.

---

[2]We will refer with the word *strip* to a group of wires. Traditionally, *strip* is used to denote the individual wires. Since we will apply corrections to individual wires, we decided to make this distinction to avoid ambiguous meanings.

Medium	Z-thickness (mm)
Aluminium box top side	3,000
Polyethylene Foam	3,100
FR4 layer	1,500
Copper	0,030
Polycarbonate box top side	1,000
Acrylic electrode layer	0,010
Glass	1,000
R-134a gap	0,300
Glass	1,000
R-134a gap	0,300
Glass	1,000
R-134a gap	0,300
Glass	1,000
R-134a gap	0,300
Glass	1,000
R-134a gap	0,300
Glass	1,000
R-134a gap	0,300
Glass	1,000
Acrylic electrode layer	0,010
Polycarbonate box bottom side	1,000
Copper	0,030
FR4 layer	1,500
Copper	0,030
Aluminium box bottom side	3,000
Aluminium base	3,000
Sum	**26,010**

Fig. 3.4 Internal structure of the *muTT* detector. Each plane is 26 mm thick and inside there are six gas gaps of 0.3 mm each. The gaps are separated by 1 mm of glass foils.

The RPC mother structure is an aluminium box with a total thickness of 26 mm. Inside this mother volume there are six gaps of R-134a gas (0.3 mm each) separated by 1 mm of glass foils.

As usual in the RPC designs, there are other materials such as foam or polycarbonate, that are used for a better isolation of the plane. The gas gaps are inside an electric field created by a high-voltage over the acrylic electrode layers.

In this particular detector, there are two layers of copper electrodes that collect the induced signal inside the gaps. This is a typical design for the RPCs using strips as element of detection [45, 27].

In Fig.3.5 is shown the disposition of the detection strips. There are two different layers, named as *up* and *down*, that are perpendicular to each other. The detection area is then represented as a grid created by the strips.
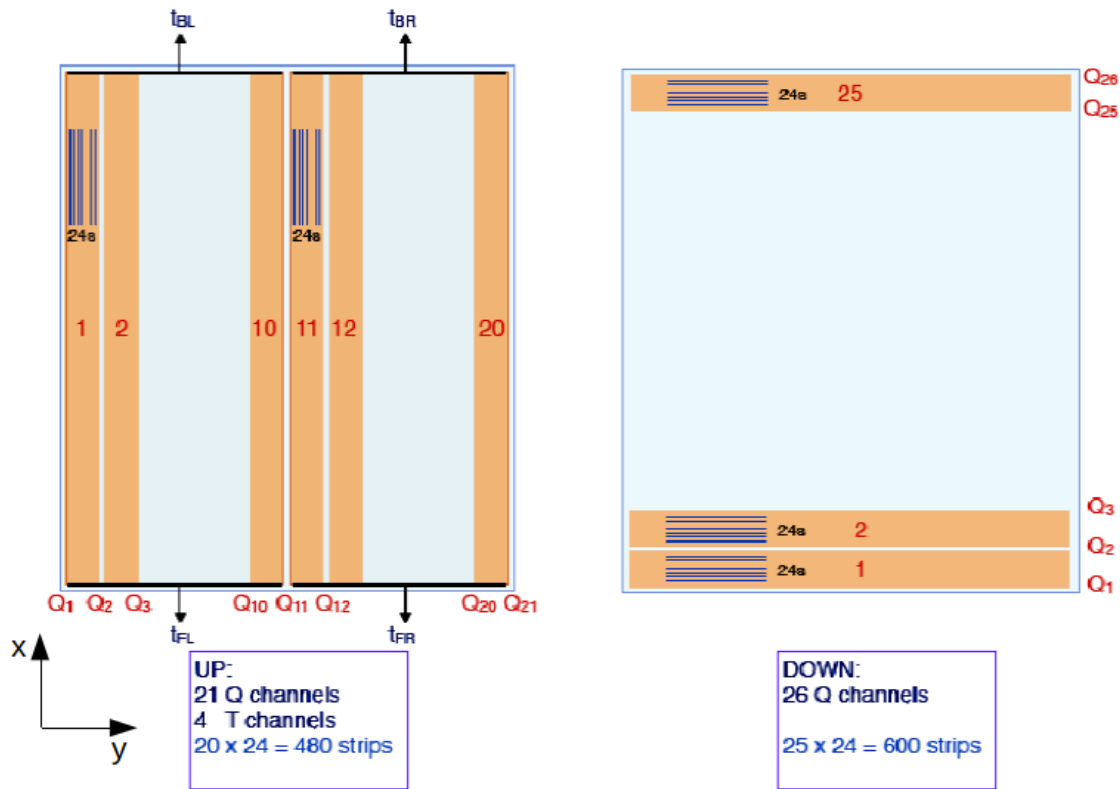
Fig. 3.5 Distribution of the detection wires and strips in a *muTT* detector. There are two layers of strips that are perpendicular to each other. The *up* set of strips measures charge and time, and the *down* set measures only charge.

In the figure, the strips are coloured in orange and each strip is formed by a set of 24 wires. The *up* layer consists on 20 strips and the *down* has 25.

The relevant physical information is obtained from the measured charge and time. When a particle crosses the detector it interacts with the gas and induces a signal in our strips. This information is grouped in channels.

Each plane is divided in two sides: left-right and front-left. As it is shown in Fig.3.5 the up layer collects the temporal information depending on the region where the particle was detected. Therefore, we can have four different time channels: front-left, front-right, back-left, and back-right. This time information is only obtained in the up layer. On the other hand, the charge is measured at both sides of the interaction strip. In the Fig.3.5 are labelled as $Q_1, Q_2, \ldots$ As the up layer has 20 strips, the total charge channels are 21; and in the down layer are 26. As a result, we have 4 time channels and $21 + 26$ charge channels.

The information collected at each charge readout channel is processed to provide the hit coordinates (x, y) based on the centre of mass of the charge distribution. The algorithm that

processes the raw information of charge and time and gives a final (x, y, t) hit coordinate was done by A. Blanco at LIP-Coimbra.

A further optimisation in processing the raw data can be done by the use of TimTrack algorithm (Chapter 2). We can propose a model for the raw data at each plane:

$$
\begin{aligned}
T_{f,m} &= t + x \cdot S \\
T_{b,m} &= t + (L - x) \cdot S + \tau \\
Q_{j,m} &= k \cdot (j \cdot \Delta y - y) \\
Q_{j+1,m} &= k \cdot [y - (j - 1) \cdot \Delta y] \\
Q_{i,m} &= k \cdot (i \cdot \Delta x - x) \\
Q_{i+1,m} &= k \cdot [x - (i - 1) \cdot \Delta x]
\end{aligned}
$$

where in these equations i and j are the number of strip where the particle interacted in the up and down layer, $T_{f,m}$ and $T_{b,m}$ are the models for the front and back times, $Q_{j,m}$ and $Q_{j+1,m}$ are the models for the charges left and right at the up layer, $Q_{i,m}$ and $Q_{i+1,m}$ are the charges left and right at the down layer, L is the length of a strip in the up layer, τ is a constant related to the time shift between both sides, and Δx, Δy are the widths of the strips on the up and down layers. As a result, we obtain the parameters:

$$
\mathbf{s} = (x, y, t, k) \tag{3.3}
$$

where k is a constant characteristic of the detector an x, y, t is the hit. Therefore, with the TimTrack method the internal parameters of the RPC design can be optimised.

For an experimental set-up formed by a lot of planes it has to be taken into account the propagation of the particle. As a consequence, in the equations should be included the dynamical parameters $X_0, X', \dots$

Chapter 4

General cosmic ray simulations

4.1 Introduction

A computer simulation is a process that can be defined as the run of a software program by a machine (or several machines as a node) that tries to reproduce an abstract model of a particular physical system [61]. This mathematical model is proposed to be a faithful representation of a real-world situation.

A simulation is therefore a controlled environment in the sense that we can set, adjust or decide the values of the parameters that play a key role in the system. A simulation should be as realistic as possible in the sense that have to describe its equivalent in the real-world scenario. This is sometimes very difficult to achieve and in many cases it is directly impossible due to the large number of parameters involved in a certain problem. In other cases, the model may include equations that are not possible to solve with a computational method or algorithm, or in other cases the model may have very restrictive boundary conditions and no representative solution can be found.

One possible way to solve these problems is to find a simplified physical system with a similar behaviour as the real one. This reduction would help us to find out the relationship between the system variables and to write a computer code that describes them [64]. As it is shown in Fig.4.1 simulations can be used to improve our models by comparison to experimental results. If the results of the simulations are significantly different, it might be an indicator that our model is not describing the real system correctly. Therefore we have to modify our model by changing the equations or the boundary conditions.

Computer simulations are widely used in science, industry or economy. For example, simulations in physics are common in meteorology, fluid mechanics or electrodynamics, among others, and are essential in the construction of mathematical models of natural systems.
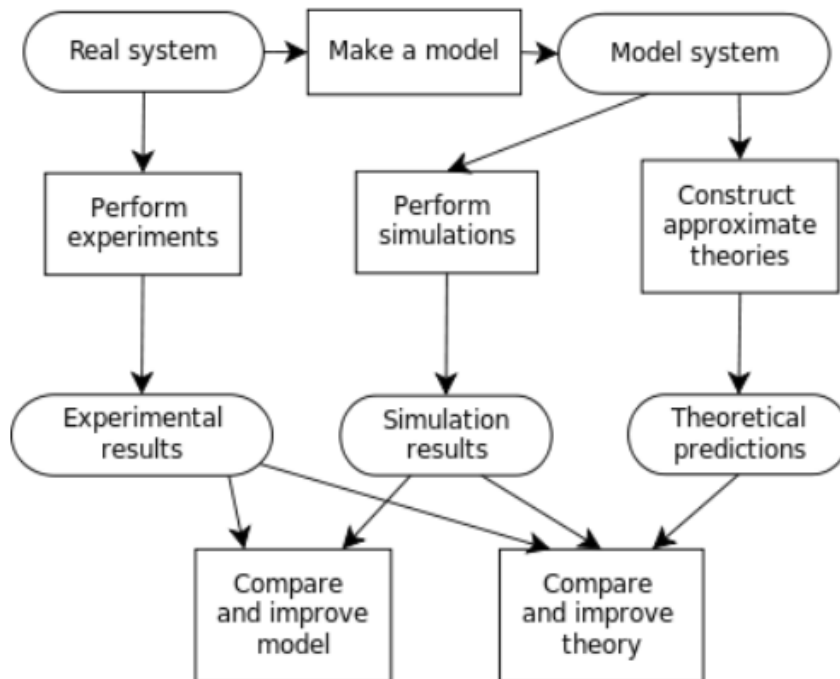
Fig. 4.1 Schematic representation of computer simulation process. Figure taken from the lectures of Álvarez [61].

Nuclear and particle physics experiments involve several types of detectors and electronics. A lot of elements must interact and work together, and therefore there are many experimental details, effects and features that must be to taken into account. Even small size experiments require a previous simulated analysis that provides a good check of our models, detectors or observables.

The Monte Carlo method [40] supposed an improvement in the study of different physical problems by the use of a new mathematical algorithm based on probabilities. A Monte Carlo simulation is, in essence, a statistical approach to a problem by the use of integro-differential equations and random numbers. Roughly speaking, we can say that a Monte Carlo simulation is like repeating the same experiment a large number of times but including random effects at each step. It generates an initial state, applies our model a large number of times to that initial state of the system and then evaluates the final stage. By the law of large numbers it is possible to give a probabilistic interpretation of the variables and then connect them to the parameters of the real system model.

Since, in principle, every stochastic process can be solved with the Monte Carlo method, it is widely used in many physical branches such as meteorology or particle physics. As an example, in 1950 are published the first Monte Carlo simulations for electron transport, that

are of special importance in medical physics. We only make a brief remark about the works of B. Wilson [50] that is one of the pioneers in tumour radiotherapy.

Monte Carlo simulations are also important in particle physics experiments. Nowadays, the experimental setups in high energy physics require a lot of manpower and infrastructures. It is therefore important to perform a previous study on the design of the facilities. These studies can cover from the behaviour of our detectors to the influence of the building in the experiment.

In the other hand, simulations can be used beyond the structural analysis. There are many situations in which we are interested on a particular theory or maybe we are studying a physical observable. The previous analysis of such variables are made by the use of a proper simulation.

These analysis seem to be necessary since they provide previous information of the experiment before it is set, and they help to understand the physics behind it.

A typical simulation in physics can involve millions of events and the only way to perform it is by the use of computers. Computer simulations are nowadays one of the best tools that we have to control all the information related to experiments in physics. The improvements in hardware and software technologies allow us to perform simulations by the use of a laptop or desk computer in a reasonable period of time. For example, a Monte Carlo simulation of a million muons hitting a material can be done and analysed in approximately ten minutes.

There are a lot of computer codes in the market that are capable of performing Monte Carlo simulations. Usually, they are written in many different languages such as C++, Python or Fortran. Beside the simulation of data we need an analysis code to extract the information of interest.

In this thesis we adopted the EnsarRoot code [26] as the standard tool for simulation and analysis. As we will explain later, EnsarRoot is an official flavour of the FairRoot framework that provides a collection of software to simulate the detectors, the experimental setup, run a Monte Carlo simulation and perform an analysis of the results. Next section is devoted to the description of the internal structure of the code and the improvements that were done during this thesis.

The implementation of the EnsarRoot simulation and analysis software was made taking as a reference two different experiments. Both of them use detectors of the Trasgo family, but they operate in different ways. Along this chapter we will present the implementations in the TRAGALDABAS [4] and *muTT* detectors [62]. In particular, it will be explained how we simulated their structure and the way that simulated data are managed.

The physics behind our simulations is directly related to the interaction of the secondary cosmic radiation with matter. This radiation is the result of interaction of particles at Earth

atmosphere. After these interactions there are several processes that compete between them, such as Bremsstrahlung and pair production. As a result, it can be said that the cosmic radiation arriving to our detector has two main components: in one hand we have muons and in the other electrons-photons. For that reason, in this chapter we will describe the multiple Coulomb scattering and the creation of electromagnetic showers as two separated pieces, but they are part of the same radiation. The observables of interest are therefore related to the scattering angle of muons due to the presence of a heavy material.

4.2 The EnsarRoot simulation and analysis framework

One of the most common software tools used to simulate geometrical structures (detectors) and then perform Monte Carlo simulations is the Geant package developed at CERN [33, 34]. The results of a Monte Carlo simulation are stored in files that usually are of several gigabytes. Thus, it is necessary to use a proper software that it is able to analyse big files of data with an efficient computer memory management.

There are also a lot of software tools designed and oriented to the analysis of great volumes of data. The ROOT package [25] is an object oriented framework developed at CERN that provides a good analysis environment. It is written in C++ and one of the advantages of Root, since it is object oriented, is that it can encapsulate the information in a more efficient way. With Root it is possible to manage huge amount of data and with them create nice editable histograms. The TClonesArray data type permit to store and manage a large number of objects optimising the use of computer memory.

It can be seen that every simulation process has two main parts: one part is devoted to generate the simulated data and the other is related to the analysis of those data. The FairRoot framework [21] was created as an interface between the information provided by the Geant simulation package and the Root analysis tools. The idea is that the user has access to the geometry of the experiment, can adjust and set the simulation parameters and at the end of the process obtains a file that can be understood by Root.

The reader could now ask why it is more convenient to use a new structure such as FairRoot and not a pure Geant simulation combined with Root (or another analysis tool) instead. FairRoot is a software framework. This means that have access to a collection of services and we can use them anytime we need them. In this case, there is a service that creates a geometry, other that sets the simulation parameters, and others.

As an example, consider a RPC detector. It is always possible, of course, to implement the geometry using only the Geant package: we build an Aluminium box, we design the gaps inside, we fill them with gas and we can add more elements if necessary, such as foam or

methacrylate. After that, we run a Monte Carlo algorithm. The Geant simulation engines include the information about the cross sections of radiation with matter. This means that the particles are leaving information during the whole event. In our particular example, the information is provided by the amount of energy accumulated in our RPC, according to the physical information inside the Geant algorithm. This information it is not the real world situation, because at the end of the day what we detect is an electric signal lately translated to position, time or other magnitudes of interest. All this information must to be structured because it might be that we are not interested on every single interaction with the active volume. This process is made by FairRoot and the user can control the whole process without any external import-export executable.

The simulations are run in FairRoot by the execution of a task list file. In this file, that the user can modify and set the parameters, choose a particle generator, select a particular geometry or perform reconstruction of the trajectories.

Since the beginning of FairRoot, in 2004, many experiments have adapted the framework to their particular detectors and geometries. For example, Panda, R3B or CBM decided to join the FairRoot structure and implement their own geometries, detectors and data management. All these flavours share the main structure of the framework but the simulated data are treated in a different way.

Other official flavour of FairRoot is EnsarRoot. As the other flavours, it shares the main structure of the code, but readapted to other experiments. These new code includes the representation of our experimental setups. The EnsarRooot code, since is a direct legacy of the FairRoot structure, allows to perform this adaptations in a simple way. This fact reduces the programming time and effort drastically, since it is based on already tested codes.

EnsarRoot was chosen as our standard framework because the configuration of the TRAGALDABAS detector was implemented as a module inside its structure. In the next subsection we will explain how a simulation with EnsarRoot is done. We can summarise the ingredients that we need to combine in a simulation process:

1. The **geometry**. The description of the detectors, the detection materials, the disposition and other components must be specified.

2. A **script** to run the simulation. This script includes all the tasks that must be performed in the process. The type of particles, how must be generated and the rest of specifications must be contained in this script.

3. We need to structure the **data**. The results of the simulation must be organised.

4. Provide a **real description** of the simulated data. Usually, the simulated data are not a real-world description. In this stage we need to transform the simulated data into simulated measurements.

5. Proceed to the **analysis** of the data.

In the next subsection it will be explained how the items above are defined inside the EnsarRoot framework. As it was mentioned, the Trasgo families of detectors are taken as standard for our simulations. We will describe the details of the code implementations.

4.2.1   EnsarRoot structure

Before explaining the details of the simulations we will give some general considerations about the internal structure of the EnsarRoot code. After this introduction and to illustrate how the ingredients of the simulations combine to each other, we will describe how a detector of the Trasgo family is simulated. A particular case of these detectors is the RPC plane detector used in the TRAGALDABAS detector at USC. The EnsarRoot code provides the classes that enable us to create the simulate the detector features and perform a further analysis of the simulations.
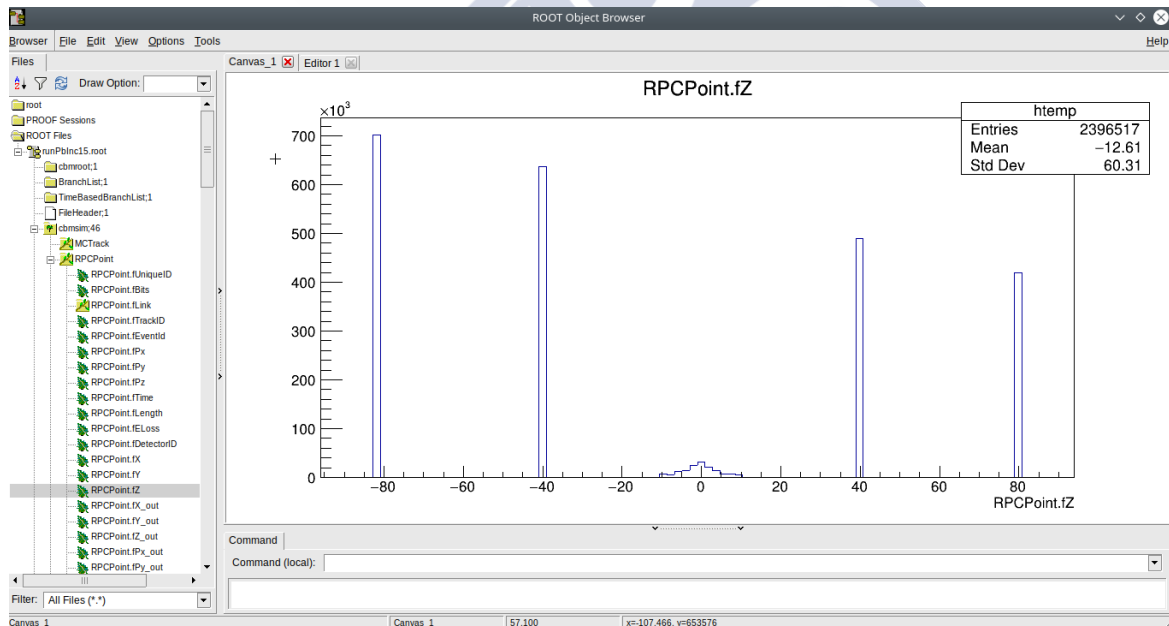
Fig. 4.2 Example of TTree produced after a simulation in EnsarRoot. All the variables resulting from the Monte Carlo simulation are structured in different data levels.

Since the EnsarRoot structure uses all the FairRoot base libraries, the required software dependences can be installed almost automatically. This means that all the classes distributed

with the Root framework can be used in a standard way by default. For example, we can group our data levels using TClonesArray or TTree structures of Root, which reduces the memory usage and execution time of the process. We enumerate the general characteristics of EnsarRoot:

1. The simulations and analysis are performed by scripts. This means that it is not necessary to create executables, everything can be done running a script with Root.

2. The TGeo class from Root is used to create the geometry of our detectors. The set up is encapsulated in a root file and it is called by the execution script.

3. Geant3 and Geant4 simulation engines can be used. Inside EnsarRoot exists the interface that connects the simulated events with the analysis tools: VMC (Virtual Monte Carlo) It is possible to use Ascii generators as well. The events are stored in a text file and are read by the code to generate the tracks.

4. Simulation results can be structured by the use of TClonesArray. This is particularly useful since we can simulate millions of events. It is convenient to store all this information in a binary format and not in a text file. The variables provided by the simulation (used by Geant, Ascii file or with other method as generator engine) are presented in a Root file and displayed in a TTree, as shown in Fig.4.2.

5. All the events can be visualised using the TEve viewer, providing a modern and interactive interface based on OpenGL libraries. The viewer displays the tracks of the particles, detectors and the information of the particle, such as energy and particle type. The figures shown in this chapter that are related to the simulated geometry are produced by the TEve viewer.

As it can be seen, the EnsarRoot code is nothing less and nothing more than an interface between the simulated data and its translation to a Root file, where the data are structured and presented in a more efficient way by the use of trees.

At this point, we will describe the characteristics of the TRAGALDABAS detector inside the EnsarRoot structure. It was mentioned that the TRAGALDABAS classes are distributed with EnsarRoot as an independent module.

4.2.2 TRAGALDABAS detector as a module inside the EnsarRoot structure

The EnsarRoot code is ideal for small and medium scale particle physics experiments. TRAGALDABAS [4] is a high resolution muon telescope devoted to the understanding of the

cosmic ray showers arriving to the Earth surface. It is located at the University of Santiago de Compostela and is based on RPC detectors of the Trasgo family. TRAGALDABAS is included as a part of the EnsarRoot framework as an independent module. This means that all the classes and other source code involving the experiment is located in a separated folder. Therefore, TRAGALDABAS has its own software libraries and specific paths to them, without any other dependence.

Simulation of the geometry

A cross sectional view of the RPC detectors used in the TRAGALDABAS detector is shown in Fig.4.3. The simulation of the RPC plane is made by the use of the TGeo geometry classes distributed with Root. As a result, the final representation of the geometry is encapsulated in a root file.
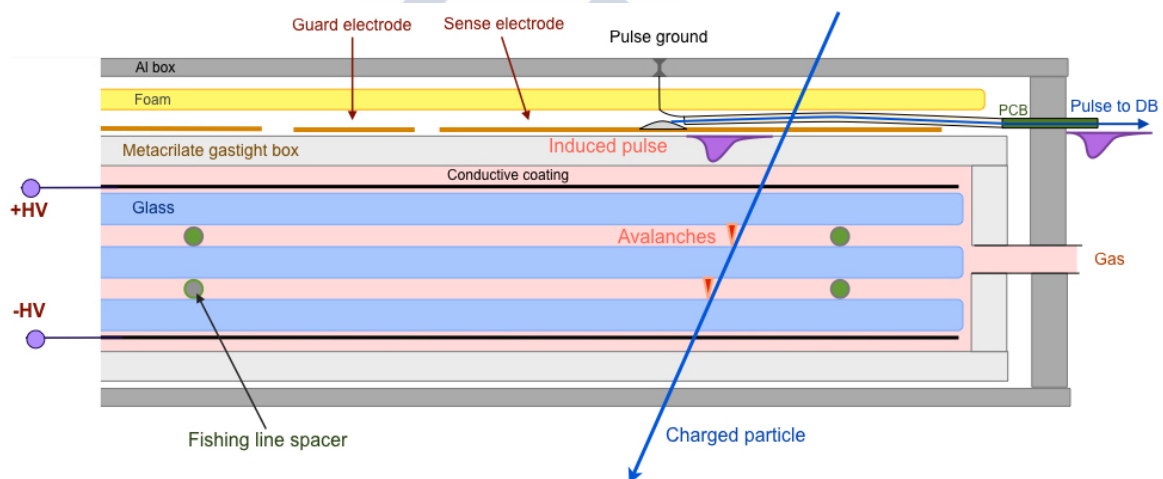
Fig. 4.3 Cross sectional view of the RPC used in TRAGALDABAS. It has two gas gaps that are the sensitive volume. Induced pulses are read out by the Front End Electronics daughter-board (DBS).

In the construction of the code that represents the geometry we are not interested on the physics of the detector, but only on the forms and materials of its components. All the elements are set inside an aluminium box. This is considered by Root as the *mother volume* of the detector, and the rest of volumes are embed inside the mother volume. One of the advantages of using the TGeo classes is that typical geometrical operations can be applied to the volumes. A volume can be easily rotated, translated and copied by the use of the methods implemented in the class. This is a of particular interest. Imagine that we need to set in our experiment the same detector several times. For example, in TRAGALDABAS we need to place four identical RPCs. It is not necessary to code the same thing four times: it is done

only once and then we copy it as times are necessary. The code sets a label to each copy of the volume and this number is an unique identifier of the detector. When a particle interacts with a particular RPC, there is no possible confusion to the code, since the identifier contains the information of the plane number.

In Appendix A we provide the code that simulates a RPC detector as the used in TRA-GALDABAS. As it can be seen, several material media are needed to construct the RPC: aluminium, foam, methacrylate, and several more. The physical properties of these materials, such as atomic mass and number, density or radiation length, are stored in a data file written in a format that the Root geometry classes can understand. For example, the material labelled as "iron" is codified as

```
iron  1  55.847  26.  7.87
      1  1  20.   .001
```

following the notation of the FairGeoMedium class. The simulation of the RPC using this code can be shown in Fig.4.4. The external aluminium box is visible in red but the rest of materials can not be seen with the viewer.
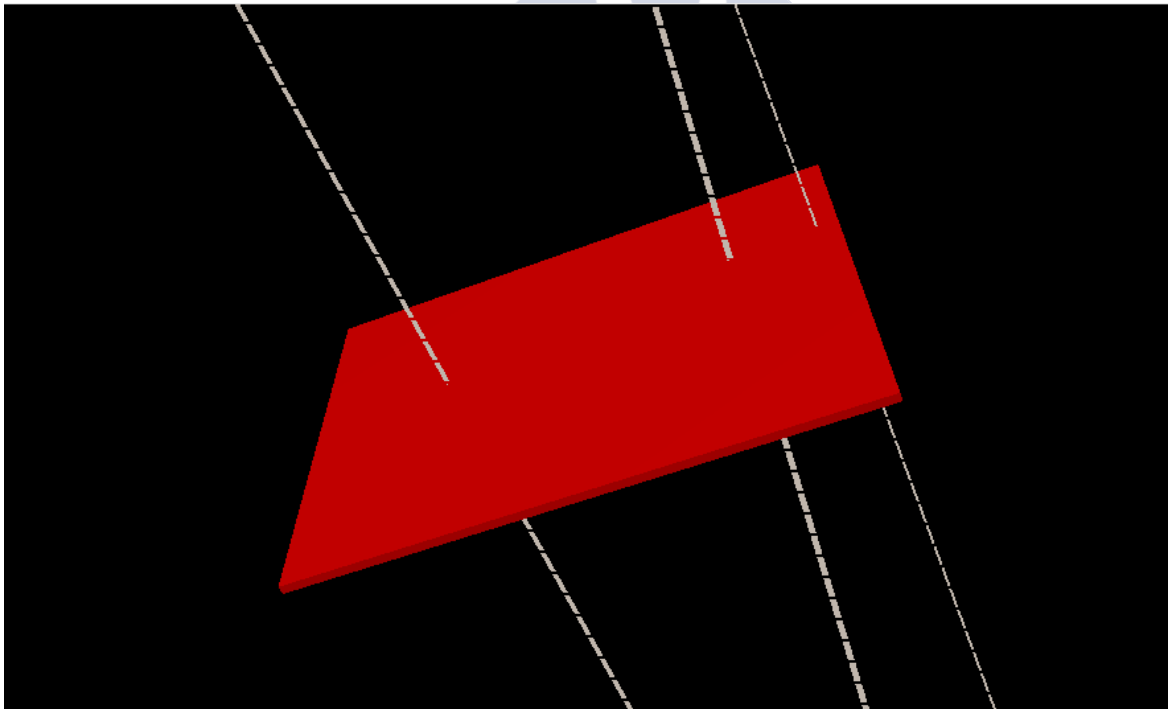
Fig. 4.4 Simulation of a RPC detector with EnsarRoot. In this figure, three muons are passing through the detector, leaving information in the two gas gaps inside the RPC.

When a particle arrives to the detector, it interacts with the different materials according to their cross sections. Usually, we are not interested on the interaction with all materials but

only the interaction with a particular medium. In TRAGALDABAS, a signal is produced by the interaction of a particle with R134a gas. This medium in the simulation is called *active volume*. In these particular RPCs we have two gaps filled with freon gas. The particle leaves information in both gaps and we need to process it to simulate the final measurement. This allows us to introduce the concept of data level. This is a crucial point in the simulation process because they contain the information that allows us to make a good real-world description of the experiment.

Data levels

A *data level* can be defined as a particular stage in the process of the information. When a particle enters a material medium there is a certain probability that an interaction between them can be produced. The Geant package contains the information about the interaction probability with material media providing the cross sections. This information provided by Geant is different to the real-world representation because it does not represent a physical measurement of our detector. A more realistic information must be formed taking the Geant data as input. These two representations are an example of data levels. In TRAGALDABAS we define them as *point* and *hit* data levels respectively.

1. The **point** level of data corresponds to the perfect information that is given by our simulation engine. In our example, Geant is told to provide information every time that a particle interacts with our active volume. This information is not real in the sense that is not what we detect in our experiment.

2. A more realistic description is given by the **hit** level of data. The information provided by the simulation must to be translated to our observed coordinates. In the case of a RPC, position and time coordinates. This process is called digitization and is different depending on the detector.

As it was mentioned before, the RPC chamber has two gaps filled with gas, and inside this medium the electronic avalanches take place. As a result, each event consists on a collection of interactions with the active volume that need to be processed. This raw information is the *point* level of data. We can say that because the information is not processed yet and it does not represent a real-world detection. The information provided by the simulation engine is related to position, momentum, energy loss or time. Our task is now to simulate the detector response to that particular interaction. In real life situations this process is not perfect. It might happen that an interaction with our active volume does not produce a signal, or maybe a single avalanche can induce signal in two pads. These features of the detector must to be
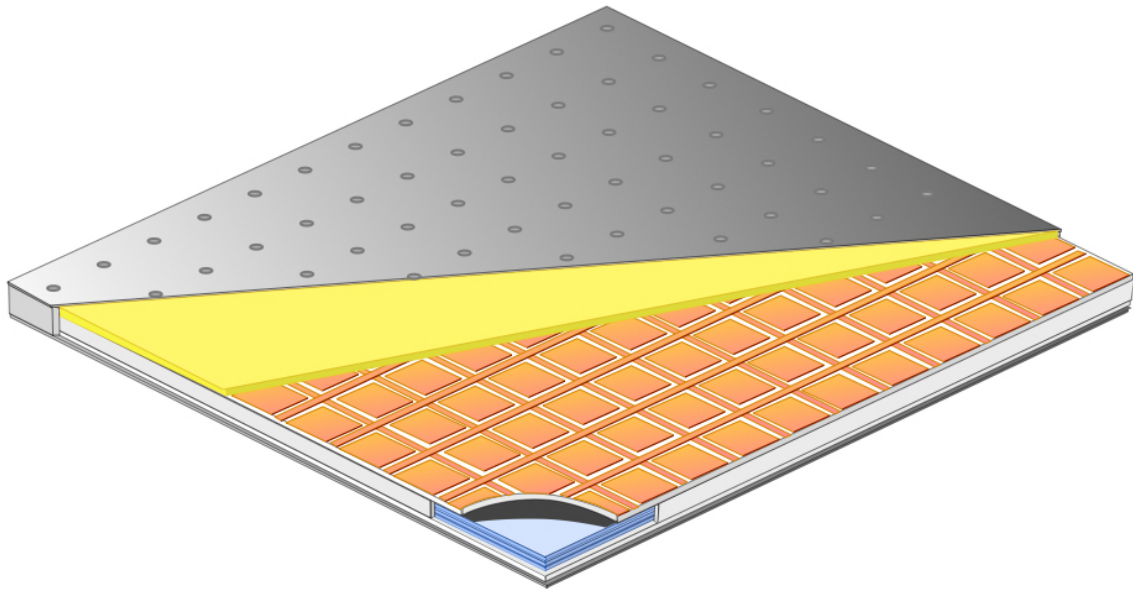
Fig. 4.5 Representation of a RPC detector used in TRAGALDABAS detector. The detection area is divided in pads.

simulated to provide a representation as realistic as possible. The improvement of the hit level of data for the TRAGALDABAS detector was implemented during this thesis. To do this, we had to take into account the following issues:

1. Since we have two freon gaps, we have two active volumes. Therefore we have two regions in which point information can be left.

2. The detector has a pad structure. This means that the avalanches produced inside the gas induce a pulse in the electrodes placed on the pads. The signal is then processed and as a result are given a position and a time.

3. We can not determine the exact position where the particle entered a RPC chamber. Since the avalanches produce a signal in a pad, the position coordinates are given by the coordinates of the centre of the pad where the signal occurred.

4. The cells are not in contact with each other because there are guard strips that avoid the signal to travel from one cell to its neighbours, as shown in Fig.4.6. Therefore, we have to know if the point is produced in the pad active zone.

The implementation of both data levels are done for TRAGALDABAS in the classes **TraRPCPoint.h** and **TraRPCHit.h** inside the TRAGALDABAS module distributed with EnsarRoot. Both data levels are stored and structured using the classes distributed by Root:

Zero position

CELDA 5-6

(- 63 mm, 61.5 mm)

CELDA 5-7

(63 mm, 61.5 mm)

(0,0)

X

CELDA 6-6

(-63 mm,-61.5 mm)

CELDA 6-7

111 mm

(63 mm,-61.5 mm)

116 mm
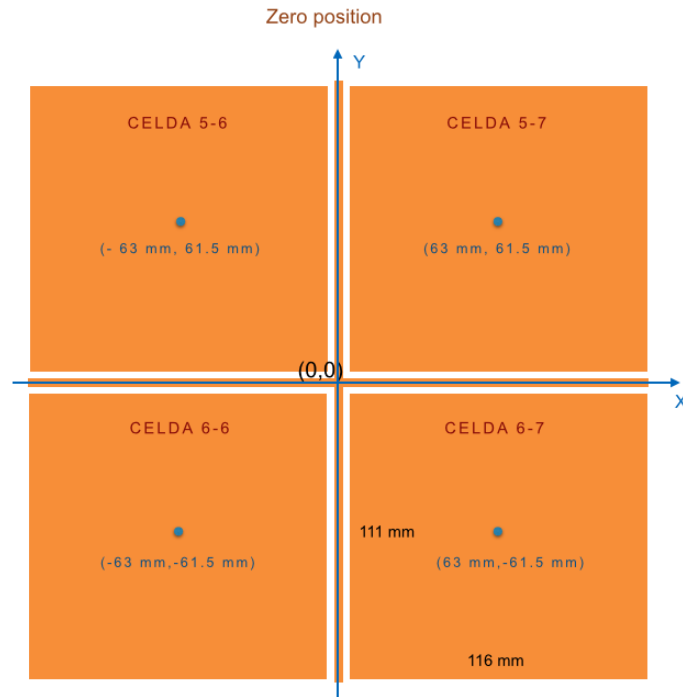
Fig. 4.6 Cell dimensions of the RPC TRAGALDABAS detector. There is a guard strip between pads that isolates them and tries to avoid induced signals between them.

- Point data level: the simulated interactions with the gas are stored in a TClonesArray. Every event has therefore a collection of all the points produced by the interaction of the incident particle. The user has always access to the position, momentum and the detector in which the interaction took place. This information is stored in a computer word and is a unique identifier of the event.

- Hit data level: if a point is found inside the active volume this not necessary imply that it will produce a hit. We assumed that a cell is fired when the charge that the particle left in the medium (it gives an idea of its energy) is bigger than a certain threshold. Additionally we can argue if the track of the particle entered, passed or disappeared inside an active zone. In those cases we process the points to get a hit and is also stored in a TClonesArray. A hit encapsulates the information concerning the centre of the cell, time and charge deposited.

Additional information is also encapsulated in the hit level of data. As we know, the TRAGALDABAS cell mapping is presented in Fig.4.7. It might be interesting to store, for a further analysis, the plane number, cell centre, row and column of the hits. This is processed by the use of the variable RPCId. The code takes the generated spatial information,

Fig. 4.7 The pads of the TRAGALDABAS RPCs are labelled according to their row and column. The origin is taken in the centre of the chamber.

fPosIn.X() and fPosIn.Y(), and makes a correspondence between them a the cell where the particle hit the plane:

```
RPCId = cpKIV*4096 +
        int((fPosIn.X()+6*12.6)/12.6)*64 +
        int((-fPosIn.Y()+5*12.1)/12.1);
Row = (RPCId & 15) +1;
Column = ((RPCId & 960) >> 6)+1;
Xcenter = 69.3-(12-Column)*12.6;
Ycenter = 54.45-(Row-1)*12.10;
```

As it can be seen, the variable RPCId is formed taking into account the dimensions of the cell. We implemented methods in the TraRPCHit class to allow the user to have access to the row, column and the coordinates of the centre of the cell where the signal was produced.

These point and hit are the two basic data levels that can be found in the TRAGALD-ABAS module. However, in the latest version of EnsarRoot we implemented a new type of data level: the **saeta** level of data. Based on the TimTrack algorithm (explained in Chapter 2) we transform the hits processed in one event and then we reconstruct the possible trajectories of the particles passing through our detector. The implementation was done in the **TraRPC-Saeta.h** class inside the TRAGALDABAS module. As we will explain later, this saeta object

contains the parameters related to the propagation of the particle, such as its initial position, slope or velocity. These saeta objects are stored in a TClonesArray structure, exactly the same way as the points and hits are stored.

Following the same method as before, more data levels can be created. For example, a new object called *shower*. With all the possible saetas we could perform a sketch of the trajectories of all particles of a single cosmic ray shower that passed through our detector. This could provide an initial idea about the cosmic ray shower that arrived to the detector. This algorithm is not still finished and must to be tested with the real data of the TRAGALDABAS detector.

Particle generators

Up to this point, we have simulated the geometry of the detector and how it manages the information provided by the simulated particles. It is necessary to simulate the characteristics of the radiation that will help us to perform studies of the interaction with materials.

The TRAGALDABAS module can use different generators:

1. Box generator: this particle generator simulates events of the so called *particle in box*. The user can adjust the type of particle, the multiplicity, the kinetic energy and angles. Also, the source can be a point or the particles can be generated from a surface.

2. TraBoxGenerator: this generator was implemented as a first cosmic ray simulator for the TRAGALDABAS detector. It generates muons from a random point from a plane at a distance z set by the user.

3. Ascii: the events are written in a text file. The code reads the particle type, energy and angles. After that, the code displays the tracks.

4. CRY: the CRY event generator [37] generates cosmic ray showers from three different elevations. CRY is not included as a native generator. We created an interface that takes the output file of CRY generator and treats it as an Ascii generator adapted for the TRAGALDABAS reference system.

These generators can be used under the Geant3, Geant4 or Fluka [24] Monte Carlo types.

Scripts to control the simulation parameters

The simulation process is run by the execution of a script. This file contains the list of task to be performed. There is another file that contains the parameters of the tasks. Both files are controlled by the user:

1. tragsim.C contains the list of tasks. The user can specify the Monte Carlo type (Geant3, Geant4 or Fluka), the particle generator (box, tragbox,ascii or CRY), the geometry file to be used, the number of events or if a saeta object is created or not.

2. tragall.C provides the parameters of the simulation. For example, we can specify the particle type or angular-energetic range. This file is called by the tragsim.C script.

To start the simulation we only execute the script with Root:

```
root -l -q tragsim.C
```

As a result, a root file is generated and the variables can be displayed in a TTree structure, as shown in Fig.4.2. The TEve viewer opens this output root file and extracts the information of the events. As an example, Fig.4.8 represents a simulation of several muons generated with the Geant4 engine (with a box generator) over the TRAGALDABAS configuration.
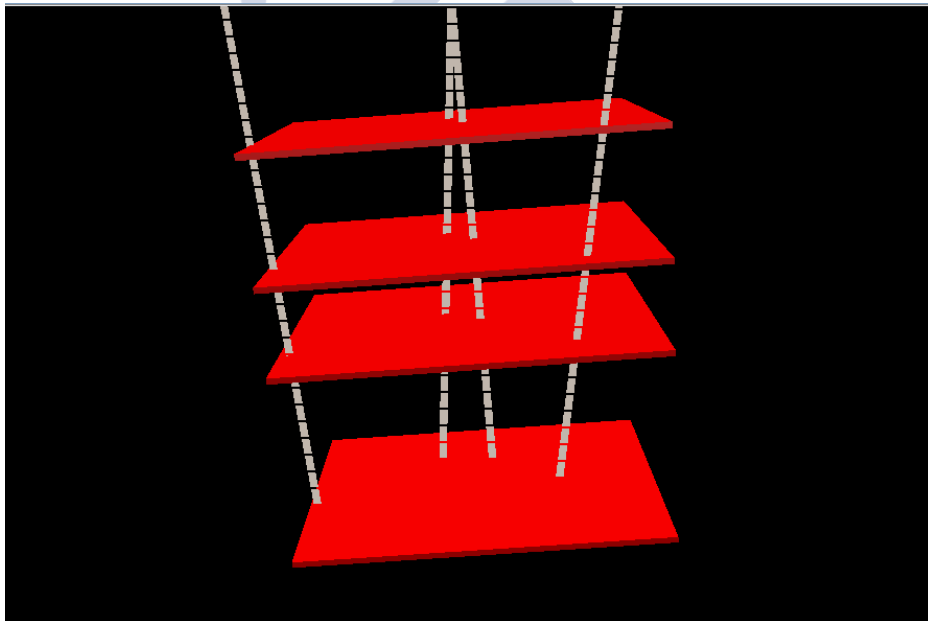
Fig. 4.8 Simulation of the TRAGALDABAS RPC configuration. It consists on four RPC detectors placed in the Particle Physics department at Universidad de Santiago de Compostela.

This root file, therefore, contains all the information of the simulated events structured in a Tree object type. Further analysis of physical observables, such as efficiency, must be performed with another script. This new file opens the Tree, takes its branches and then the physical observables can be created.

4.2.3   Simulations with the *muTT* detector

The *muTT* detector is a new type of RPC chamber designed in LIP-Coimbra with an optimised time and spatial resolutions. This new model of RPC is used as the standard detector in our tomography system.

The design of the *muTT* is completely different to the RPC detectors used in TRAGALD-ABAS. First, the number of gas gaps is different. In TRAGALDABAS the active volume consisted on two gaps filled with freon. In *muTT* we have six gaps. Secondly, the detection method is different. The RPC in TRAGALDABAS was divided in detection pads and the final measurement corresponds to the spatial coordinates of the centre of the cell that was fired. As a contrast, *muTT* consists on two layers of strips per plane. The two layers are set perpendicular to each other, and the spatial coordinates are measured with the time-of-flight of the signal along the strip. These differences must to be taken into account to implement a suitable code to simulate the physics of the new *muTT*.

As it will be explained at the end this chapter, before the final tomography system was set, we performed several tests of the detectors in a simplified structure. This configuration consisted on four *muTT* planes as shown in Fig.4.9 and served as a test field for our calibration and reconstruction algorithms. By now, we are only concerned about the simulation of the geometry of the detector. Contrary to the TRAGALDABAS detector, *muTT* is not created as a module of the EnsarRoot framework. The geometry of the *muTT* was programmed as a simplified version of the TRAGALDABAS RPC. Therefore, we reused the code of the Appendix A to create a new code for the *muTT*.

The external structure of the *muTT* is the same as TRAGALDABAS: the aluminium box is the mother volume and the rest of materials are embed inside it. The main difference is in the number of gas gaps. The *muTT* geometry contains only one gas gap instead of six. This simplification is motivated by the fact that, since the *muTT* uses detection strips, the position of the interaction can be known. In this approximation, no further digitization of the Monte Carlo information is performed, and therefore we work only in the *point* data level.

This is another advantage of the EnsarRoot framework: it is not necessary to create new code for this detector. We only needed to simplify the already built geometry and work in an appropriate data level. No new data level has to be created and the same scripts inside TRAGALDABAS are reused.

4.3   Multiple Coulomb scattering

Every material has a high amount of electrons that are located in the electron cloud of the atoms. They are light, negative charged particles that in general are weakly bounded
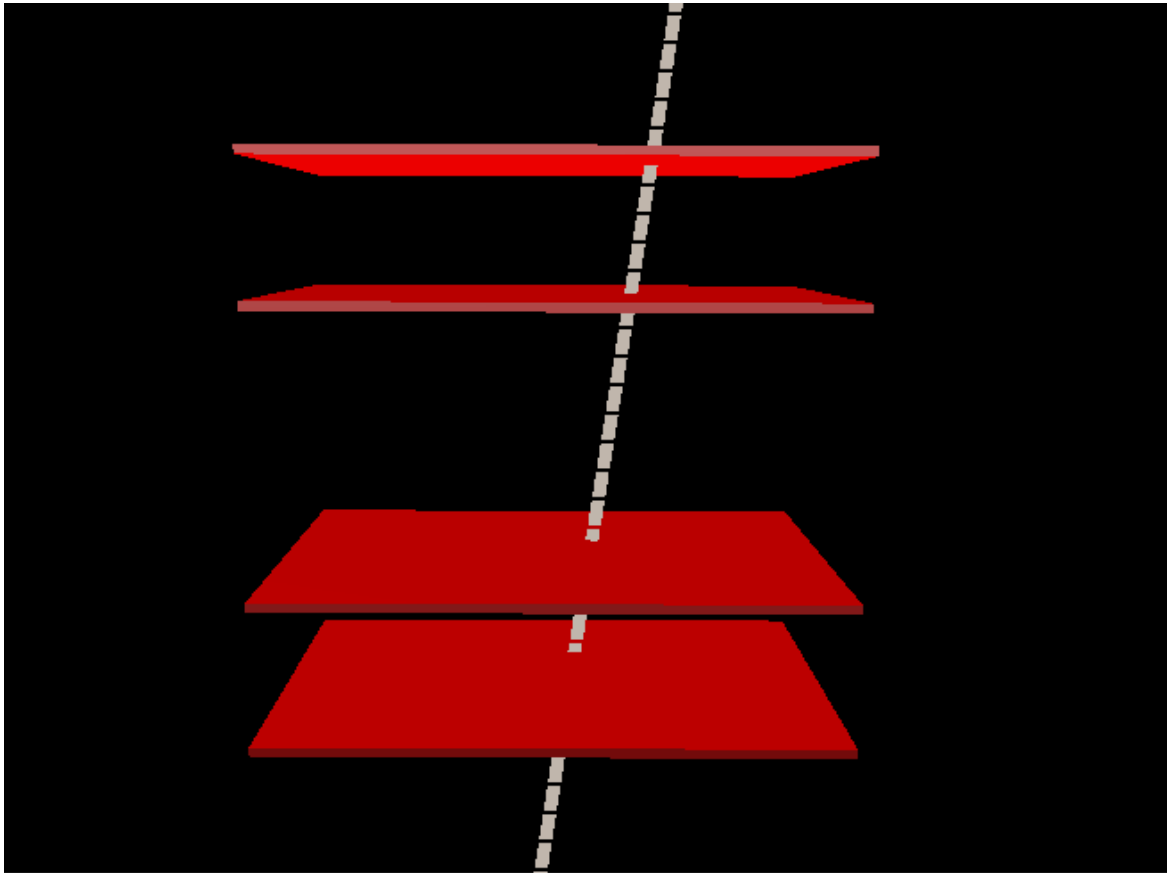
Fig. 4.9 Simulation of the *MuTT* detector in EnsarRoot.

to the atoms. As a consequence, every charged particle passing through the material will experience an electromagnetic interaction between the charge of the particle and the electronic medium. The particle is then suffering the effect of numerous infinitesimal forces that cause a deflection in its trajectory, pushing the particle infinitesimally along its path. This effect is called multiple scattering. This is a highly random process and can be very different for two particles incoming with same energies and angles. An schematic overview of the multiple scattering process can be seen in Fig.4.10 As it can be seen in the figure, a particle meets the outer surface of a material and starts to suffering infinitesimal changes in its trajectory because of the Coulomb interaction between the charges. The process ends when the particle is out of the material.

In this representation we can assume that the energy transfer to the media is negligible since the mass of the electrons are much smaller than the one of any incoming particle (except the case of the electrons). This is the reason why the changes in the trajectory of the particle is usually infinitesimal.
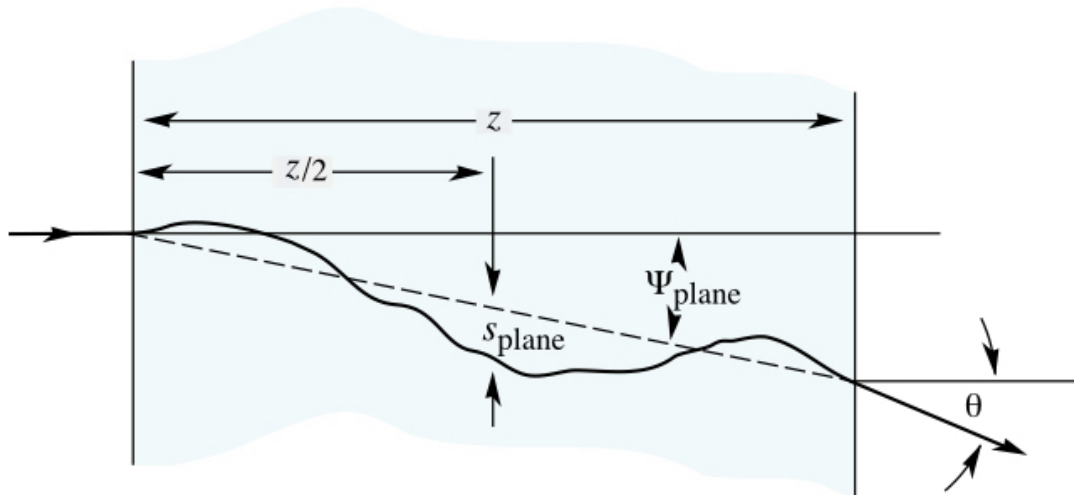
Fig. 4.10 Multiple scattering representation taken from [7]. A particles arrives to the surface of a material and experiences infinitesimal changes in its trajectory inside the material.

According to this description, we would expect that the probability distribution of the deflection angle, θ, would be Gaussian. However, in a real life situation we find out that the material can contain very bounded electrons and therefore the interaction with the incident particle is not infinitesimal. This is the reason why high deflection angles can be found. As a result, there are tails in the distribution, that correspond to anomalous high deflection angles [22, 7].

In the hypothesis that the medium is thin enough and the tails of the angular distributions are not significant, the central part of the probability distribution can be assumed to follow a Gaussian distribution.

In the literature it is common to find the distribution of the projection of the total dispersion angle. If we call θ the total angle, its projection θ_0 in any transversal plane is corrected by a geometrical factor:

$$\theta_0 = \theta/\sqrt{2}$$

The projected angle θ_0 follows a Gaussian distribution of width

$$\theta_0 = \frac{13.6\,\text{MeV}}{\beta c p} z \sqrt{x/X_0}\left[1 + 0.038\ln\left(x/X_0\right)\right] \tag{4.1}$$

in the central region. In this equation, p, βc and z are the momentum of the particle, the velocity and the charge number of the incident particle. Also the angle depends on the quantity X_0, called radiation length. This number is characteristic of every material and

can be considered as a natural unit for every pure electromagnetic processes that take place inside a certain material. It will also appear in the study of electromagnetic cascades due to Bremsstrahlung electrons since it represents the mean distance that an electron travels until it loses all but $1/e$ of its energy.

For each material of atomic number Z the radiation length can be approximated to

$$X_0 = \frac{716.4\,A}{Z(Z+1)\ln(287/\sqrt{Z})}\ g/cm^2 \tag{4.2}$$

where A is the mass number and Z is the atomic number. This radiation length is tabulated and it can be seen that decreases as the atomic number Z increases. It is also common to find the radiation length in units of mass width (g/cm^2). This is denoted in literature as ε and is particularly useful because in this units the radiation length is less dependent on the density and we have a better guess about the number of scattering centres per mass unit. This is made just multiplying the linear thickness (L) times density, expressing the X_0 in $g \cdot cm^{-2}$.

The behaviour of the scattering angle and radiation length are the basis of the muon tomography of heavy materials by the use of cosmic muons. Our simulations concern about the behaviour of this angle and it is the reason why we propose new observables related with it.

4.4   Electromagnetic showers

Muons, when they pass trough a material, are losing progressively their energy until they finally decay in lighter particles (namely: an electron and two neutrinos) and gammas. On another hand, high energy electrons and high energy photons produce new electrons, positrons and photons when they interact with the material. These new electrons, positrons and photons form what is called an *electromagnetic shower* and are the result of two different processes:

1. High energetic electrons loss part of their energy by the emission of a photon when they cross the matter or are deflected by another charged particle. This is due to the Bremsstrahlung or *braking radiation*.

2. Photons with an energy bigger than 1.022 MeV, in presence of matter, may produce an electron-positron pair.

An electromagnetic shower can be sketched in Fig.4.11.

It seems to be reasonable that each stage of the electromagnetic shower takes place, on a mean value, after a radiation length X_0 is travelled. Also it is usually assumed that the energy
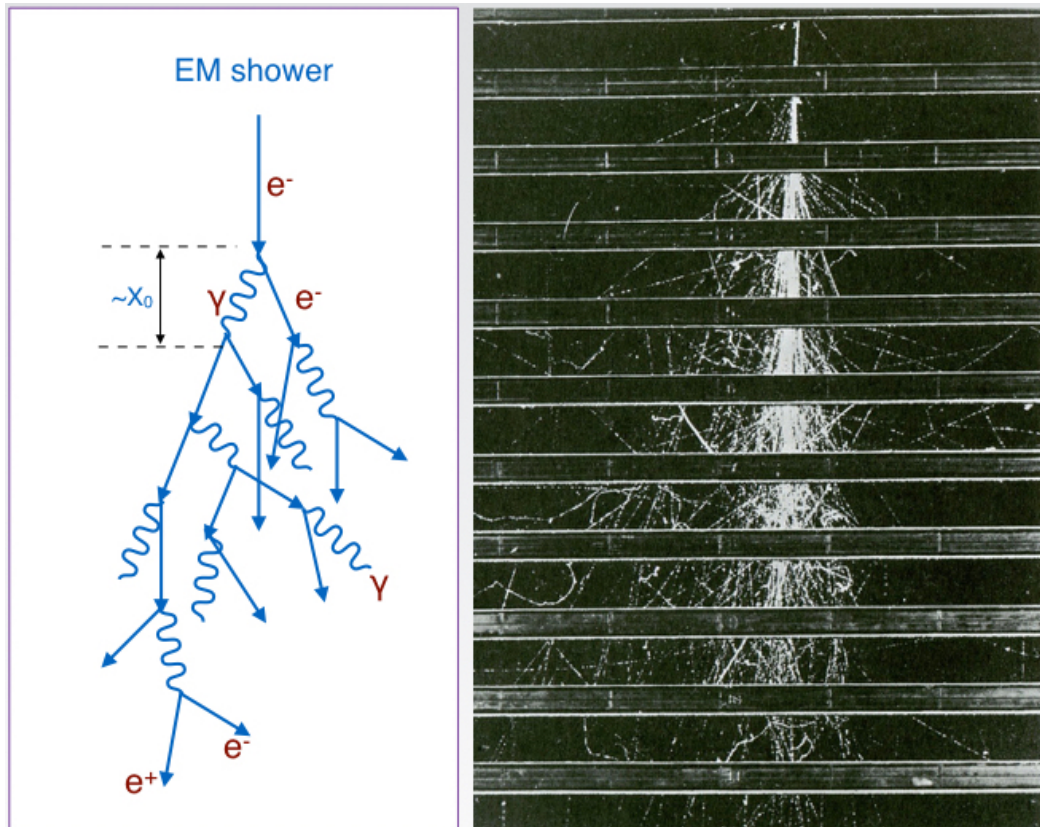
Fig. 4.11 Schematic representation of an electromagnetic shower induced by a high energy electron (left). On the right side we can see an electromagnetic cascade induced by a cosmic ray photon or an electron in a MIT-cloud chamber with brass plates.

of the initial particle is equally distributed between the final particles. With this hypothesis, X_0 is the natural unit of length for the electromagnetic interactions independently of the medium.

This process of pair production-Bremsstrahlung keeps on going until all the initial energy is lost. Then they are stopped and absorbed. After n stages, the mean energy of each particle is a factor $1/2^n$ of the initial energy E_0 and eventually an energy E_c is reached. At this critical energy this process is not dominant and the electrons tend to lose their energy through excitation-ionisation process. Finally, the cascade comes to an end.

The critical energy for iron is around 27 MeV, for lead 8 MeV and for uranium 6.5 MeV. Therefore, electromagnetic showers tend to be larger in heavy materials compared to the light materials and the electromagnetic showers vanish faster in X_0 units.

The use of electromagnetic radiation[1] for the detection of heavy materials is problematic. In one hand, the flux of electromagnetic radiation is smaller than the flux of muons arriving to the sea level. On the other hand, the small value of the radiation length of heavy elements makes that the showers produced inside the material quickly attenuate. The final energy of the electrons is so small that it is almost impossible that they can even reach our detectors.

However, it is possible that electromagnetic radiation could be used as a quick indicator of the presence of a heavy element and therefore it would be possible to determine if a container needs a more detailed analysis.

4.5   Structure of the simulations

In the previous sections we have introduced the different elements and tools that we need to perform our simulations. We have chosen the EnsarRoot framework and the physics concerns the analysis of angular distributions and other observables related to it.

The simulations are divided in two main blocks. One is devoted to the systematic study of the scattering angle in different materials and energies. As it was mentioned before, there are two main processes:*muon* and *electromagnetic* components, as shown in Fig.4.12. The other
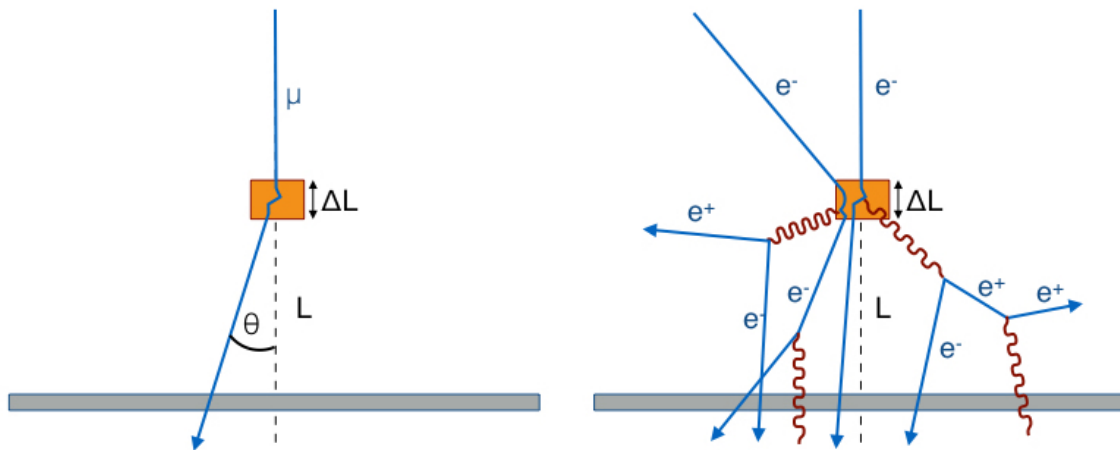
Fig. 4.12 Structure of the simulations. On the left side we can find the muon dispersion and on the right side the generation of an electromagnetic shower.

set of simulations is related to the data obtained with the real detector, *muTT*. We simulated the different setups with the detector configuration and proposed several observables that can be used to the material identification.

---

[1]From now we will denote as *electromagnetic radiation* to electrons and photons without distinction, since they are both responsible of the electromagnetic showers.

To be as much realistic as possible, in the first set of simulations we have used the typical energetic range of the cosmic rays and a RPC as the reference detector.

4.5.1 Scattering angle simulations

This part of the simulations concerned to the systematic study of the scattering angle of a cosmic ray due to the interaction with heavy materials. Since the only variable of interest is the angle, we fired vertical particles over the material, as shown in Fig.4.12. The trajectory of the particle is perfectly determined and we can control all variables of the particle.

In table 4.1 we make a list of the particles used as projectiles and their kinetic energy. The values of the table are chosen according to the information of cosmic ray flux at sea level.

Particle	Energy (GeV)										
μ	0.1	0.2	0.3	0.4	0.5	1.0	2.0	4.0	8.0	16.0	32.0
e,γ	0.1	0.2	0.3	0.4	0.5						

Table 4.1 List of particles and kinetic energies used in the simulations.

The electromagnetic component of cosmic radiation is not easy to study because the angular part is not a variable of interest. Electromagnetic showers are formed and therefore there is no point asking about the angular deflection of the incident electron or photon. In this case, we proposed the number of produced secondary particles during the interaction. Furthermore, if the secondary electron has energy enough to reach the detector we can determine the deflection angle compared to the incident particle.

Table 4.2 shows the list of materials and their thickness that were used in our simulations. Some of the intrinsic properties of the materials were also shown. As we mentioned in previous section, it is convenient to use the variable $\varepsilon = x \cdot \rho$ to express the thickness of the material, where ρ is the density.

The simulated experimental setup can be seen in Fig.4.13 where particles have a perpendicular trajectory to a dense material. We used the same RPC detector as the used in the TRAGALDABAS detector. As it was mentioned, this detector has two gaps filled with freon gas. These are our active volumes. Since we are interested in the general angular properties independently of the experiment, we first ignore the signal discretization implemented for TRAGALDABAS. That means that we only keep the geometry of the detector but we are not interested on the hits data as it would be seen in TRAGALDABAS, i.e, we only work in point data level.

Material	Z	$\rho(g \cdot cm^{-3})$	X (cm)	ε $(g \cdot cm^{-2})$	X/X_0
Fe	26	7.87	1	7.9	0.6
			5	39.4	2.8
			10	78.7	5.7
			20	157.5	11.4
Pb	82	11.35	1	11.4	1.8
			5	56.8	9.0
			10	113.5	17.9
			20	227.0	35.7
W	74	19.30	1	19.3	2.9
			5	96.5	14.3
			10	193.0	28.6
			20	386.0	57.1
U	92	18.95	1	19.0	3.1
			5	94.8	15.6
			10	189.5	31.3
			20	379.0	62.5

Table 4.2 List of materials used in the simulations and some of their intrinsic properties. The number ε represents the quantity (thickness·density). The last column represents the thickness in radiation length units.
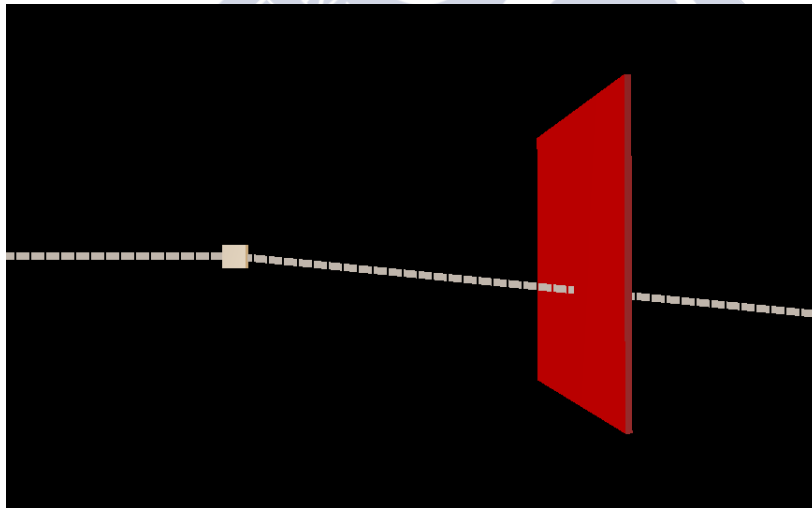

Fig. 4.13 Schematic view of the setup of the simulations. A particle is arriving perpendicular to the surface of the material.

4.6   Analysis of the scattering properties

In this subsection we present the results of the first set of simulations for muons and electrons.

1. In the case of muons we fired one million of particles for each thickness-material configuration.

2. In the electromagnetic case we only fired ten thousand electrons, due to the big amount of secondary particles that are produced in the electromagnetic shower. Computationally, it is no possible to manage them in memory.

We performed the simulations inside the EnsarRoot framework using the standard FairBox generator. As we are mainly interested on the deflection angle, we placed the source of emission at 30 cm from the upper surface of the material and fired particles with θ and ϕ equal to zero. This means that the incident particles are vertical and it is no worth placing the detector at a long distance.

4.6.1 Muon inelastic scattering: angle

As it was mentioned before, a charged particle inside a material medium will experience a random process in which it will change its trajectory. This process is the inelastic Coulomb multiple scattering. There is a non-zero probability that the muon would have a big angle due to a non-infinitesimal deviation inside the material. This *anomalous* angle are much more likely in heavy elements since they have more bounded electrons per volume unit.

An example of angular distribution can be found in Fig.4.14. In this figure we can see that the central part of the distribution follows a Gaussian shape, but there are tails corresponding to the anomalous scattering angles. The example in the figure corresponds to muons with 0.2 GeV of kinetic energy on 1 cm of lead. We used logarithmic scale on the *y-axis*. In that representation a Gaussian distribution takes parabolic form centred in the origin, and it becomes easier to determine the interval in which the distribution has a Gaussian shape. Moreover, the tails of the distribution are more visible and can be quantified.

For the analysis of the angle we used the following magnitudes:

1. First, we note that the angle can be positive or negative. We gave the angle a sign whether the particle deviated to positive or negative value of the *x-coordinate*. This is made to force the distribution to be centred on zero.

2. One first observable is the deviation obtained by a Gaussian fit over the central part of the distribution: σ_g.

3. Other observable is the standard deviation of the sample data:

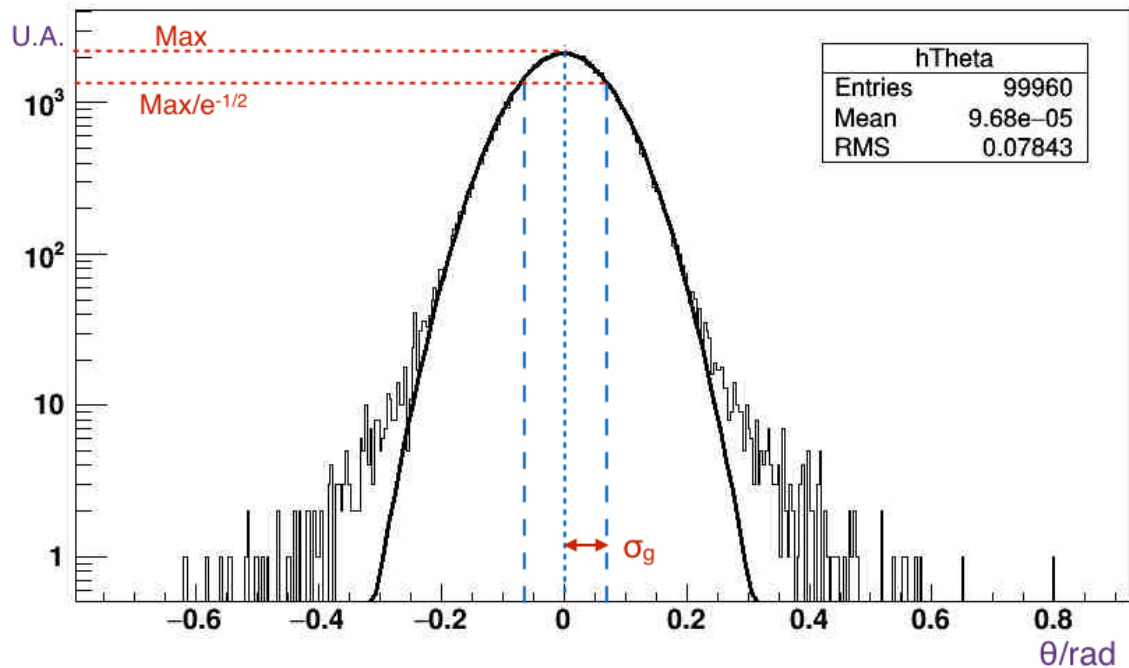$$\sigma_m = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\theta_i - \bar{\theta}\right)^2} \tag{4.3}$$

Fig. 4.14 Histogram of the scattering angle for muons with kinetic energy of 0.2 GeV on 1cm of lead.

where N is the size of the sample, θ is the value of the angle and $\bar{\theta}$ is the mean value of the angle distribution.

4. The difference between both deviations:

$$\Delta\sigma = \sigma_m - \sigma_g \qquad (4.4)$$

This quantity is very useful to determine the importance of the tails in the distribution.

Figure 4.15 shows the Gaussian angular dispersion, σ_g for muons over different materials of different thickness. It is observed, as expected, that this variable decreases as the energy of the particle increases. It is also observed that it increases as the thickness of the target increases. If we compare two materials with the same thickness and energy it can be seen that σ_g increases with the atomic number, Z. Figure 4.16 represents the same results for σ_g but using logarithmic scale on $y-axis$. The use of logarithm allows us to better distinguish the high energy region.

With the aim of extrapolate the values shown in previous figures to a more realistic situation, we studied the scattering angle when the target material is placed at distances L
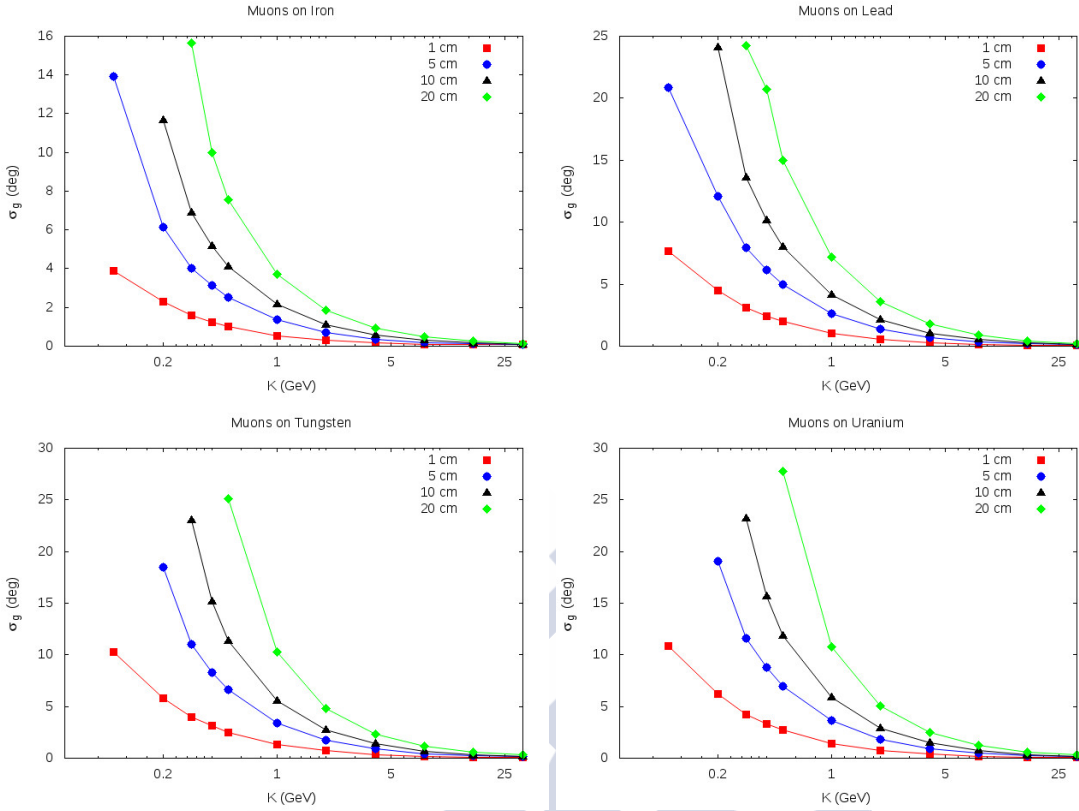
Fig. 4.15 Gaussian angular dispersion, σ_g for muons on Fe, Pb, W and U.

0.7 m, 1.5 m and 2.25 m above the RPC detector. These distances were selected taking into account the size of the container and its distance to the detection plane in a real life situation.

Our hypothesis is that we are able to extrapolate to any other distance L and provide approximated values of the angular dispersion. For that, we define the lateral linear dispersion, σ_x as

$$\sigma_x = L \cdot \sigma_m \tag{4.5}$$

and can be seen in Fig.4.17

The results are obtained for energies between 0.6 GeV and 8 GeV, that correspond with almost the 80% of muons and we present them in tables 4.3, 4.4, 4.5, and 4.6

Almost 50% of muons that arrive to Earth surface have energy less than 2 GeV. It is observed from the tables that in almost all heavy materials, muons with an energy less than 2 GeV present σ_x values of several centimetres. This could mean that even by the use of detectors with a poor angular resolution a reconstruction of a dense target is possible.

Iron: table of typical σ_x (cm)											
Energy	Target-detector distance										
of	0.75m				1.5m				2.25m			
muon	Thickness (cm)				Thickness (cm)				Thickness (cm)			
(GeV)	1	5	10	20	1	5	10	20	1	5	10	20
0.6	1.4	3.5	5.6	10.1	2.9	6.9	11.1	20.2	4.3	10.4	16.7	30.3
1	0.8	1.9	2.9	5.0	1.5	3.8	5.8	9.9	2.3	5.6	8.6	14.9
2	0.4	1.0	1.5	2.5	0.8	2.0	3.0	5.0	1.2	2.9	4.4	7.4
4	0.2	0.5	0.8	1.2	0.4	1.0	1.5	2.5	0.6	1.5	2.3	3.7
8	0.1	0.2	0.4	0.6	0.2	0.5	0.8	1.2	0.3	0.7	1.1	1.9

Table 4.3 Lateral uncertainties for muons of several energies on iron targets of different thickness. The material was placed at distances of 0.75 m, 1.5 m y 2.25 m from the detector.

Lead: table of typical, σ_x (cm)											
Energy	Target-detector distance										
of	0.75m				1.5m				2.25m			
muon	Thickness (cm)				Thickness (cm)				Thickness (cm)			
(GeV)	1	5	10	20	1	5	10	20	1	5	10	20
0.6	2.6	6.5	10.6	20.1	5.2	13.0	21.1	40.2	7.8	19.4	31.7	60.3
1	1.4	3.4	5.4	9.4	2.8	6.8	10.8	18.8	4.2	10.2	16.2	28.3
2	0.7	1.8	2.8	4.7	1.5	3.5	5.5	9.3	2.2	5.3	8.3	14.0
4	0.4	0.9	1.4	2.3	0.8	1.8	2.8	4.7	1.1	2.7	4.2	7.0
8	0.2	0.5	0.7	1.2	0.4	0.9	1.4	2.3	0.6	1.4	2.1	3.5

Table 4.4 Lateral uncertainties for muons of several energies on lead targets of different thickness. The material was placed at distances of 0.75 m, 1.5 m y 2.25 m from the detector.

Tungsten: table of typical σ_x (cm)											
Energy	Target-detector distance										
of	0.75m				1.5m				2.25m			
muon	Thickness /cm				Thickness (cm)				Thickness (cm)			
(GeV)	1	5	10	20	1	5	10	20	1	5	10	20
0.6	3.3	8.7	15.0	35.1	6.6	17.3	30.1	70.3	9.9	26.0	45.1	105.4
1	1.8	4.5	7.3	13.6	3.6	9.0	14.6	27.2	5.3	13.4	21.8	40.7
2	0.9	2.3	3.6	6.3	1.8	4.6	7.2	12.6	2.8	6.9	10.9	18.9
4	0.5	1.2	1.8	3.1	1.0	2.3	3.6	6.2	1.4	3.5	5.4	9.2
8	0.3	0.6	0.9	1.5	0.5	1.2	1.8	3.0	0.8	1.8	2.7	4.5

Table 4.5 Lateral uncertainties for muons of several energies on tungsten targets of different thickness. The material was placed at distances of 0.75 m, 1.5 m y 2.25 m from the detector.
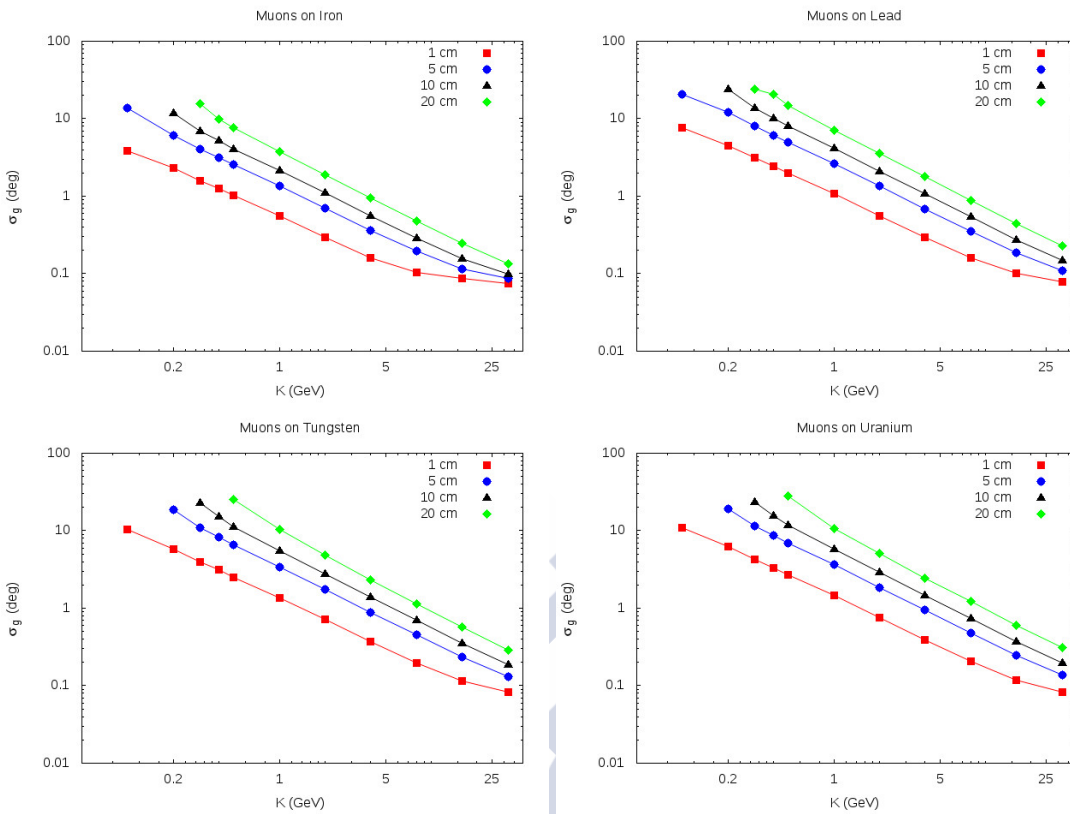
Fig. 4.16 Gaussian angular dispersion, σ_g for muons on Fe, Pb, W and U in logarithmic scale.
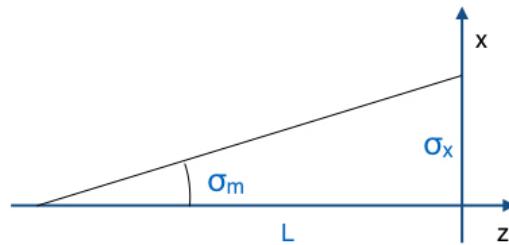

Fig. 4.17 Definition of linear lateral dispersion, σ_x.

4.6.2   Muon inelastic scattering: analysis of the properties of materials

Using the same simulations as in the previous sections it would be interesting to analyse the results taking into account the target material. In Fig.4.18 we represent the Gaussian deviations σ_g for all materials as a function of thickness in radiation length units. We performed the study for three different energies, 1 GeV, 4 GeV and 16 GeV. It is observed that, for the same muon energy, σ_g grows with x/X_0 independently on the material. This is

Uranium: table of typical, σ_x (cm)											
Energy	Target-detector distance										
of	0.75m				1.5m				2.25m			
muon	Thickness (cm)				Thickness (cm)				Thickness (cm)			
(GeV)	1	5	10	20	1	5	10	20	1	5	10	20
0.6	3.5	9.2	15.8	39.5	7.1	18.4	31.5	79.0	10.6	27.5	47.3	118.6
1	1.9	4.8	7.7	14.2	3.9	9.5	15.4	28.4	5.8	14.3	23.0	42.6
2	1.0	2.4	3.8	6.6	2.0	4.9	7.7	13.2	3.0	7.3	11.5	19.9
4	0.5	1.2	1.9	3.2	1.0	2.5	3.8	6.5	1.5	3.7	5.7	9.7
8	0.3	0.6	1.0	1.6	0.5	1.2	1.9	3.2	0.8	1.9	2.9	4.8

Table 4.6 Lateral uncertainties for muons of several energies on uranium targets of different thickness. The material was placed at distances of 0.75 m, 1.5 m and 2.25 m from the detector.
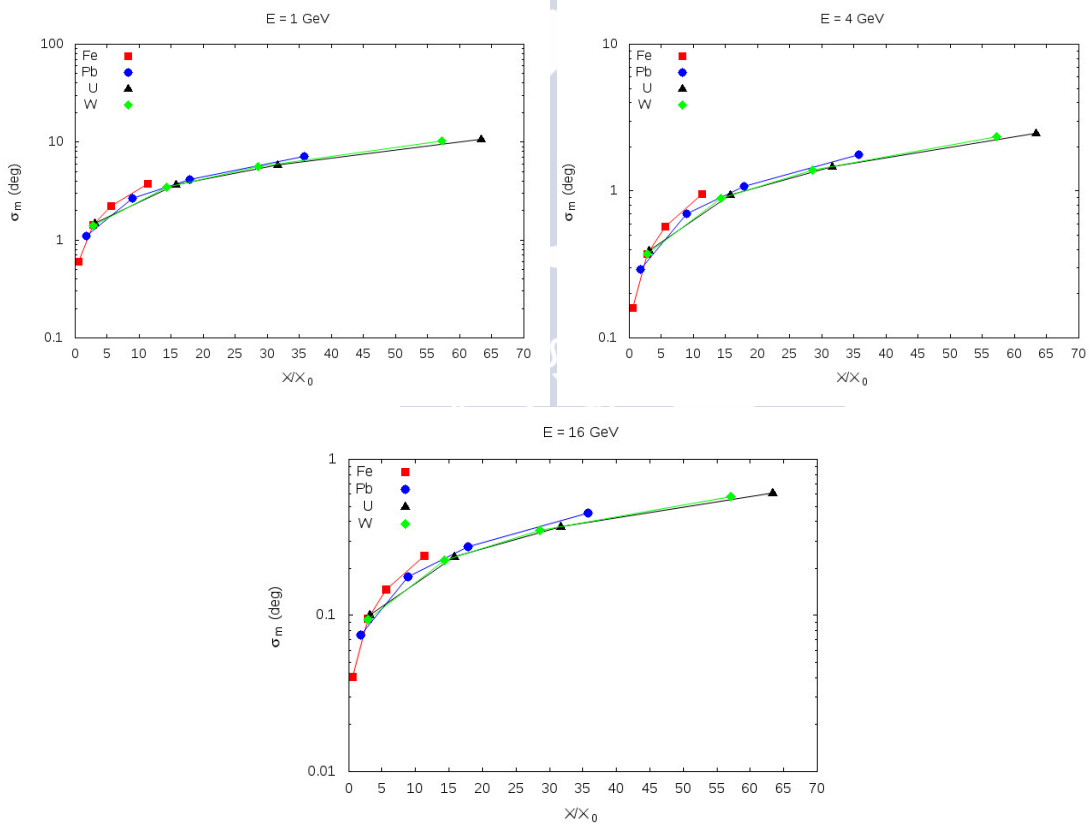

Fig. 4.18 Gaussian angular dispersion, σ_g as a function of x/X_0 for muons of 1, 4 and 16 GeV.

what we expected, since the scattering process is essentially of electromagnetic nature and

X_0 is a natural unit in those processes. A soft systematic effect it is observed depending on the material for high Z.

If we represent the same observable but as a function of ε we can see that the major difference comes from iron. The three heaviest elements present almost the same behaviour, but iron shows a significant difference compared to the others. This dependence of the results with the material, more notable for the three heaviest lead, tungsten and uranium, opens the possibility that an estimation of the material can be done as a function of radiation length or ε. For this, a good estimation of the momentum of the muon is needed.
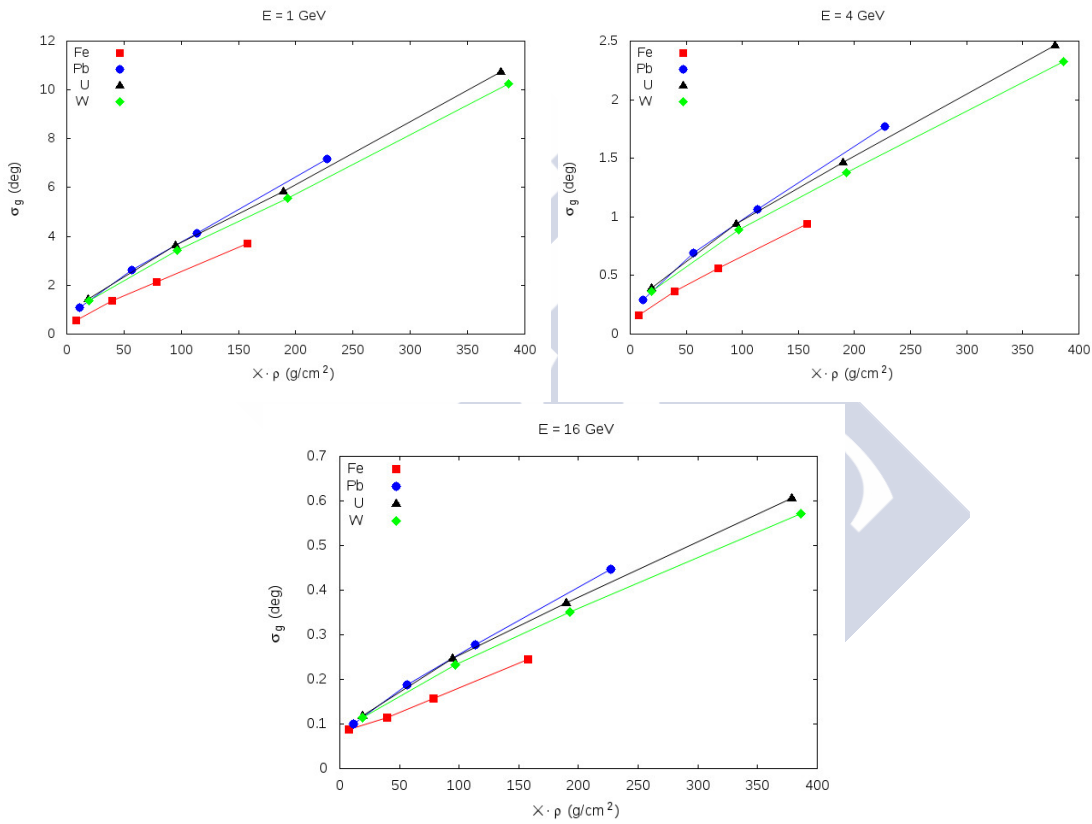

Fig. 4.19 Gaussian angular dispersion, σ_g as a function of ε for muons of 1, 4 and 16 GeV.

Finally, we made a study of $\Delta\sigma$ in order to quantify the importance of tails in the angular distribution, as seen in Fig.4.20

4.6.3   Muon inelastic scattering: analysis of kurtosis

One variable that can be representative from the practical point of view would be the kurtosis of the angular distribution. This variable could give information about the tails of the
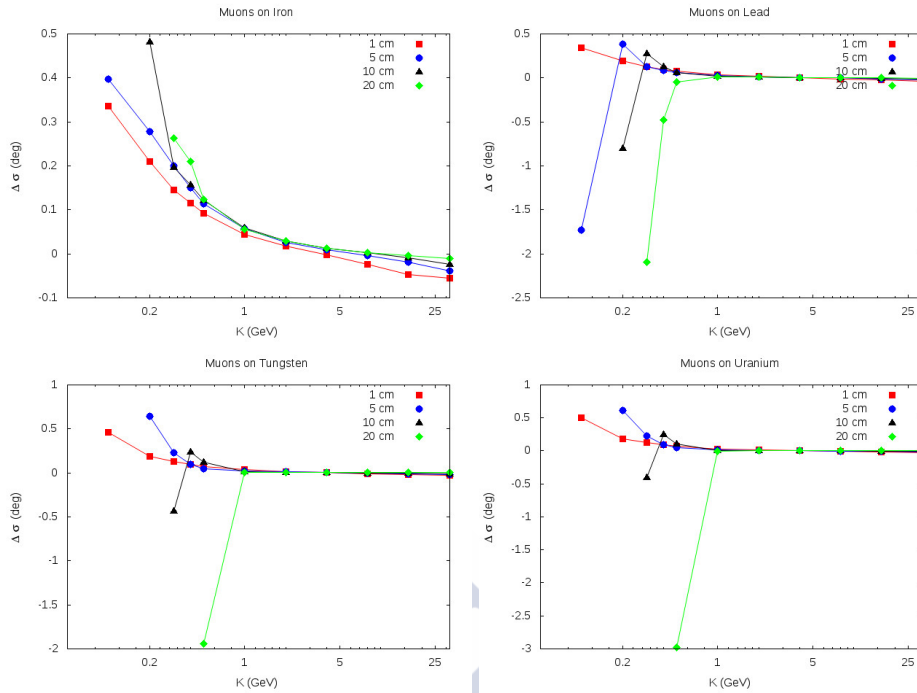
Fig. 4.20 Difference between σ_g and σ_m

distribution and how important they are compared with the rest of values. The goal is to study if there is a pattern or relation between the kurtosis and the materials. If this relation exists it could be possible to identify heavy materials by the use of kurtosis.

For a sample of data, the kurtosis is defined as

$$k = \frac{\mu_4}{\sigma^4} \tag{4.6}$$

where μ_4 is the momentum of the distribution around the mean value of fourth order:

$$\mu_4 = \frac{1}{N} \sum_{i=1}^{N} (x_i - \bar{x})^4 \tag{4.7}$$

It is common to refer the kurtosis to the Gaussian distribution:

$$k = \frac{\mu_4}{\sigma^4} - 3 \tag{4.8}$$

because the kurtosis of a normal distribution is 3. Therefore, with this definition, the kurtosis of the normal distribution is zero.

The kurtosis coefficient is a very useful observable to inform about the presence of data in a sample that are anomalously different. If there are just a few of anomalous data, or outliers,

the kurtosis tend to be extremely high. For example, if a distribution of data is bi-valued, it will have a large value of sigma. Therefore, the kurtosis coefficient is smaller compared to the kurtosis of a Gaussian distribution. We calculated the kurtosis coefficient for the angular
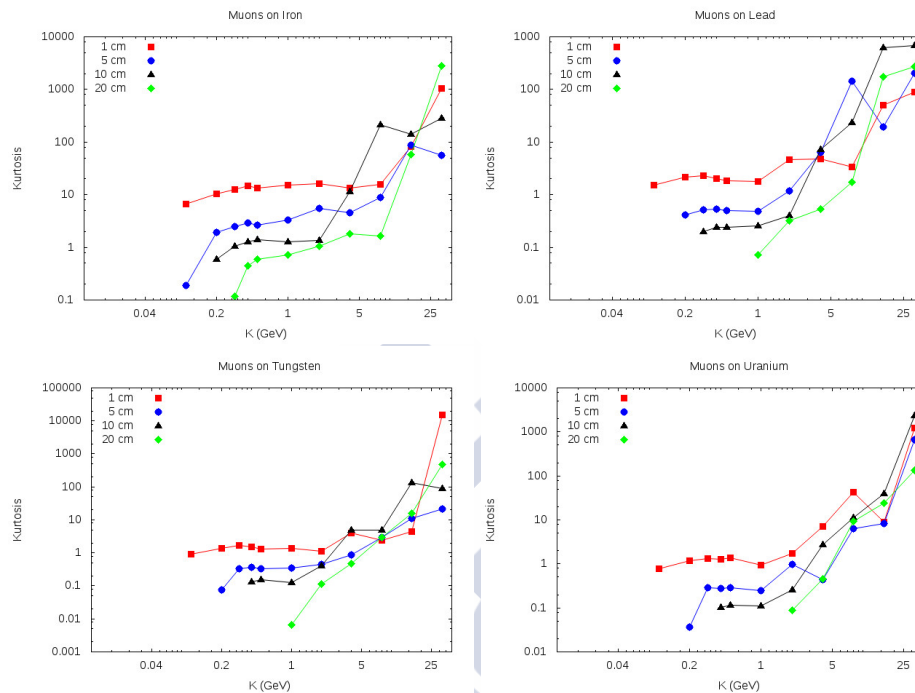

Fig. 4.21 Kurtosis coefficient of the angular distribution of muons on heavy materials.

distributions of muons over different materials. The results are presented in Fig.4.21. It observed, in general, that the kurtosis coefficient is bigger than the other heavy materials. Also, the kurtosis decreases with the width, showing that the behaviour is more Gaussian.

The most important result observed in the figure is that, beyond 2 GeV the kurtosis takes huge values for lead, uranium and tungsten. This effect is a consequence of the physics of the process. For high energetic muons the sigma of the distribution trends to be small. That is to say, typically those muons will not experience a significant scattering. This means that the mean value of the distribution is almost zero and the sigma is small. But, there is a probability that some of those muons will scatter, and the angle of scattering will be high. The mean value and sigma will not change significantly. In the kurtosis, μ_4 is always positive and is adding for every element of the distribution. The great value in the kurtosis is explained because we can find points in the distribution that are very distant to the mean value and the sigma is very small. Therefore, in the numerator we will find a high value divided by a small value in the denominator and then the kurtosis will be large.

As a conclusion, strong scattering in high energetic muons can be an indicator of high Z material, but it is maybe possible to find strong scattering with low energetic muons and

light materials. Maybe the muon energy can be used to weight the angular distribution, and therefore our detector must have a good time resolution.

4.7   Electromagnetic showers

The study of the electromagnetic showers can be used as a complementary analysis to the muon scattering. The rate of electromagnetic components in cosmic rays with energy enough to pass trough heavy materials is negligible. But maybe the estimation of secondary background and back scattered electrons could give some useful information, at least, for a fast identification of materials inside the containers.

Since electrons and gammas produce electromagnetic showers the computational memory is easily flooded. Therefore, we limited the number of particles to $N_0 = 10000$ over the same materials and thickness.

The observable that we considered is the ratio N/N_0, that represents the total number of secondary particles detected in the electromagnetic shower compared to the initial number of particles. Figure 4.22 shows some interesting conclusions. In heavy elements, in targets
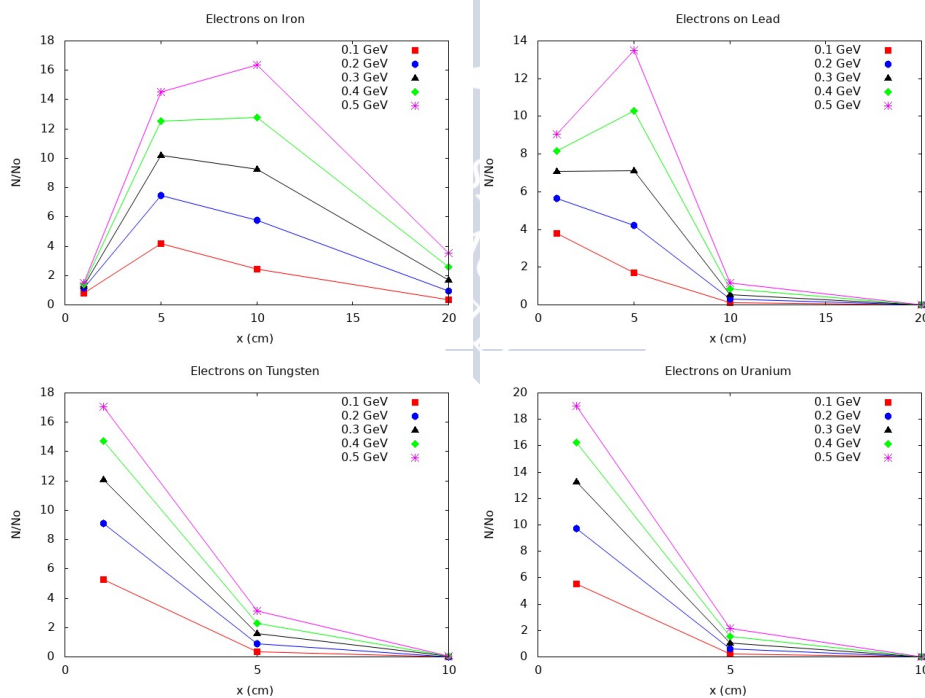

Fig. 4.22 Ratio of detected secondary particles compared to the initial number of particles N/N_0.

of several centimetres, absorption process is dominant since few particles are able to be

detected. The electrons-positrons do not have enough energy to travel and are absorbed inside the material or in air. However, thin materials are capable to produce high amount of secondary particles in heavy elements. Multiplicative factors of 20 can be observed. So, in the presence of heavy elements, it is possible that they are acting as radiators of secondary particles. Maybe if we are able to detect them they could be used as an indicator of the presence of heavy element inside the container.

4.8   Analytic approximation to the study of scattering properties

This section is devoted to the study of the scattering from an analytical point of view. The goal is to provide a method that can be used to improve the detector features. With this analysis it might be possible to make quick estimations of the configuration of the detector, the size of the sample to achieve some angular resolution, among others.

The starting point of the analysis consists in the idea that the angular uncertainty when we reconstruct a muon track ($\delta\theta$) provides uncertainty in the position of the point where the interaction muon-material took place (δZ)

In this approximation, we assume that $\delta\theta << \theta$ and the scattering angle θ is small. This is not always true for low energy muons hitting a heavy-Z material, for example.

With this assumption, we obtain that

$$\delta r = \delta\theta \cdot \sqrt{L^2 + X^2} \tag{4.9}$$

and

$$\Delta Z = \frac{\delta r}{\sin\theta} \simeq \frac{\delta\theta}{\theta}\sqrt{L^2 + X^2}. \tag{4.10}$$

Combining both equations:

$$\delta Z = Z \cdot \delta\theta \sqrt{1 + \frac{1}{\theta^2}} \tag{4.11}$$

This last equation shows how the uncertainty in the reconstructed vertex (δZ) decreases with the angle θ, assuming that the angular resolution is fixed. This indicates that the best imaging will be produced by the low energy muons, since they are the muons with high scattering angle.

Since most of the muons have in general low scattering angle, if we assume that the material is at a distance L above the detector it is possible to estimate, for a scattering angle θ the number of muons n that are needed to obtain a particular resolution δL. If we equal

both uncertainties:

$$\delta L = \frac{\delta Z}{\sqrt{n}} \tag{4.12}$$

and we assume that $Z = L$, we find that

$$n(\theta) = \left(\frac{\delta Z}{\delta L}\right)^2 = \delta\theta^2 \left(1 + \frac{1}{\theta^2}\right)\left(\frac{L}{\delta Z}\right)^2 \tag{4.13}$$

that it can be seen that is a decreasing monotone function in θ. That is to say, for a given detector with a known and fixed resolution, the more scattering, the less number of particles are needed to obtain a better quality in the image.

In case $\delta\theta$ is not small enough, we obtain:

$$n(\theta) = \left(\frac{\delta\theta}{\theta}\right)^2 \left(\frac{L}{\delta L}\right)^2 \tag{4.14}$$

that can be rewritten as

$$n(\theta) = \frac{\left(\frac{\delta\theta}{\theta}\right)^2}{\left(\frac{\delta L}{\delta L}\right)^2} \tag{4.15}$$

If we know the angular distribution of the muons, $f(\theta)$ and the number function $n(\theta)$ that provides the number of muons needed to obtain a certain resolution, it is possible to define a density function $d(\theta)$ that can be used to estimate the optimal scattering angle and the number of muons that are requested. We define the density function as:

$$d(\theta) = \frac{f(\theta)}{n(\theta)} \tag{4.16}$$

What does this function mean? Let's assume that we have a normalised scattering distribution for muons. We assume as well that the rate of incidence per time unit δt is N muons. The number of scattered muons in a particular angular interval centred in θ_i and with length $\Delta\theta$ would be

$$N_i = f(\theta_i) \cdot \Delta\theta \tag{4.17}$$

If $n_i = n(\theta_i) \cdot \Delta\theta$ muons are needed in that interval to obtain that particular resolution, the ratio $d_i = N_i/n_i$ represents the relation between the number of muons in the interval by the number of muons that we need. Consequently, the inverse represents the number of intervals δt that we need to obtain that resolution. We take as an example of calculation the situation shown in Fig.4.23. We know the normalised $f(\theta)$ and $n(\theta)$. With them it is constructed the function $d(\theta)$. In the figure, we can see that the maximum value of the function is obtained
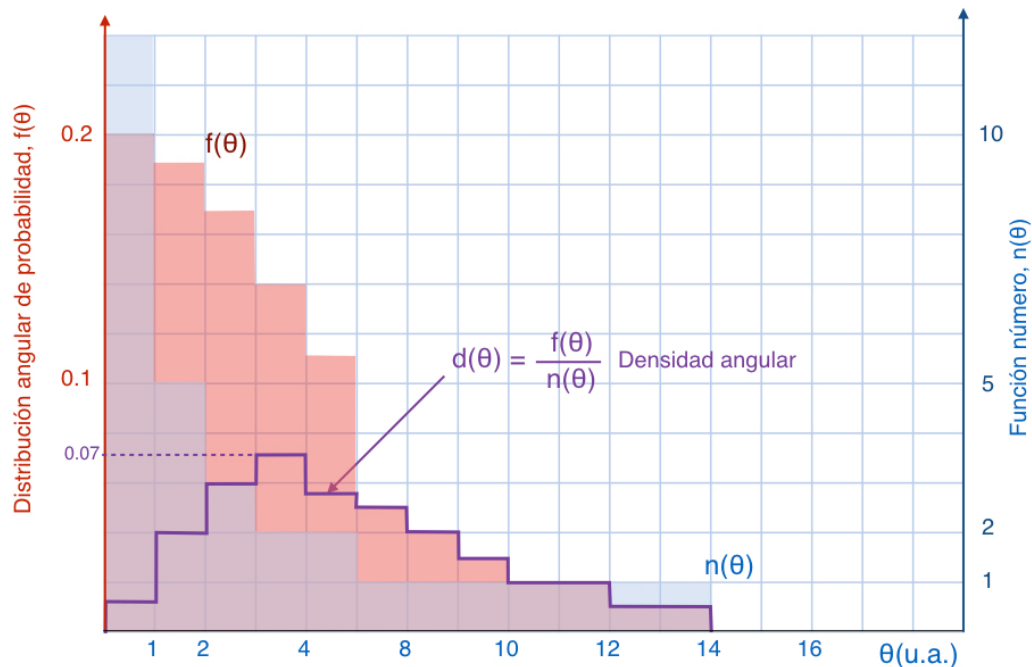
Fig. 4.23 Example of angular density caltulation to obtain a certain resolution.

inside the fourth interval $d_4 = 0.07$. This means that every muon in that interval contributes with a 7% of the total sample size that we need to see the target with the desired resolution. It would be needed, as a mean value, the inverse of that proportion: that is to say, 14 muons.

In a similar way, it is possible to estimate the number of events that we need to reconstruct the position of the vertex along the z-axis using the whole detector. The formalism is done in one dimension, but the extension to the other two is straightforward. As before, we consider small scattering angles.

According to the previous equations, if we consider a detector that provides an angular resolution $\delta\theta$, the resolution in the position of the vertex can be written from an interval θ_i with n_i in it:

$$\delta Z_i = \frac{Z}{\sqrt{n_i}} \cdot \frac{\delta\theta}{\theta_i} \tag{4.18}$$

Taking into account the resolution in the position of the vertex provided by all scattered muons at all possible angles, it is possible to compose them as a simple weighted superposition to

obtain the total δZ:

$$\frac{1}{\delta Z^2} = \frac{1}{\delta Z_1^2} + \frac{1}{\delta Z_2^2} + \frac{1}{\delta Z_3^2} \cdots \tag{4.19}$$

or

$$\frac{1}{\delta Z^2} = \left(\frac{n_1 \cdot \theta_1}{Z \cdot \delta \theta}\right)^2 + \left(\frac{n_2 \cdot \theta_2}{Z \cdot \delta \theta}\right)^2 + \left(\frac{n_3 \cdot \theta_3}{Z \cdot \delta \theta}\right)^2 + \cdots \tag{4.20}$$

and then it is easy to obtain

$$\delta Z = Z \cdot \delta \theta \left(\sum_{i=1}^{n_\theta} \frac{1}{n_i^2 \cdot \theta_i^2}\right)^{\frac{1}{2}} \tag{4.21}$$

where n_θ is the number of angular intervals.

If we call f_i to the probability that a scattered muon is inside the interval θ_i, the expectation value for the muons inside that interval will be $n_i = N \cdot f_i$, being N the total number of fired muons. With this, the previous equation will take the form:

$$\delta Z = \frac{Z \cdot \delta \theta}{N} \left(\sum_{i=1}^{n_\theta} \frac{1}{f_i^2 \cdot \theta_i^2}\right)^{\frac{1}{2}} \tag{4.22}$$

Finally, he number N of muons that we need to achieve a resolution δZ with a detector providing an angular resolution $\delta \theta$ at a distance Z is:

$$N = Z \left(\sum_{i=1}^{n_\theta} \frac{1}{f_i^2 \cdot \theta_i^2}\right)^{\frac{1}{2}} \frac{\delta \theta}{\delta Z} \tag{4.23}$$

In this estimation we assumed ideal muons without uncertainties in its trajectory. As a consequence, this is only a semi-ideal approximation that needs real data to be modified and developed in a more realistic way.

Chapter 5

Simulation of physical observables in *muTT*

5.1   Introduction

In Chapter 3 we presented a general description ot eh RPC detectors. As it was mentioned, there is a new model of RPC that will be used as standard of the muon tomography system. This detector is called *muTT*.

In this chapter we study the simulated behaviour of several physical observables in the *muTT* set-up. The simulations have two main goals. First, to study the image reconstruction by the use of different algorithms. Second, to study the capability of the system to distinguish materials of different density, in particular, those heavy elements which are candidates to be radioactive.

The formation of images is possible by the use of reconstruction algorithms such as TimTrack. As it was explained in Chapter 4, the Coulomb multiple scattering is the process that deflects the cosmic ray muons when they pass-through a material. The simulations shown a clear correlation between the scattering angle and the density of the material. By the study of the trajectories of the particles it is possible to provide the coordinates of the scattering centre.

The distinction of materials according to their density is a more complicated task. It is true that the deflection angle depends on the density (given in X/X_0 units) but the reconstruction algorithms by themselves are not able to distinguish between materials. Different materials, with different densities and thickness can present a very similar scattering behaviour.

In this chapter we present a reliable simulation of the physical structure and target configuration of run number 27. This particular run is the reference of the physical analysis performed in Chapter 7.

We propose different physical observables that could provide information about the density of materials, or at least make possible a distinction between them. These quantities are related to the spatial and angular magnitudes, and they will be presented in this chapter.

As it will be seen, we use two different reconstruction methods (POCA and TimTrack) and many observables can be defined by the difference between them. The TimTrack method is fully explained in Chapter 2 and some technical details were avoided and the final results were presented instead.

5.2   Configuration of the simulations

In this section we will describe the parameters and configurations used in the simulations. We present the geometry of the structure, the layout of the targets, energy, and angle of the incident particles.

Physical structure

For all of our simulations we used the geometrical structure shown in Fig.5.1. This arrangement of the planes is a representation of the real structure used to calibrate the *muTT* planes (see Chapter 3). With this, we can compare real data with our simulations.The standard unit of distance in EnsarRoot is the cm. Therefore, all distances shown in figures are cm, if not specified otherwise.

The origin of coordinates was chosen to be in the centre of the structure, between the upper and lower RPC planes. The *z*-axis was taken as positive along the propagation direction of the cosmic radiation.

The detector is divided in two regions: up and down. The name of a plane contains information about its position in the detector by the use of a letter and a number. We use the letters *U,D* referring to *up* and *down*.

The order of the plane is given by a number. We then have U1 (at $z = -80\ cm$), U2, D1 and D2 (at $z = 80\ cm$) planes. The origin is therefore between U2 and D1 planes. This system is very useful if at some moment we decide to put additional planes to perform future experiments.

In the real structure, at $z = 0$ there is an aluminium plane where the test materials are placed. The simulations take this into account, despite is not an important parameter.

Fig. 5.1 Overview of the physical structure used in the simulations. The target materials will be placed at $z = 0$. We took z as positive along the propagation of the cosmic rays. Distances are given in cm.

Target configurations

In Chapter 7 we took as a reference of analysis the set of data named as *run 27*. In that, three bricks of different materials; iron, lead and tungsten, were placed between upper and lower RPC planes.

To compare our simulated observables with the real data, we made a simulation of the same arrangement as in run 27. An overview of the run 27 is shown in Fig.5.2.

As we can see, the materials present in run 27 are not of the same size. The tungsten brick is smaller than the iron and lead bricks. In order to study the general behaviour of the observables depending on the material, we created an alternative layout. This new configuration consists on three bricks of the same size with the same materials as in run 27. Comparing the simulations with both configurations we can identify and exclude possible effects produced by the size of materials.

In Table 5.1 we define the configuration of both simulations scenarios. In the first scenario, named *Simul1*, we reproduced the same target configuration as in run 27. We have two bricks of dimensions $10 \times 20 \times 10 \ cm^3$ (2 l) made of iron and lead. The third brick is made of tungsten and with dimensions $10 \times 10 \times 10 \ cm^3$ (1 l). In Table 5.1 are shown the positions of the centres of the bricks together with their lengths along the three coordinate axis.

The second scenario, *Simul2*, consists on four bricks with the same transversal surface ($10 \times 10 \ cm^2$) but with different thickness. This Simul2 configuration is useful to perform systematic studies of the properties of observable quantities. Therefore, in each Simul2 configuration we have the same material (iron, lead or tungsten) but in four different thickness.

We will mention explicitly whether Simul1 or Simul2 was used in the set of simulations.
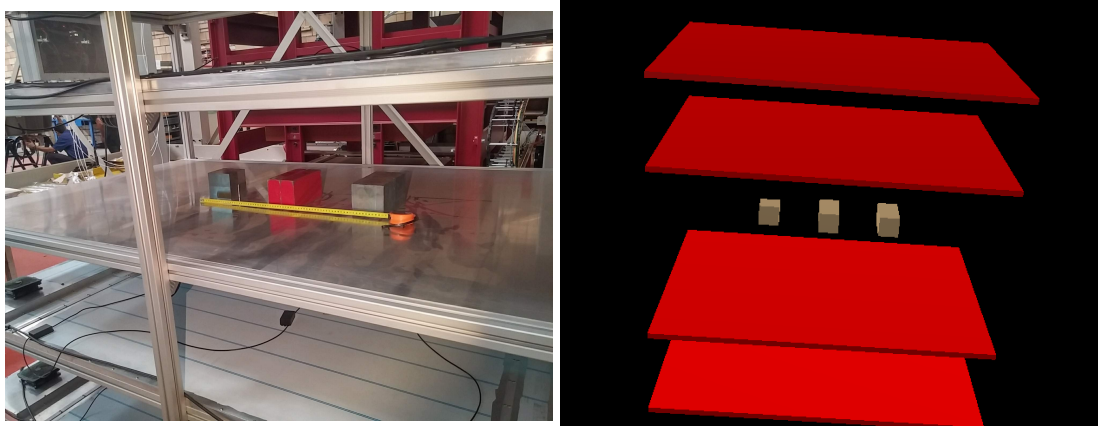
Fig. 5.2 Left: a picture of the real arrangement of the run 27. Right: EnsarRoot simulation of the run 27. It consisted on 1l of tungsten, 2l of lead and 2l of iron. The targets are centred along the *x*-axis.

Material	Centre	l_x	l_y	l_z
Fe	(-30,0)	10	20	10
Pb	(0,0)	10	20	10
W	(30,0)	10	10	10

Simul1

Material	Centre	l_x	l_y	l_z
	(-35,0)	10	10	1
Fe, Pb, W	(-25,0)	10	10	5
	(15,0)	10	10	10
	(35,0)	10	10	20

Simul2

Table 5.1 Distribution of materials in both simulation scenarios. The table contains the coordinates (x, y) of the centre of the brick and the lengths along the three axis (in cm). On the left side we have *simul1*, a representation of the run 27. On the right side, *simul2*, a configuration of four bricks with the same transversal surface, but different thickness.

Energy ranges

The next ingredient of the simulations are the primary particles. In the chapter devoted to the cosmic rays we presented in Fig.1.1 the flux of particles at sea level. In this table, we see that muons are the most important contribution to the cosmic ray flux at sea level. Therefore, our simulations were oriented to the study of the scattering properties of muons.

In our simulations, we divided the energy ranges of the simulated particles (muons) in three groups. In Table 5.2 can be seen, together with the values in energy, the number of muons that we expect at each energy range.

Despite the low energy muons are the less abundant, they provide the more amount of scattering information. The high energy muons do not provide much information about the scattering properties. These muons typically pass through the materials without deflecting.

Range	E (GeV)	Expected muons
Low	0.1–1.0	31
Intermediate	1.0–3.0	30
High	3.0–6.0	25

Table 5.2 Definition of the energy ranges used in the simulations. The table shows the relative intensity of muons that we expect for each energy range. These values are obtained from Fig.1.1.
.

Despite their scattering information is not very useful, they provide a background for the scattering observables.

5.2.1 Reconstruction algorithms

In this subsection we present the two algorithms that we used to form the images of the materials. We define two algorithms: the first method is the TimTrack and the other is the knows as Point Of Closest Approach (POCA). A full theoretical description of the TimTrack method can be found in Chapter 2. In this chapter we only present the final results of the algorithm.

TimTrack

The TimTrack method can be applied to find the interaction vertex of a muon with a material by the Coulomb multiple scattering.

The TimTrack algorithm is based on the least squares method and is able to provide the physical parameters related to the trajectory of the particle. It is assumed that the experimental data follow a particular model that includes the physical parameters, such as slopes, initial point or velocity, among others. The method is general and can be used in a lot of different situations with data following several models.

To obtain the position of the vertex with TimTrack we used the *supersaeta* model. For detailed information of the equations, see Chapter 2. The supersaeta vector is obtained satisfying these boundary conditions and considerations:

1. We take the detected points in the planes as input.

2. The model assumes that the particle follows a straight trajectory before and after its interaction with the material.

3. We assume that the incoming and outgoing trajectories have a point in common. This point is the deflection vertex.

After the method is applied, we obtain not only the vertex (X, Y, Z) but also the kinematic information that determines the whole path of the particle. The supersaeta vector contains the following parameters:

$$\mathbf{s} = \left(X_0, X_1', X_2', Y_0, Y_1', Y_2', T_0, S, Z_0\right) \tag{5.1}$$

where in this equation (X_0, Y_0, Z_0) is the position of the vertex, (X_1', Y_1') and (X_1', Y_1') are slopes of the trajectories before and after the material, T_0 is the initial time referred to an external clock and S is the inverse of the velocity (slowness).
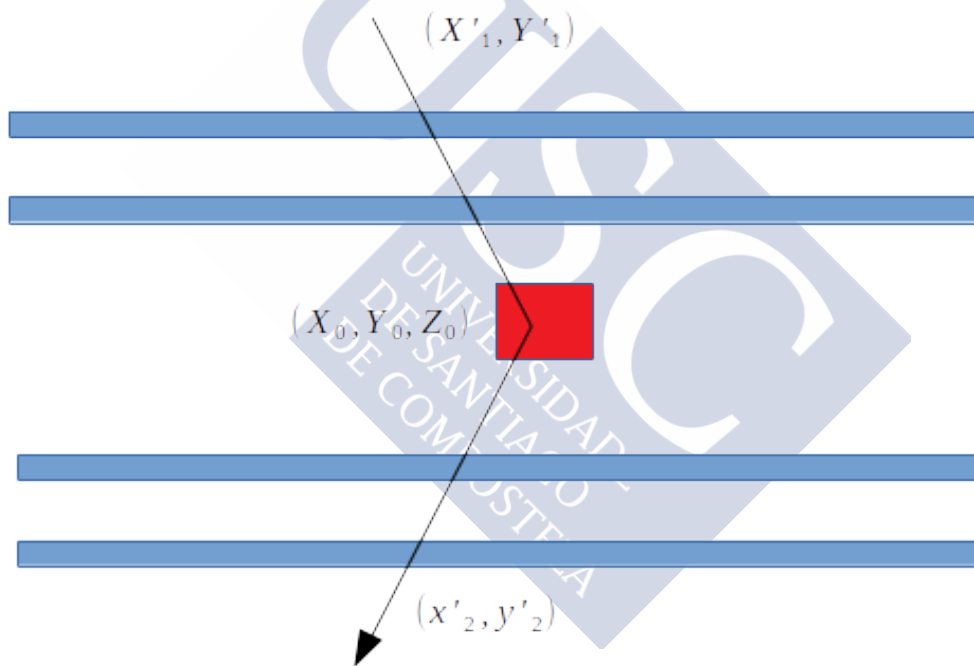

Fig. 5.3 Representation of a supersaeta. We solve the TimTrack equations to find a vector with the kinematic parameters. The incoming and outgoing trajectories must have a common point: the deflection vertex.

In the figure 5.3 can be seen an schematic representation of a supersaeta. This new object contains the information of the incoming and outgoing trajectories. As a boundary condition, it must to be fulfilled that both trajectories have a common point. This point is the vertex. With the TimTrack method we can obtain the values of all parameters and their mutual correlations at one step, using as input the space built with the experimental data.

Along this chapter we will use the words TimTrack and supersaeta indistinctly. All the vertices obtained with TimTrack are the solution of the supersaeta model.

POCA

One of the most used and simple methods in vertex reconstruction due to multiple scattering is the Point Of Closest Approach. This method is purely geometrical and does not provide information about the dynamics of the particle or the correlation between these parameters. POCA reconstructs under this conditions:

1. The measured points are taken as the input of the method.

2. With the points in the upper planes we build a straight line, while the same is done with the lower planes.

3. For the two straight lines, we find the points that fulfil that the distance between them is minimal.

4. The vertex is given by the mean value of the previously calculated points.

The equations of the lines up and down are, as a function of the parameters s and t:

$$P(s) = P_0 + s \cdot \mathbf{u} \qquad Q(t) = Q_0 + t \cdot \mathbf{v} \tag{5.2}$$

where $\mathbf{u}$ and $\mathbf{v}$ are their direction vectors. The relative vector between the lines is, therefore $\mathbf{w}_0 = P_0 - Q_0$. The values of the parameters that give the closest distance between the two lines are:

$$s_c = \frac{be - cd}{ac - b^2} \qquad t_c = \frac{ae - bd}{ac - b^2} \tag{5.3}$$

where

$$
\begin{aligned}
a &= \mathbf{u} \cdot \mathbf{u} \\
b &= \mathbf{u} \cdot \mathbf{v} \\
c &= \mathbf{v} \cdot \mathbf{v} \\
d &= \mathbf{u} \cdot \mathbf{w}_0 \\
e &= \mathbf{v} \cdot \mathbf{w}_0
\end{aligned}
$$

If we substitute the values of the parameters, we obtain the points (one at each line) that satisfy that their distance is minimal, $P(s_c)$ and $Q(t_c)$. The vertex is given by the middle

point between both values:

$$(X,Y,Z) = \frac{P(s_c) + Q(t_c)}{2} \tag{5.4}$$

Both reconstruction methods are programmed as an independent script. This file takes the root file produced by the simulation and extracts the information concerning the measurements at each plane. Taking these points as input both algorithms are applied.

To check both reconstruction methods, we applied them to the Simul1 (run 27) configuration. To perform the simulation, we fired muons with different angles and energies between 0.1 and 3 GeV. The results of the simulated vertices reconstructed with both methods are shown in Fig.5.4.
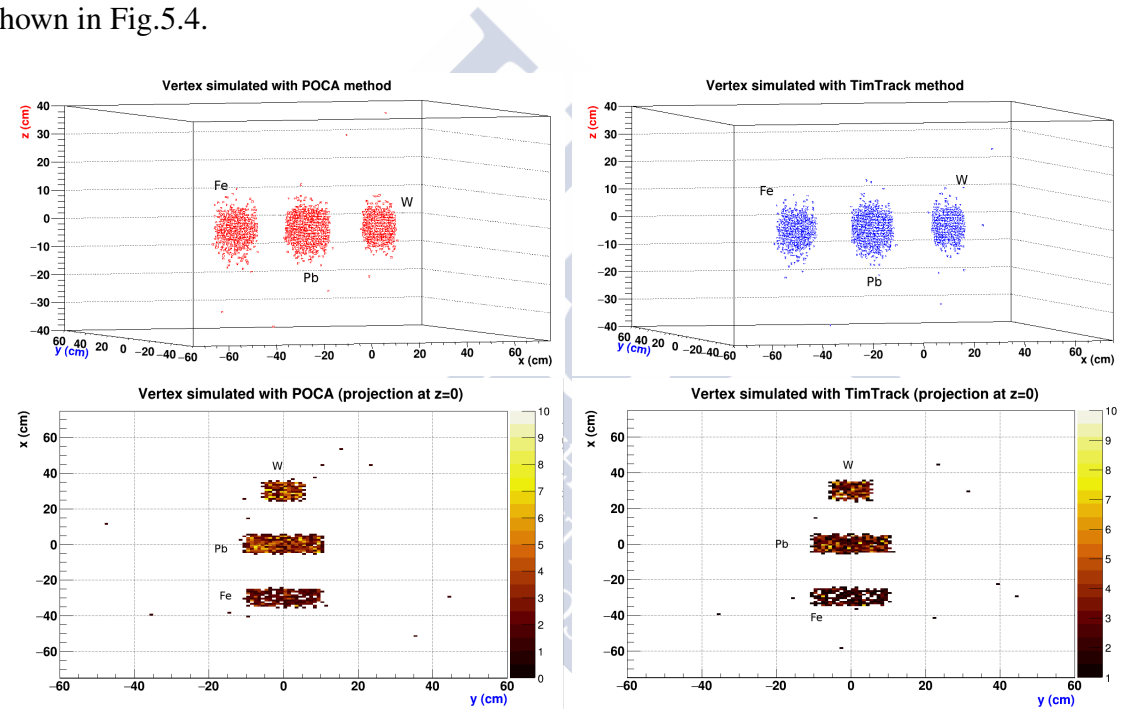

Fig. 5.4 Test of the reconstruction methods. We reconstructed the deflection vertices with both methods for the Simul1 target distribution (run 27). Left: reconstruction using POCA. Right: reconstruction with TimTrack. Their respective 2D projections at $z = 0$ are shown at the bottom of the figure.

As it can be seen, both methods provide a good reconstruction of the shapes of the materials. It is also clear that they are very close in the reconstruction, there are not significant differences at a first glance.

On the left side of the figure are shown the vertices reconstructed with POCA, together with the projection at the $z = 0$ plane. The right side contains the same information for the TimTrack.

With the projections at the origin it is possible to estimate the dimensions of the bricks. Lead and tungsten seem to present a higher density of vertices at $z = 0$ compared to iron.

If we compare the projections at the origin with the real shapes, we can see that both methods are capable to reproduce the actual dimensions of the bricks.

5.3   Angular resolution of the detector

An estimation of the angular resolution of the *muTT* detector can be done. This variable can be interpreted in two different ways. First, the angular resolution could provide information about the capability of our detector to distinguish between two different materials. An angular resolution of the order of mrad is required to this, as it is mentioned in the work of Baesso et al. [19].

Second, the estimation of the angular resolution gives us an idea about the efficiency of our reconstruction method. The information of the generated particle is completely known and therefore it is possible to compare the reconstructed track to the generated by Monte Carlo.

An analytical expression for the angular resolution can be provided by the formalism done in 4.8. In that section, it was formulated a relation between the angular resolution and the resolution (uncertainty) of the measured vertex. This section can be used to compare and make a theoretical approximation of the angular resolution.

To estimate the angular resolution of the *muTT* we performed a simulation without target materials. We generated a million muons with an energy range between 1.0 GeV and 5.0 GeV. The muons have angles below 15 degrees of inclination and are free in the azimuthal component. The particles are generated randomly from a virtual plane placed at 50 cm above U1. With these parameters we obtain enough events that pass through the four detectors.

The reconstruction of the tracks is made with the TimTrack method using a non linear model, as shown in Chapter 2. We consider only the events that cross the four planes, as in the real *muTT* case.

The TimTrack method provides the propagation parameters of the particles. In particular, we can define the polar angle from the projected slopes X' and Y'. This reconstruction angle can be compared to the angle generated in the Monte Carlo simulation. In Fig.5.5 is shown the distribution of the residual angle. The angular resolution can be estimated as the difference between the reconstructed angle with TimTrack and the angle provided by the Monte Carlo simulation.

A resolution of 1.8 mrad was achieved. This number is obtained by a two-step fitting. First, we fit the original distribution to a Gaussian function and extract the sigma of this
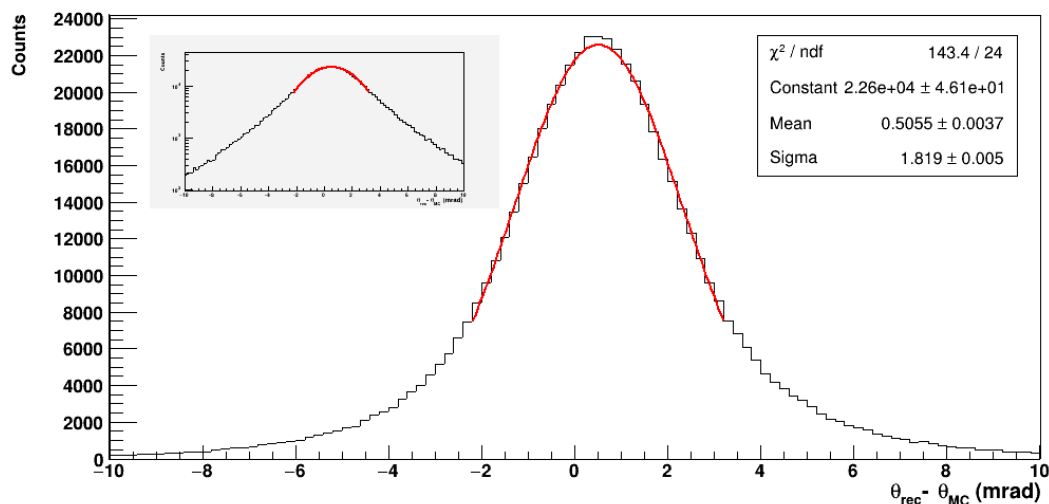
Fig. 5.5 Angular resolution of the *muTT* detector obtained from the distribution of the residual angle. The final parameter is obtained by a double fitting to 1.5σ. In the pad above is represented the same distribution but in logarithmic scale.

curve. A second Gaussian fit is made only at 1.5σ range. In the figure is also represented the distribution in logarithmic scale.

This value is consistent to the estimation of angular resolutions of the order of mrad. After this result we can affirm that the reconstruction method is providing a value of the resolution good enough to reconstruct volumes of high density materials accurately and even detect the presence of high density materials.

5.4   Physical observables

The combination of the two independent reconstruction methods, TimTrack and POCA, makes possible to provide more clear images of the sample bricks. However, as important as the shape it is to determine whether if the brick is made of a heavy material or not, and if possible, how heavy it is.

Some groups propose the study of the final distribution of the particle tracks. They consider variables involving angular scattering and momentum of the particles [47]. Another approach is made by the use of maximum likelihood methods combined with clustering algorithms [43].

In this thesis we propose some physical observables that might provide information about the distinction between heavy materials. They are defined as the differences of the variables obtained using both reconstruction methods, POCA and TimTrack.

A quick summary of the names and definitions of the observables is found in Table 5.3.

Name	Symbol	Definition
Vertex differences	dr	$(x_{\mathrm{poca}} - x_{\mathrm{ss}}, y_{\mathrm{poca}} - y_{\mathrm{ss}}, z_{\mathrm{poca}} - z_{\mathrm{ss}})$
Polar deviation	$d\theta$	$\theta_{\mathrm{poca}} - \theta_{\mathrm{ss}}$
Relative polar deviation	$d\theta/dz$	$(\theta_{\mathrm{poca}} - \theta_{\mathrm{ss}})/(z_{\mathrm{poca}} - z_{\mathrm{ss}})$
Spherical deviation	$d\Omega$	$\arccos(\vec{v} \cdot \vec{w})$
Time delay	dt	$t_4' - t_4$

Table 5.3 Definition of the physical observables. A detailed description of the variables can be found in the text.

To avoid effects related to the size of the materials we performed the simulations using the Simul2 configuration. Muons were generated as an uniform rain over the detector with a fixed angle. All the parameters related to the simulations of the observables are found in Table 5.4

Parameter	Value
Number of muons	10^6
Distribution	Simul2
Energy	Low, Intermediate, High
Angle	0, 5, 10, 15 (degrees)

Table 5.4 Parameters of the simulations of the physical observables.

5.4.1   Difference in vertex coordinates

One of the first observables that can be presented is the difference in the position of the scattering vertex obtained with both methods. These differences can contribute to estimate the sharpness of the reconstructed image.

The vertical component (z) is the most significant variable in the reconstruction process. Since most of the muons are close to a vertical incidence, the z component is more sensible to uncertainties in the reconstruction process.

As an example of this fact, in Fig.5.6 is shown the uncertainty in position associated to the reconstruction process. The more the particle scatters, the better reconstructed vertex. This fact looks more evident in the POCA method. If both up and down trajectories are parallel, the condition of closest approach between them can be satisfied anywhere along
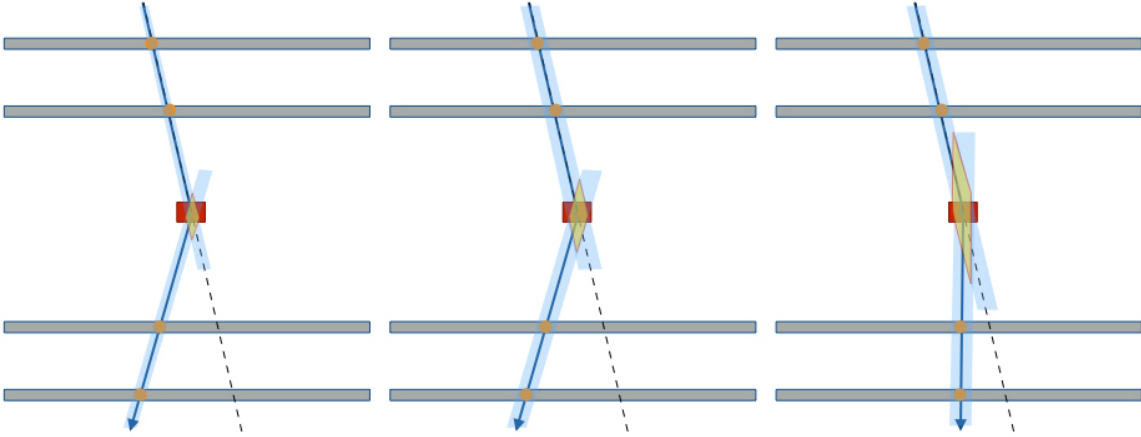
Fig. 5.6 Uncertainties in the reconstruction process. Depending on the relative angle between the incident and scattered directions, the uncertainty in the vertical component is changing.

the trajectory. As a result, the uncertainty in the longitudinal component is more relevant compared to the transverse dimensions. This is the motivation to consider z and dz (defined later) as the standard reference variables of the reconstruction process.

Both reconstruction methods, POCA and TimTrack, provide information about the scattering vertex. If we consider the difference between the z-coordinate of the vertex,

$$dz = z_{\text{poca}} - z_{ss} \tag{5.5}$$

it might be possible to stablish a relationship between this variable and the uncertainty in the width of the material. As we mentioned, the uncertainty in z component is related to the scattering angle, and at the same time this angle is related to the mass width of the material.

We performed a systematic study of the behaviour of dz for different materials for several energies, thickness and incident angles. As an example, Figure 5.7 shows the difference dz for vertical muons on 20 cm of tungsten at low energies.

The differences in all components can be grouped in a vector that we named dr:

$$dr = (dx, dy, dz) = (x_{\text{poca}} - x_{\text{ss}}, y_{\text{poca}} - y_{\text{ss}}, z_{\text{poca}} - z_{\text{ss}}) \tag{5.6}$$

where $x_{\text{poca}}, y_{\text{poca}}, z_{\text{poca}}$ are the vertex coordinates obtained with the POCA method and x_{ss}, y_{ss}, z_{ss} are obtained with the TimTrack.

This vector has the information about the lateral and vertical dispersion and can be used as a boundary condition for the reconstruction of materials.
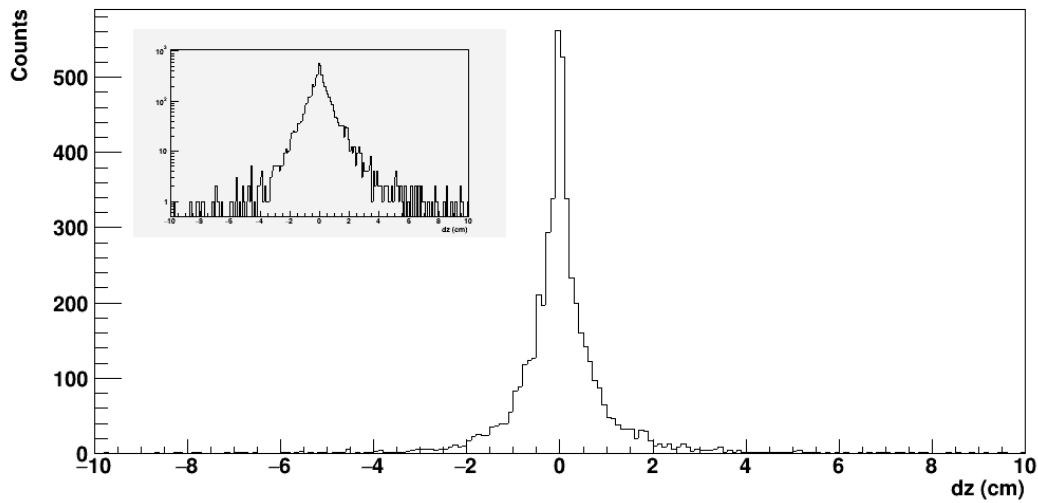
Fig. 5.7 Difference dz in tungsten. The simulated particles are vertical muons of low energy on 20 cm of W.

The systematic study of the observable was done following the parameters defined in Table 5.4. As an example, Figure 5.8 shows the results of simulated low energy muons on tungsten with an incident angle of 15 degrees.
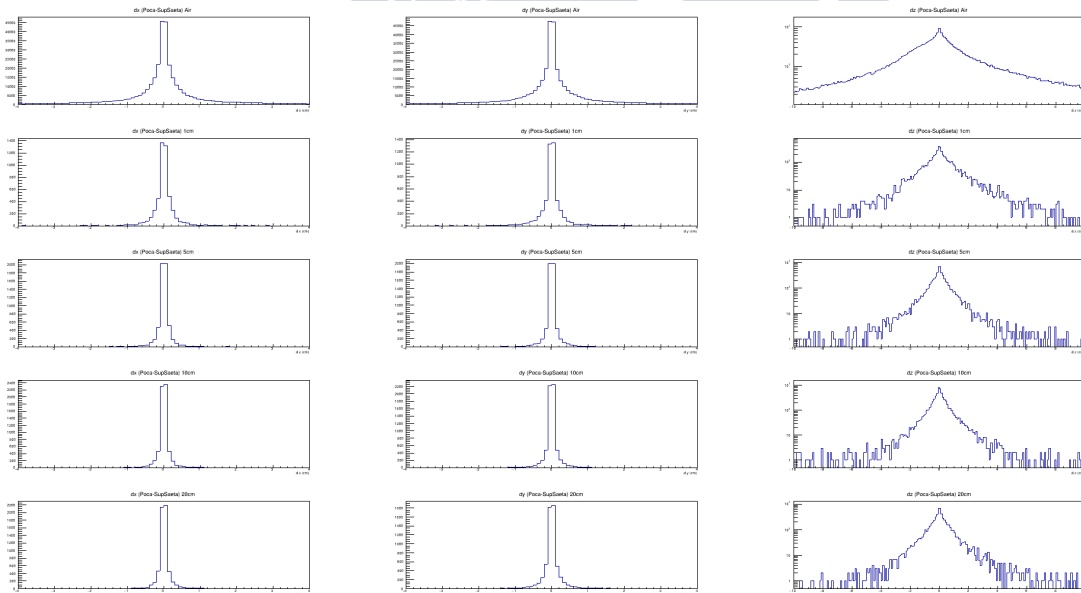

Fig. 5.8 Difference vector dr in air and tungsten for muons with an angle of 15 degrees and low energy range (0.1 GeV to 1.0 GeV). The first row corresponds to air. Second to fifth rows are tungsten with 1, 5, 10 and 20 cm thickness.

The figure shows, in columns, the components of the *dr* vector (dx, dy, dz). The first row corresponds to air. The other rows correspond to 1cm, 5cm, 10cm and 20cm of tungsten. The whole group of results, for all angles and materials, can be found in the Appendix B. These studies of the reconstruction methods with different angles try to explore if there is a privileged direction in the process.

After the study of these figures, we conclude that the reconstruction methods are independent on the incident direction of the muons. Therefore, there are no further corrections as a function of the angular components.

5.4.2 Polar deviation

Another set of observables that we studied are related to the angular variables of the scattering process. We define the scattering angle θ as the relative angle between the incident an scattered directions. We study the difference $d\theta$, defined as

$$d\theta = \theta_{\text{poca}} - \theta_{\text{ss}} \tag{5.7}$$

where θ_{poca} is the scattering angle obtained with the POCA method and θ_{ss} the one obtained using TimTrack.
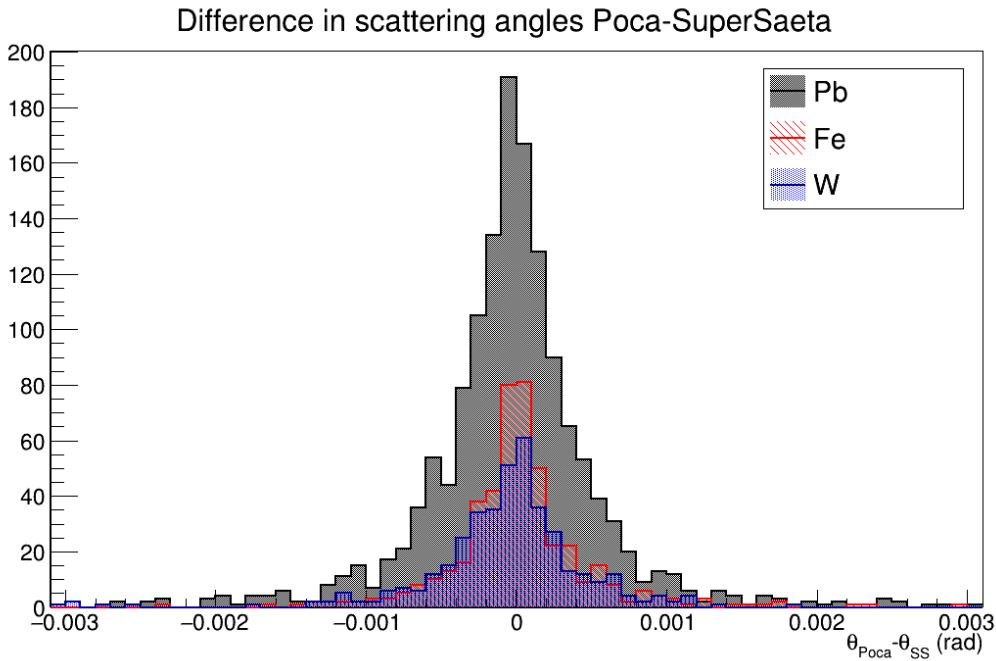

Fig. 5.9 Polar deviation, $d\theta$, for a simulation using the Simul1 configuration (run 27). No further distinction between materials can be done with this variable.

The Figure 5.9 shows the distribution of the $d\theta$ variable for the three different materials, Fe, Pb and W. To test the efficiency of the observable, we used the configuration Simul1 (run 27). As we are only interested on the scattering angle the muons were simulated as vertical. From this figure 5.9 it seems obvious that there is a dependency of the variable with the density of the material. However it is impossible to resolve them. So, clearly the variable $d\theta$ is not providing enough information by itself, as initially expected.

We proposed a new observable that combines both angular and dz differences. The definition is:

$$\frac{d\theta}{dz} \tag{5.8}$$

and we named it **relative polar deviation**. We performed a systematic study of this observable for the three materials with the parameters specified in Table 5.4 but considering only vertical muons. The evolution of the observable and its dependency on material and energy can be seen in Appendix B.

For this study we used the Simul2 configuraion. Therefore, the sample materials are centred along the y axis. It seems to be logical to study the behaviour of this observable as a function of the reconstructed x coordinate. With this study we can see how the observable behaves at each individual brick.
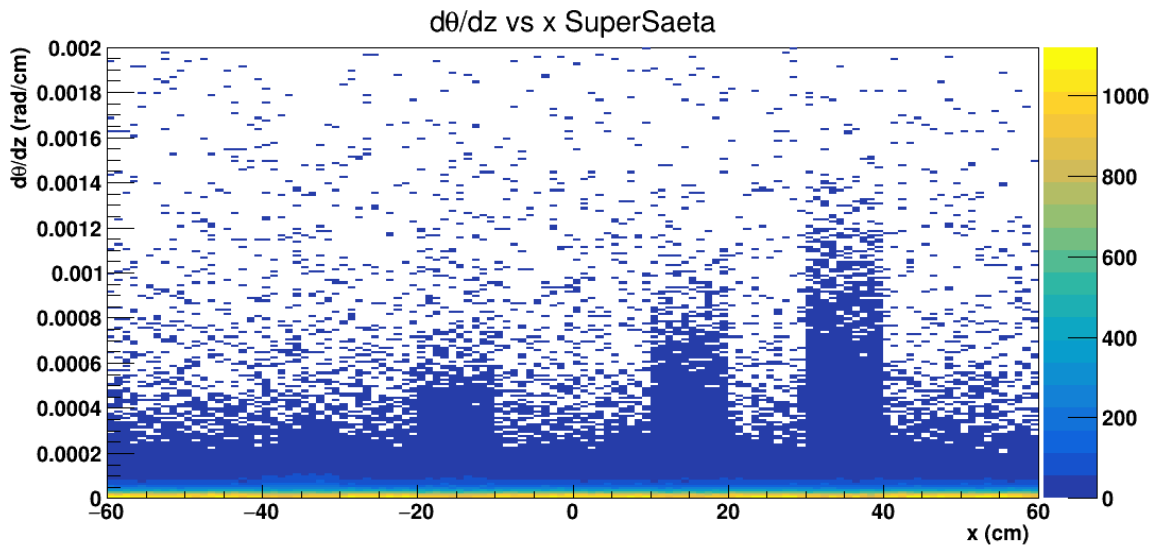

Fig. 5.10 Observable $d\theta/dz$ as a function of x_{ss} for vertical muons on W with intermediate energy range.

We show Figure 5.10 as an example of the observable for the case of vertical muons over W (1cm, 5cm, 10cm and 20cm) with an energy in the intermediate range. As it can be seen,

there is a clear dependency on the thickness of the material. Also, the position of the bricks are delimited and can be distinguished clearly.

One of the conclusions that we take from this figure is that, with this observable, the bricks appear above a certain background level. The systematic study of this images with different materials and energies could provide a threshold value for each material. In Fig.5.10 can be seen that there is a value around $3 \cdot 10^{-4}$ rad/cm that is compatible with a background. This means that there is a threshold that might depend on the material. This background can be used to perform further cuts in the reconstruction method an maybe to improve identification.

From Fig.5.10 it is possible to extract information of the observable for each individual brick. We took these histograms and performed an integration of the values along the *x* position of each brick.
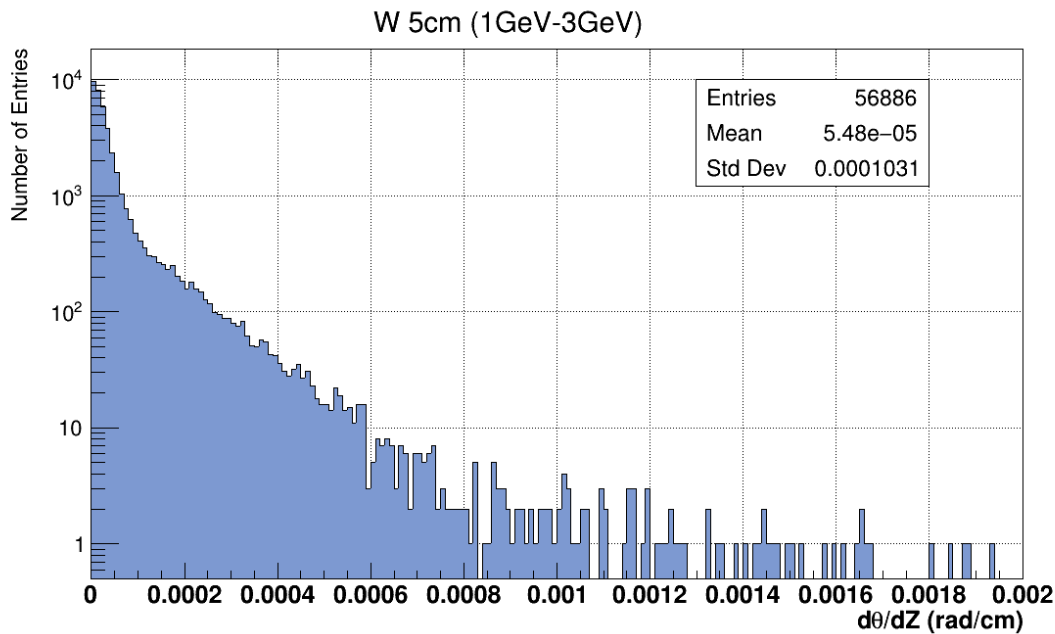

Fig. 5.11 Integrated $d\theta/dz$ in a brick made of tungsten with 5cm thickness. The figure is the result of a projection of the observable in the interval $-15 < x < -5$ cm.

As an example, the Fig.5.11 shows the integrated relative polar deviation of muons of intermediate energy through a tungsten brick of 5 cm thickness. This histogram is the result of the integration of all the values of the observable in the interval $-15 < x < -5$ cm. In this figure, it can be seen a change in slope at around 0.0001 rad/cm. The range below this value is considered as background. We performed a systematic study of the evolution of the observable isolating each brick.

The rest of figures done in the systematic study with other materials and energies are found in Appendix B.

Range	Material	Background (rad/cm)	1 cm	5 cm	10 cm	20 cm
	Fe	7	b.b.	b.b.	10	14
Low	Pb	7	b.b.	12	14	16
	W	7	11	13	14	16
	Fe	2	b.b.	3	4	6
Int.	Pb	2	b.b.	4	6	7
	W	2	3	5	7	10
	Fe	2	b.b.	b.b.	b.b.	b.b.
High	Pb	2	b.b.	b.b.	2.5	3
	W	2	b.b.	3	3.5	4

Table 5.5 Position of the maximum cut for the values of $d\theta/dz$ for different materials and energies. The entries must be multiplied by a factor 10^{-4}. In the table is shown the value of the background. The values are taken from the collection of figures similar to Fig.5.10. In the table, b.b. stands for *below background*.

As it can be seen in Table 5.5, the position of the background is dependent on the energy of the muons. This table also contains the positions of the maxima for each brick obtained from the histograms similar to Fig.5.10. These values determine the upper cut to the observable for each material. With this, we associate for each material and thickness a range of values of the observable $d\theta/dz$.

5.4.3 Spherical angle observable

Next proposed observable is related to the angle between the scattering directions provided by TimTrack and POCA. We named this relative angle between the methods as spherical angle, and we denoted it as $d\Omega$. The definition of the observable is made by the use of scalar product between both out trajectories:

$$d\Omega = \arccos(\vec{v} \cdot \vec{w}) \tag{5.9}$$

where $\vec{v}$ and $\vec{w}$ are the direction vectors of the out trajectories with POCA and TimTrack methods. Figure 5.12 is divided in two sides. In one side, it is shown the value of the relative angle between methods, $d\Omega$. As with the other observables, it was made a systematic study of the dependency of the solid angle with materials, energies and angles of incidence to determine if there is a threshold value of the observable that allows us to cut the final
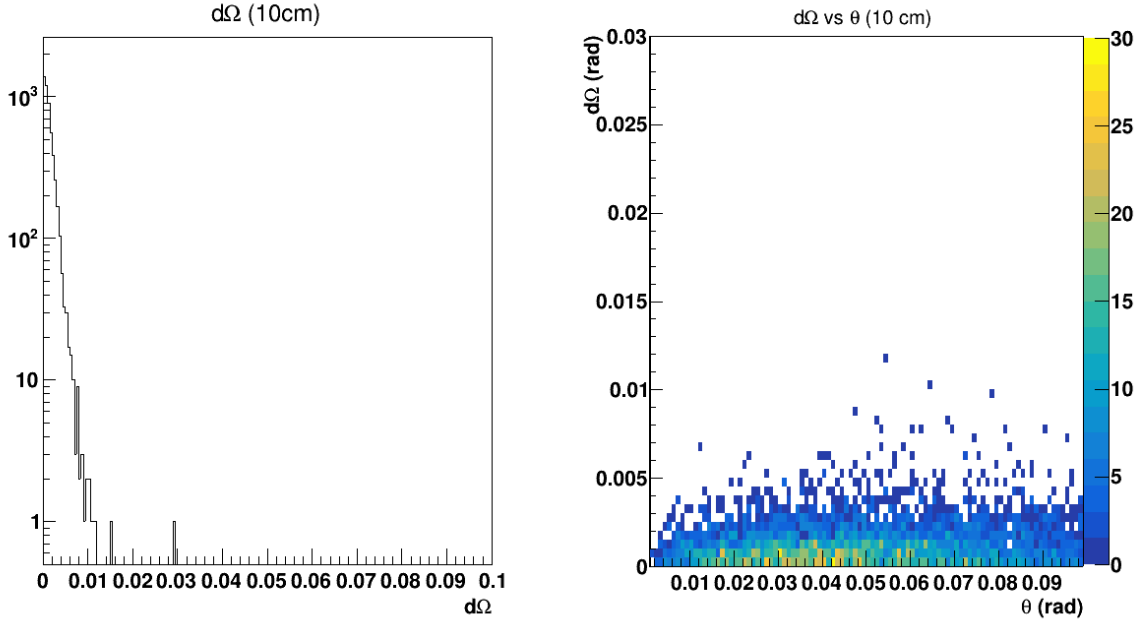
Fig. 5.12 Observable $d\theta/dz$ as a function of x_{tt} for vertical muons on W with intermediate energy range.

reconstruction methods and separate between materials. The right side of the figure shows the dependency of the solid angle with the scattering angle of the muon. This histograms are devoted to find the muons that provide the best information in the scattering process.

The complete set of histograms can be seen in Appendix B, where the reader can observe the evolution of the observable with energy and the different materials.

5.4.4   The metric observable

This kind of observable was already defined by C. Thomay et al. [47]. In that work, the track parameters and the position of the deflection vertex are calculated at the same time by the minimisation of a chi-square type operator. Their experimental set-up consists on six RPC detectors distributed three up and three down. With this, the operator to be minimised has the form:

$$E = \sum_{i=1}^{3} \frac{\left(h_i - (v_x + k_{x,upper} \cdot t)\right)^2}{\sigma_{h_i}^2} + \sum_{i=4}^{6} \frac{\left(h_i - \left(v_x + k_{x,lower} \cdot t\right)\right)^2}{\sigma_{h_i}^2} \qquad (5.10)$$

with $t = z_i - v_z$ and analogously made for the y component. Using the Minuit [41] minimisation package provided by Root they obtain the values of the vertex positions (v_x, v_y, v_z) and the track parameters k taking as an input the position of the planes z_i, the measured hits h_i, and their uncertainties σ_{h_i}.

Also, they defined an observable called weighted metric distance as:

$$\tilde{m}_{ij} = \frac{|\mathbf{v}_i - \mathbf{v}_j|}{(\theta_i \tilde{p}_i) \cdot (\theta_j \tilde{p}_j)} \tag{5.11}$$

where $\mathbf{v}_i$ and $\mathbf{v}_j$ are the scatter vertices of two tracks in the same volume, θ_i and θ_j are the scattering angles of the muons and $\tilde{p}_i$ and $\tilde{p}_j$ are the normalised momenta. The normalisation constant was taken as $1.0\ GeV$. Using this observable it is possible to distinguish between air and *something else*. This variable is used as a discriminator to determine whether the deflection was produced in heavy materials or not.

In this thesis we performed a variation of this metric observable. Since we have two methods to compare with, we modified their definition of the observable. Instead of using the difference between two vertices in the same volume, we propose:

$$\tilde{m}_{ij} = \frac{|dr|}{(\theta_i \tilde{p}_i) \cdot (\theta_j \tilde{p}_j)} \tag{5.12}$$

where in this equation, dr is the vector formed by the difference in coordinates, θ_i is the incident angle, $\tilde{p}_i$ is the normalised momentum in the upper planes, θ_j is the scattered angle, and $\tilde{p}_j$ is the normalised momentum in the lower planes.

This change in the definition of the metric is motivated by the fact that we have two independent reconstruction methods and the difference between them could have information about the density of the material. As it can be seen, this observable includes the momentum of the particles and its scattering angle. One important thing about this observable is the fact that it is always positive and decreases with the density of the material. That is to say, air has the greatest value and the rest of materials are below that value.

We simulated this observable with the Simul1 configuration to see its behaviour in three different materials.

Figure 5.13 shows the logarithm of the metric calculated as in equation (5.12). The results that we obtain are not the same, but they are consistent to those of [47]. From this simulation can extract two main conclusions:

1. The metric has a maximum value for air, and then is always decreasing with the density.

2. It is not easy to discriminate between heavy materials, but is possible to provide a cut to improve the reconstruction of the image.

We proposed the value of the metric in air as a threshold to exclude the contribution of vertices that are reconstructed outside the bricks. In our case, we imposed the condition that a vertex is only reconstructed if and only if the value of $\ln(\tilde{m})$ is less than 12.
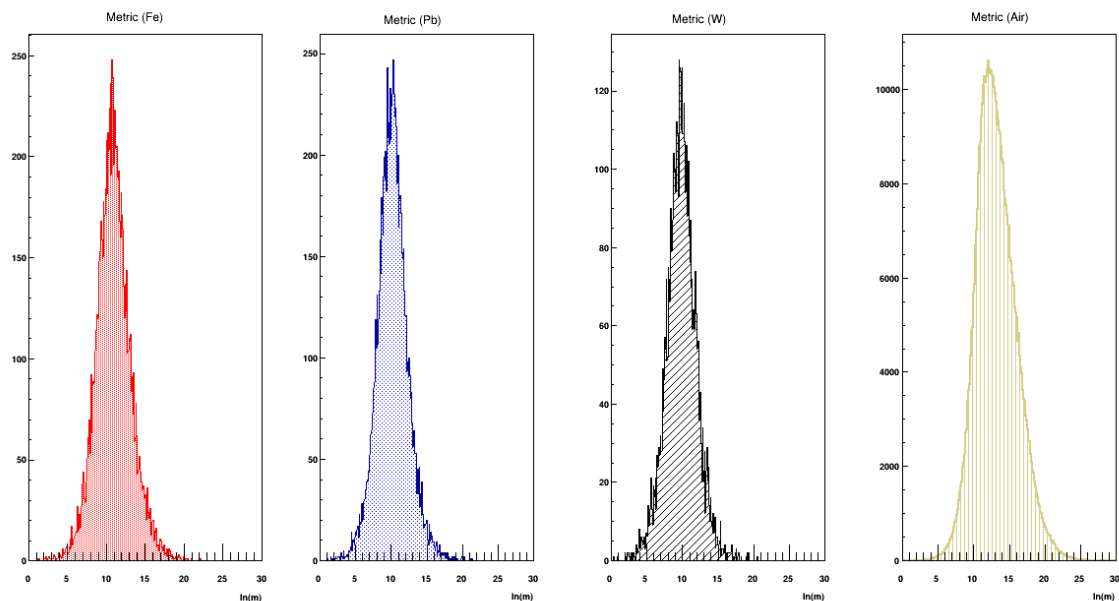
Fig. 5.13 Logarithm of the metric observable. From right to left: air, tungsten, lead, and iron. The value of air can be used as a discriminator in the reconstruction process.

To visualise the effect of imposing a metric cut in the reconstruction process, we show the set of figures 5.14 and 5.15.

In both figures, the space is divided in cubic bins of $1 \times 1 \times 1$ cm^3 volume. To observe how the reconstruction evolves, we set an additional cut in the bin content. In these figures, the top histograms have no cut in bin content. The next rows show cuts in bin content of 1, 2, and 3. This means, that we show only the bins that have a minimum population above 1, 2 or 3. Therefore, with both cuts we do exclude the contribution of air and the points outside the bricks.

As a conclusion, a cut in metric produces a more efficient reconstruction compared to the situation with no cuts. The reconstructed images of the bricks are more clear even with low cut in the bin content.

5.4.5 Time delay

We define this observable as the temporal difference between a non deflected trajectory of the muon and the reconstructed time:
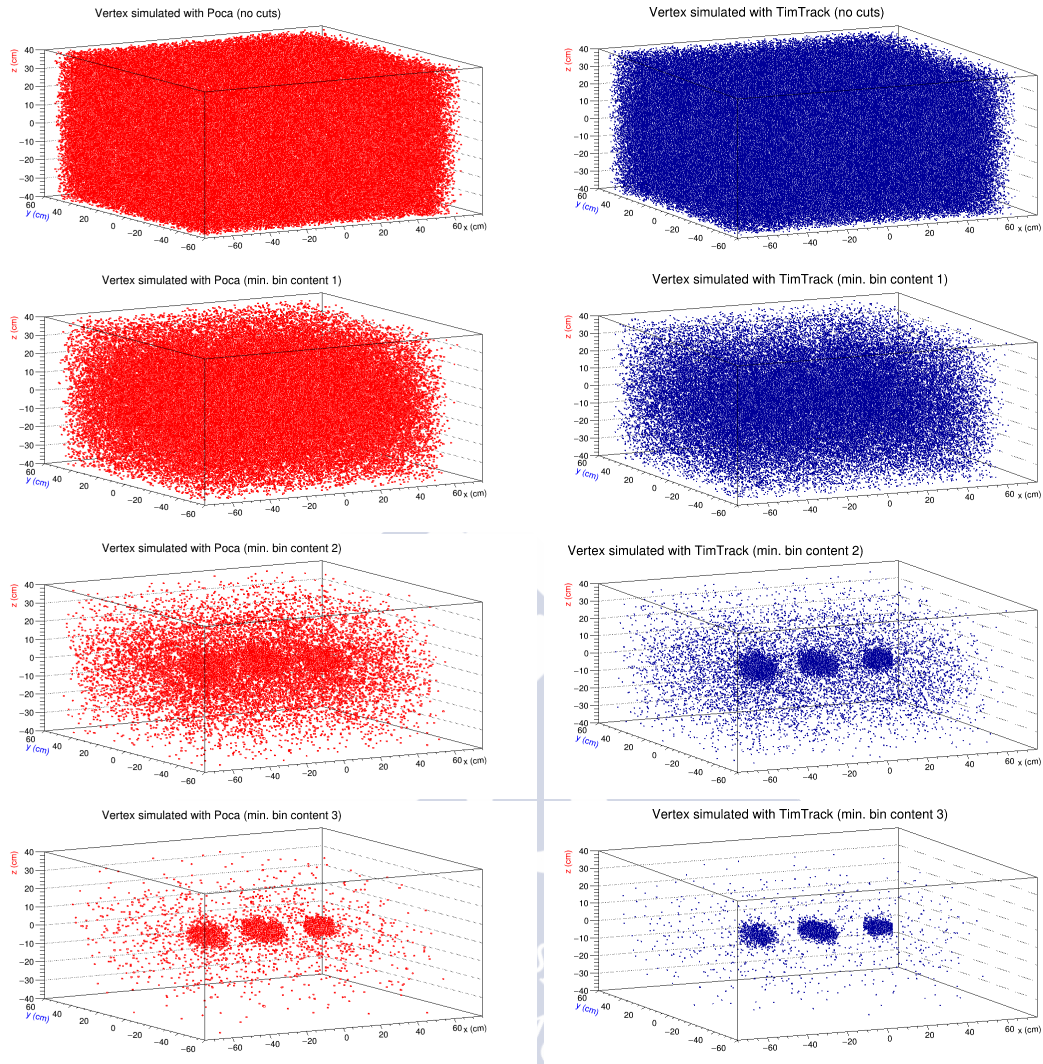
$$dt = t'_4 - t_4 \qquad (5.13)$$

Fig. 5.14 Evolution of the reconstruction methods without a cut in the metric variable. On the left side is shown the evolution of POCA and on the right side of TimTrack.

where t_4' is the estimated time in the plane number 4 (D2 in our notation) if the incident muon might follow a straight line trajectory. The t_4 is the Monte Carlo simulated time at the same plane.

We analysed the time delay for the Simul1 configuration using muons without angular restriction. In Fig.5.16 we can see the results of the time delay as a function of the scattering angle θ.

As it can be seen, the time delay dt has a parabolic dependency with the scattering angle. This correlation between the time delay and the deflection angle can be used as a further correction in the determination of uncertainties.
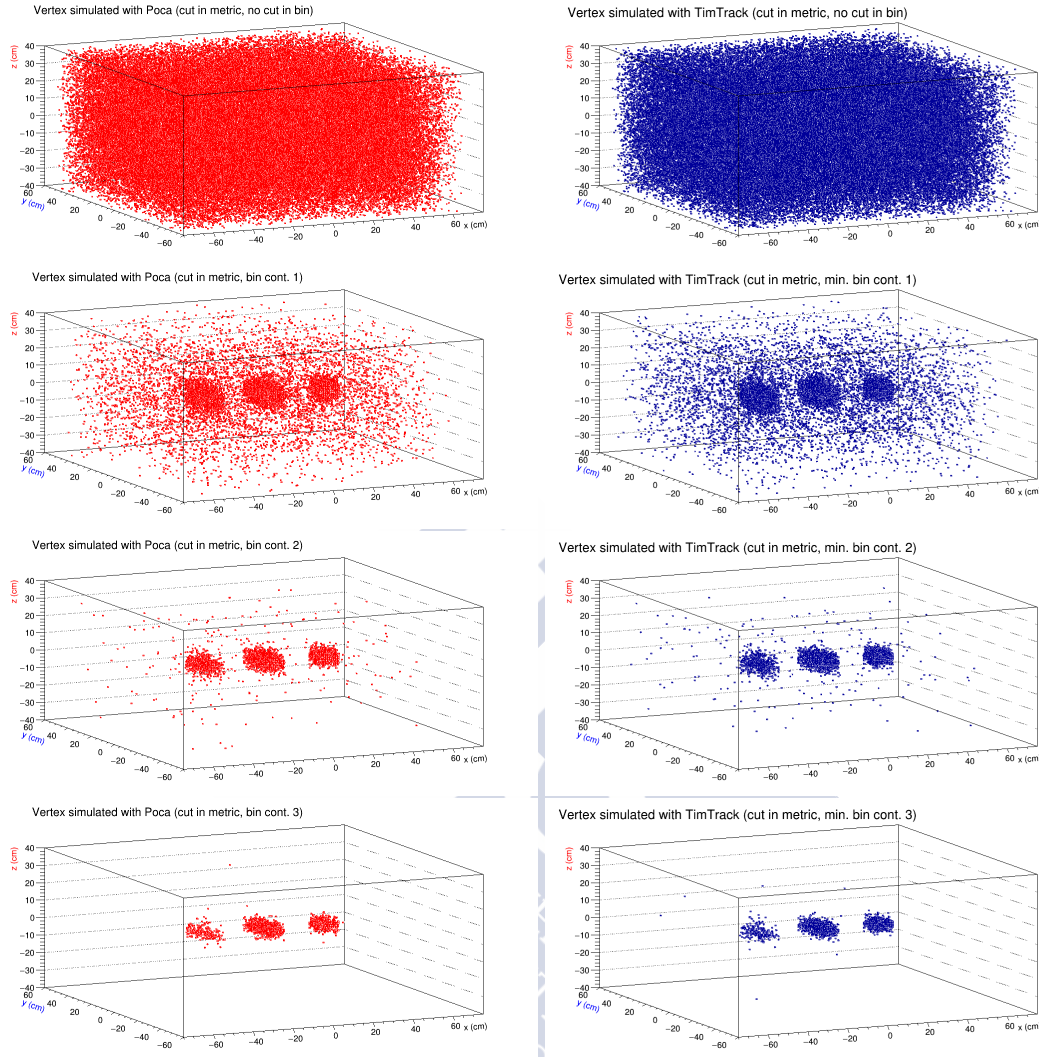
Fig. 5.15 Evolution of the reconstruction methods with a cut in the metric variable. A vertex is represented if the logarithm of the metric is below 12. The left side of the figure shows the evolution for POCA and the right for TimTrack.

5.5   Sharpness coefficient

We defined a sharpness coefficient as a possible indicator of the quality of the reconstruction methods. This definition is not unique. In our case, we defined the sharpness coefficient as the relation:

$$s = \frac{\text{number of vertices outside material}}{\text{number of vertices inside material}} = \frac{n_{out}}{n_{in}} \qquad (5.14)$$

Using this definition, a smaller coefficient means a better sharpness.

We simulated a rain of vertical muons above the top surface of the bricks. With this procedure, we ensure that the muon is always hitting the material and only the material.
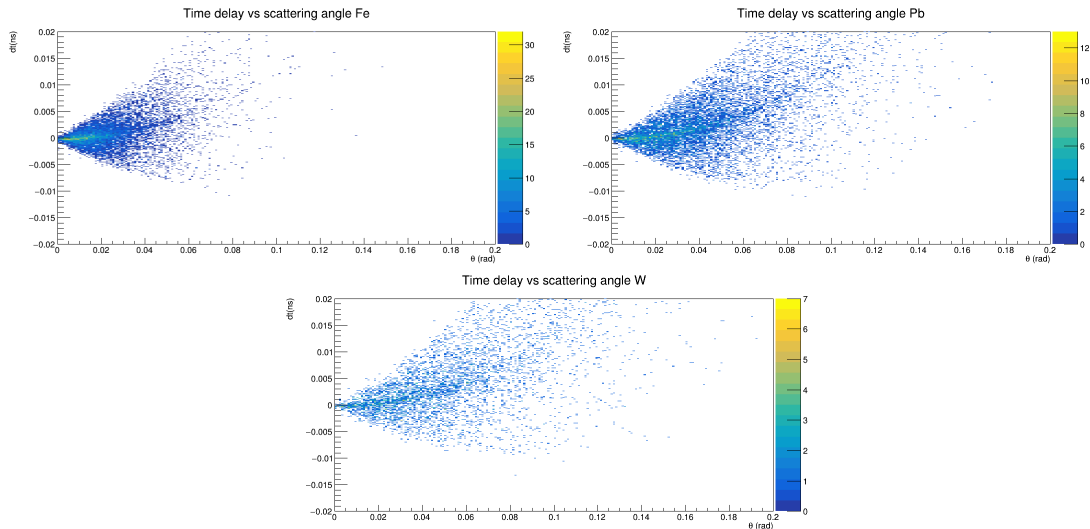
Fig. 5.16 Time delay as a function of the scattering angle.

Therefore, the scattering vertex is always inside the brick. If the reconstruction method provides a vertex outside the brick, this will translate into a bad sharpness coefficient.

For this simulations we only used bircks of Pb and W. We used the energy ranges shown in Table 5.2.

Pb	Energy	d (cm)	s_p	s_{tt}	W	Energy	d (cm)	s_p	s_{tt}
	Low		29.5	30.1		Low		23.8	24.6
	Int.	1	24.7	25.5		Int.	1	19.1	20.1
	High		24.1	25.2		High		19.7	20.2
	Low		2.6	2.7		Low		2.1	2.1
	Int.	5	1.9	1.9		Int.	5	1.4	1.4
	High		1.8	1.8		High		1.3	1.4
	Low		0.9	1.0		Low		0.9	0.9
	Int.	10	0.6	0.6		Int.	10	0.4	0.5
	High		0.5	0.5		High		0.4	0.4
	Low		0.6	0.6		Low		0.7	0.7
	Int.	20	0.3	0.3		Int.	20	0.2	0.3
	High		0.2	0.2		High		0.2	0.2

Table 5.6 Sharpness coefficients for Pb and W bricks. We used the energy ranges previously defined in Table 5.2 and four material thickness.

In Table 5.6 are presented the results of the sharpness coefficients are defined in equation (5.14). With this definition of sharpness it can be seen that the POCA and TimTrack method are quite similar.

Another strategy to improve the sharpness in the reconstruction was made by combining both POCA and TimTrack methods together. The idea consists on rescale both distributions separately and after that combine them in the same histogram. Since the most sensitive and characteristic variable is the z position of the vertex, we corrected it for the TimTrack and POCA methods. We created a new pair of variables:

$$
\begin{aligned}
z'_{\text{poca}} &= z_{\text{poca}} - \langle z \rangle_{\text{poca}} \\
z'_{\text{ss}} &= z_{\text{ss}} - \langle z \rangle_{\text{ss}}
\end{aligned}
$$

where z_{poca} and z_{ss} are the calculated z components with POCA and TimTrack and $\langle z \rangle_{\text{poca}}$, $\langle z \rangle_{\text{ss}}$ are the mean values of their respective distributions. This correction is basically a displacement to increase the statistics around the centre of the bricks. After this correction is done, both variables can be plotted together. As a result, we duplicate the statistics at each brick obtaining a better final sharpness. We performed a systematic study of the reconstruction with these two new variables in different materials for vertical muons.

Chapter 6

Calibration of the muTT detector

6.1 Introduction

In Chapter 3 we presented a general description of the Resistive Plate Chambers detectors (RPCs) and we introduced a new model of RPC called *muTT*. This new model of RPC, that belongs to the *Trasgo* family of detectors, was chosen as the reference detector of the muon tomography system.

In this chapter we will explain the novel technique of using the TimTrack algorithm as a calibration tool of the *muTT* detector. With this method, several global parameters, such as the alignment of the planes or the spatial and time resolutions, were obtained. These calibration constants are important not only to understand the intrinsic *muTT* features but also to detect and correct structural problems of the detector.

6.2 Structure of the calibration process

The *muTT* detector was designed and built by the LIP-Coimbra group. It was transported from Coimbra to O Porriño (Pontevedra, Spain) The detector is located at the DIGAFER industrial hangar, at the same place where the prototype of the muon tomography scanner is build.

After the installation of the RPCs, the first tests of the electronics were made on August 2017. Some adjustments and hardware corrections were necessary and the first stable data were taken over Fall 2017.

The four RPCs were laid in a simplified structure of the scanner. This makes the correction and calibration process easier. The structure is shown in Fig.6.1.
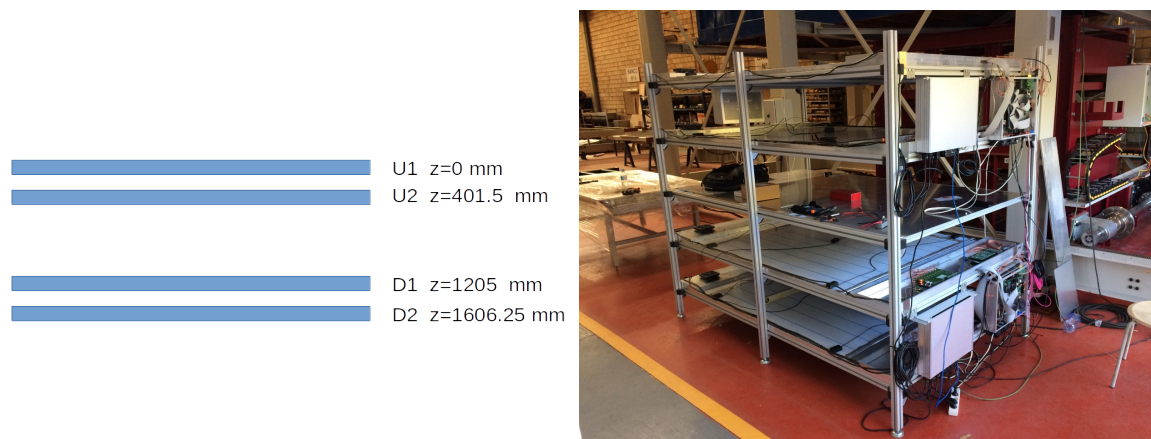
Fig. 6.1 On the left side of the figure is shown an schematic representation of the simplified structure of the *muTT* detector. On the right side, we show a picture of the detector as installed inside the industrial hangar. Same figure as Fig.3.3

As already mentioned in Chapter 3, the planes are denoted by their relative position with a letter and a number: U1, U2, D1, D2. The origin of the vertical *z*-component is chosen at the U1 plane. The calibration and reconstruction algorithms are applied to this particular detector lay-out.

The calibration process was divided into four main different components:

1. Calibration in time for the four RPC planes.

2. Global alignment of the planes.

3. Time and space corrections due to the position of the hit.

4. Further software corrections.

Space and time calibrations were made using the TimTrack method (Chapter 2) We developed an algorithm to perform the whole spatial and temporal calibrations together. After running the program, the calibration parameters are stored in text and Root files. As we will explain in next sections, the use of Root files makes the storage of the calibration parameters easier.

To perform the calibration of the detector we used an empty run, i.e., without any target material. Therefore, this run contains only cosmic ray data. We labelled this run with the number 31:

- **Beginning**: September, 27th 2017 at 9:00 am.

- **End**: October, 4th 2017 at 1:30 am.

- **Number of events**: 701478.

The DIGAFER industrial hangar is located at 30 meters above sea level with a roof protecting the structure. The average atmospheric pressure was 1018 mbar, the temperature oscillated between $18.5 - 20.0$ Celsius and the relative humidity was around 60%.

The experimental measurements are translated to physical observables using a transformation algorithm. This algorithm is developed bu A. Blanco at LIP-Coimbra. We used as an input the data already processed with this algorithm.

The events are recorded under the trigger condition of having only one particle hit in each of the four detectors. That is the most probable case of having a muon crossing the four planes. So, the trigger was set using the four U1, U2, D1, and D2 planes.

This particular run was performed after some adjustments in the acquisition process, and for that we considered as the first long-stable run. That electronic configuration was kept with no major changes until the end of the preliminary tests.

6.2.1   Time calibration

Each RPC plane of the *muTT* detector is divided in two different zones which are labelled as *left* and *right*. The general discussion of this case was introduced in Chapter 2, subsection 2.6.4, Fig.2.5. As explained, each side of the plane is connected to a different Time-To-Digital-Converter (TDC) and therefore the measurement in time is different depending on the side of the detector where the hit was detected. As there are four independent RPCs the total number of different regions is eight.

It is necessary to perform a synchronisation of all regions of the detector to obtain a correct value of time. This was made assuming that a particular region is measuring the exact value of time and then we estimate the temporal shifts relative to that region. With this, the measured times can be corrected by this temporal shift depending on the region the particle was detected.

Step 1: Left side events only

It was taken the convention that the good value of time is given by the U1 left side. Therefore, we will estimate the correction shifts to be applied relative to the U1 left, that is, we will refer all other times with respect to the U1 left time.

The first step is to synchronise the three left sides of U2, D1 and D2 respect the left U1. For that purpose, we selected only the events of the run that crossed the detector on the left side, as shown in Fig.6.2.

The calibration was made by the use of the TimTrack method. After we selected the left side events, the corresponding tracks were reconstructed following a model that introduces a
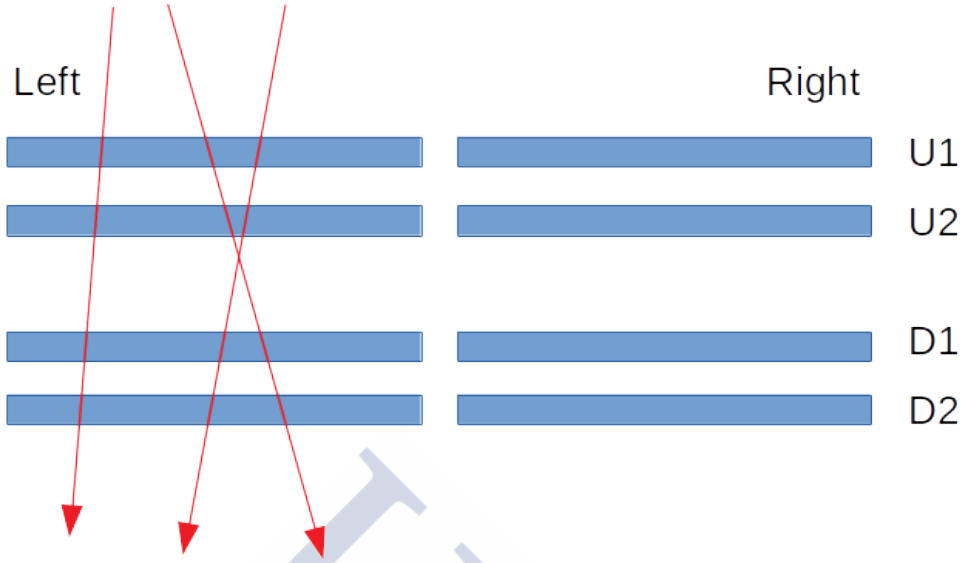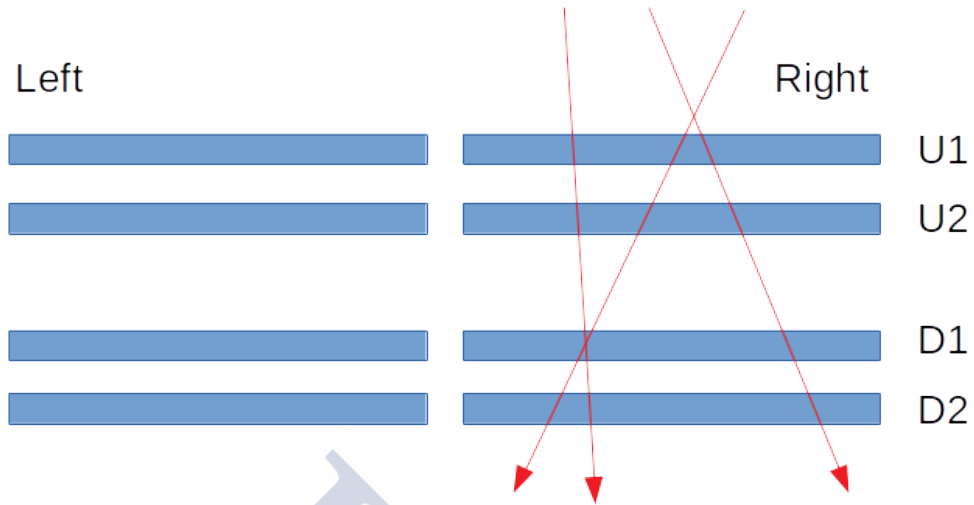
Fig. 6.2 Events that cross only the left side of the *muTT*. After the TimTrack is applied we synchronise the left side of the planes respect the U1-left.

temporal shift in planes U2, D1 and D2:

$$
\begin{aligned}
x_m &= X_0 + X' \cdot z \\
y_m &= Y_0 + Y' \cdot z \\
t_m &= T_0 + k \cdot S \cdot z + \tau_i
\end{aligned}
\tag{6.1}
$$

where x_m, y_m and t_m are data following the model, X_0, X', Y_0, Y', T_0 and S are the components of a vector called *saeta* (see Chapter 2), z is the independent variable (vertical displacement) and τ_i is the temporal shift of plane $i = 2, 3, 4$ relative to U1. By the use of the TimTrack with data following the model above, it is possible to determine the shifts τ_i corresponding to the left side of the detector. Since the model is not linear, the calculation process is iterative: an initial solution is proposed and then is modified; and this output is the new input for a next iteration. The process finishes when a convergence is reached. The convergence criteria that we adopted is based on the difference in the Euclidean module of the saetas. In this case, two iterations were enough to reach converged values.

Step 2: Right side events only

Following the same strategy as in the previous step, we look for events that occur only on the right side of the detector. With this, we refer the times to the right side of U1.

Fig. 6.3 Events that cross only the right side of the *muTT*. After the TimTrack is applied we synchronise the right side of the planes respect the U1-right.

The reconstruction was made with TimTrack and assuming that the data follow the same mathematical model as in the previous case, written in equation (6.1). Since the description of the data is the same in both cases, these two steps can be combined and computationally be executed at the same time.

Step 3: Right side U1 and left U2, D1 and D2

Once the separated sides are synchronised and referred to the first plane U1, we need to compensate or correct both sides. For that, we considered events that cross the detector from right U1 and pass through the left side of the others.

The mathematical model that describes the data is different. Since we are interested in determining the left-right global shift, a parameter τ_{1r} must to be included.

$$
\begin{aligned}
x_m &= X_0 + X' \cdot z \\
y_m &= Y_0 + Y' \cdot z \\
t_m &= T_0 + k \cdot S \cdot z + \tau_{1r}
\end{aligned}
\tag{6.2}
$$

In this case, there are no $\tau_{2,3,4}$ and finally we synchronised both sides of the detector.

We present in Table 6.1 the numerical results of the time corrections to be applied to the data depending on the detection plane.
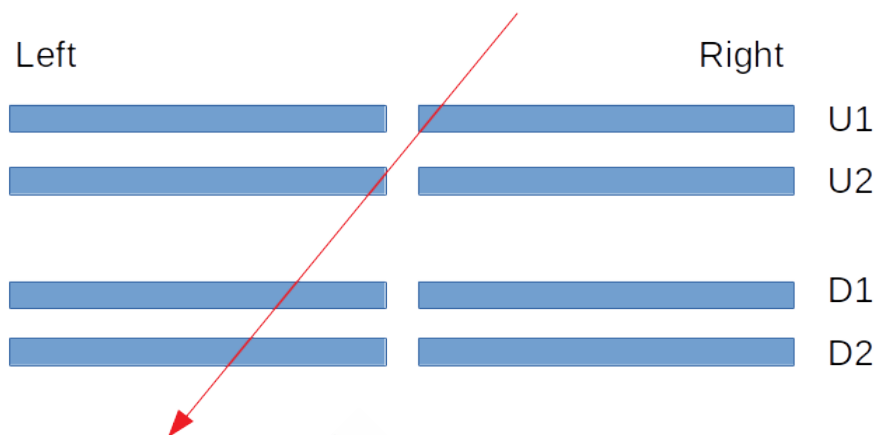
Fig. 6.4 Events that cross the detector from right to left

Plane	τ_{il} (ns)	τ_{ir} (ns)
1	0	0.126887
2	0.571072	2.16441
3	31.5284	31.9045
4	-4.66287	-4.93934

Table 6.1 Values of the time corrections for each *muTT* plane.

As we will explain later, these values are stored in a text file together with the rest of calibration parameters. This file is then used as a global calibration file for every future analysis.

6.2.2 Position calibration and alignment

The RPC planes are not perfectly aligned between them. If there is a relative shift the measurements in position are not as realistic as desired. This is the case because the reference origins are not along the same line.

This global calibration in the position of the RPCs is made using the TimTrack method. Since the time is already corrected, the model does not include time parameters. The model is non-linear:

$$
\begin{aligned}
x_m &= X_0 + X' \cdot z \\
y_m &= Y_0 + Y' \cdot z \\
t_m &= T_0 + k \cdot S \cdot z
\end{aligned}
\tag{6.3}
$$

and, as explained, it must to be solved using an iterative process. The different $\tau_{il,r}$ are already stored and we do not need to recalculate them. The variable of interest in the position calibration are the residuals:

$$\Delta x = x_{\text{rec}} - x_{\text{m}} \qquad \Delta y = y_{\text{rec}} - y_{\text{m}} \tag{6.4}$$

for each plane, where x_{rec} is the reconstructed x-coordinate with the TimTrack method and x_{m} is the measured coordinate (with similar definition for the y-coordinate).

For a given data file, the steps that we followed to calculate the total corrections in position for each plane are:

1. Open the measured data file.

2. Read the time calibration parameters. In our case, we read them from a calibration file.

3. Apply the temporal corrections to the experimental data.

4. Use the TimTrack method with the corrected data as input.

5. Construct an histogram of the residual Δx (the same procedure for Δy).

6. Obtain the mean value of this histogram.

7. We add this value to the correction that we used at that step. This is the new correction to be used in the next iteration.

8. The value of the correction is decreasing at each step. When there is no significant change in the value we considered that a convergence was reached. We stop the process.

9. We store the value of the total correction.

In Table 6.2 can be found the values of the spatial corrections that need to be applied to the experimental data.

This strategy that we use to find the total calibration in position can present some disadvantages. The main problem is the oscillation of the residuals at each step. Since the corrections are made plane-by-plane at each step, it is difficult to adjust perfectly the four planes at the same time. Another problem, related to the previous one, is that the TimTrack method could provide information of a local and not the global minimum. In this case, some spatial residuals trend to increase and other decrease. Therefore, there is not a global convergence.

Plane	Δx (mm)	Δy (mm)
1	-2.62171	-0.369186
2	1.00447	0.630083
3	2.64132	-1.28205
4	-4.04285	1.38945

Table 6.2 Summary of the spatial displacements for each plane.

All these global corrections, in space and time components, are stored automatically in a calibration file that is generated by our code. Together with this text file we will use further calibration parameters stored in Root files, as we will explain later.

Format of the calibration file

After the application of the calibration algorithm, the parameters are stored in a text file. It is used as a global calibration file. Therefore, if we want to analyse a particular run, we open the calibration file and correct the measured data with these numbers.

The structure of the calibration file is the following:

$$
\begin{array}{cccccccc}
\Delta x_1 & \Delta y_1 & \Delta z_1 & \tau_{1l} & \tau_{1r} & \alpha_{11} & \alpha_{21} & \alpha_{31} \\
\Delta x_2 & \Delta y_2 & \Delta z_2 & \tau_{2l} & \tau_{2r} & \alpha_{12} & \alpha_{22} & \alpha_{32} \\
\Delta x_3 & \Delta y_3 & \Delta z_3 & \tau_{3l} & \tau_{3r} & \alpha_{13} & \alpha_{23} & \alpha_{33} \\
\Delta x_4 & \Delta y_4 & \Delta z_4 & \tau_{4l} & \tau_{4r} & \alpha_{14} & \alpha_{24} & \alpha_{34}
\end{array}
$$

where Δx_i, Δy_i, Δz_i are the spatial corrections at plane i; τ_{il}, τ_{ir} are the temporal shifts at plane i for the left-right sides; and α_{ij} are the angular corrections between planes. In this work, Δz and the α_{ij} are not estimated. We propose this format of the calibration file taking into account possible future corrections.

6.2.3   Slewing effect

The *slewing* is an effect produced by the read-out electronics that should be corrected in the analysis of the data. When a particle interacts with an RPC, a signal is produced and its amplitude is related to the amount of charge integrated on the strips. This signal is sent to the read-out electronics and processed afterwards.

The signals that are discriminated by a constant threshold suffer a delay between the beginning of the pulse and the moment that they cross the threshold. This delay depends on the ratio between the charge and amplitude of the signal. As a consequence, large signals

arrive before than the others. Therefore, there is a time shift that is dependent on the amount of charge producing the signal.
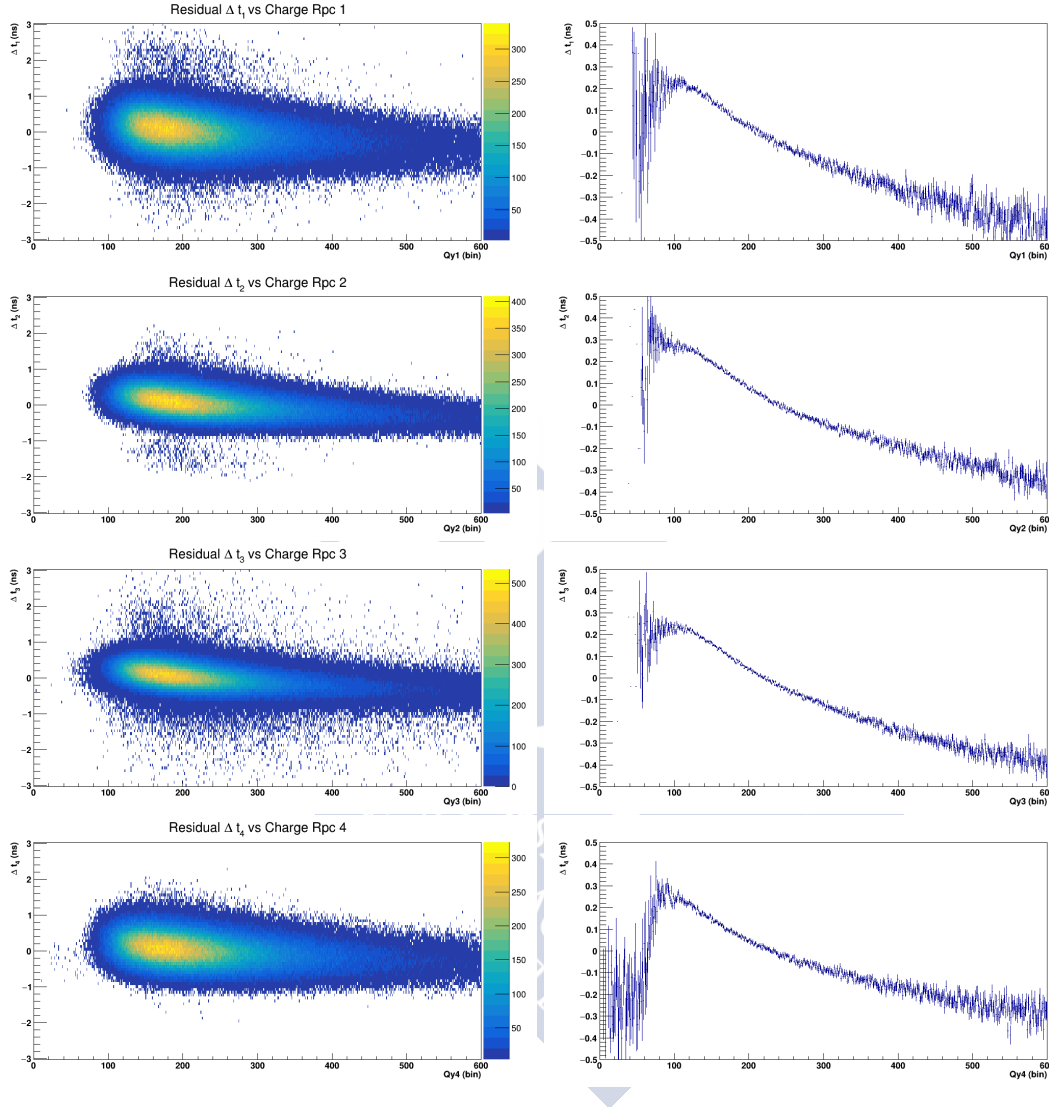

Fig. 6.5 Slewing effect. On the left side of the figure we represent the histograms of the residual in time as a function of charge for the four planes. On the right side we represent the profiles of the histograms.

In Fig.6.5 we show the two dimensional histograms that relate the residuals in time with the integrated charge in the upper layer of the *muTT* RPCs. The histograms are on the left side of the figure, and on the right side we show the profiles of those histograms. The profiles are one dimensional histograms that represent the mean value of a variable for each bin of the other.

As it can be seen, the histograms show a descendent curve shape. This means that the events with a low charge present more time delay than the more intense signals.

The profiles shown in Fig.6.5 are used as a calibration pattern for the experimental data. This means that for a certain measurement of the charge, we correct it with the value provided by the profile histogram. Therefore, we correct the measured time with the mean value of the time residuals at that charge position.

6.2.4   Corrections due to position of the hit

The corrections that we present here are complementary to the global spatial and temporal calibrations of the detector already performed in subsections 6.2.1, 6.2.2, and 6.2.3. In this case, we study the contributions to the calibration parameters of the internal components of the RPC planes.

In each RPC plane, the wires that form a strip are not equal, and therefore the parameters can vary from one strip to another and even inside the same strip. Thus, it seems to be reasonable that the spatial and time corrections must to be implemented depending on the position of the hit of the particle. The global parameters obtained in the first calibrations are further modified with these fine tuning corrections.

Time correction

Each strip is formed by twenty-four individual wires. When a particle interacts with the detector, a signal is produced an it travels through the wires to the electronic system. The length of the wires are not the same for each group os strips. In the *muTT* detector, the twenty-four wires that form a strip have a common point where the signal is collected. Therefore, if the signal is travelling by the lateral wires it covers a larger distance compared to the central wires of the strip. As a consequence, there is a delay of the signal that has to be corrected depending on the position of the hit. In Fig.6.6 we show an schematic representation of a fictitious general strip. The red line in the figure represents the limit of the strip, but there is an additional length to the collector read-out point.

To estimate this time correction, we studied the dependency of the τ_i with the *y*-coordinate of the hit, as shown in Fig.6.7. Each RPC plane has two detection layers of strips. One of them (upper layer) measures the time component. The strips of this layer are transversal respect the *y*-coordinate. This means that each individual layer has a different *y*-coordinate and therefore there is a dependency of time with the *y*-coordinate.

In this Fig.6.7 we represent the two dimensional histograms of the variables τ_i and *y*-coordinate of the hits. On the right side of the figure we present the profiles of these
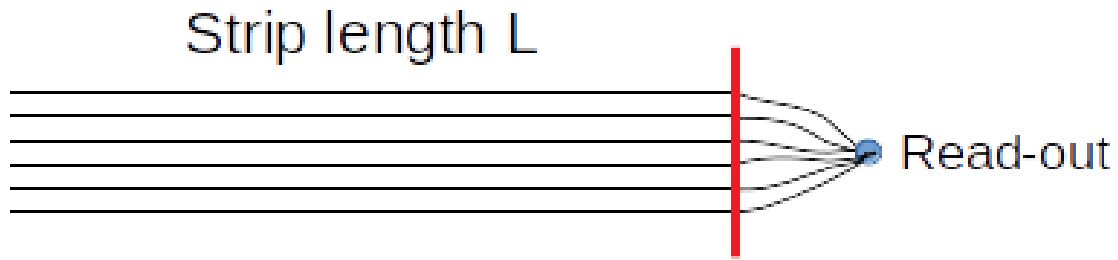
Fig. 6.6 Schematic representation of a strip. The length of the wires are not the same in a strip. The signal travels a variable length from the end of the strip (red line) to the read-out point.

histograms. This figure shows a clear parabolic dependency of the τ values with y-coordinate. This fact is consistent with the assumption that the travels more distance on the edge of the strip than in the central part. The profiles of the two-dimensional histograms provide an average correction value per wire at each strip.

A similar study is made to determine the contribution of the x coordinate in the time component. Figure 6.8 shows the two dimensional histograms of the τ_i and x variables, together with their correspondent profiles.

The figure, as expected, does not present a clear structures as for the y-coordinate case (Fig.6.7). The layer of strips transverse to the x-axis (down) does not measure time. Therefore, the dependency of time with the x-coordinate is flat.

We use both sets of profiles (τ_i with x and y coordinates) to perform further corrections on the temporal variables of the data.

Space corrections

In a similar manner as the time variable, we studied the dependency of the spatial residuals with x and y coordinates. We show the results of the study in Fig.6.9 and Fig.6.10.

On the left side of these figures are represented the two dimensional histograms that relate the residuals Δ_x, and Δ_y (as defined in equation (6.4)) as a function of the value of the coordinates x and y. On the right side we present their correspondent profile histograms.

The histograms present clear structures that could reveal some general problems with hardware and with the reconstruction method. First, it is observed that there are some problematic strips at planes D1 and D2. In Fig.6.10 are clearly seen two abnormal regions. These strips correspond to the up layer (the layer that estimates the y-coordinate) and are around the central region. Also, in Fig.6.9 it can be seen also a dead region of the down layers of D2 plane. These problems are affecting to the reconstruction algorithm and this fact can make the corrections more difficult.
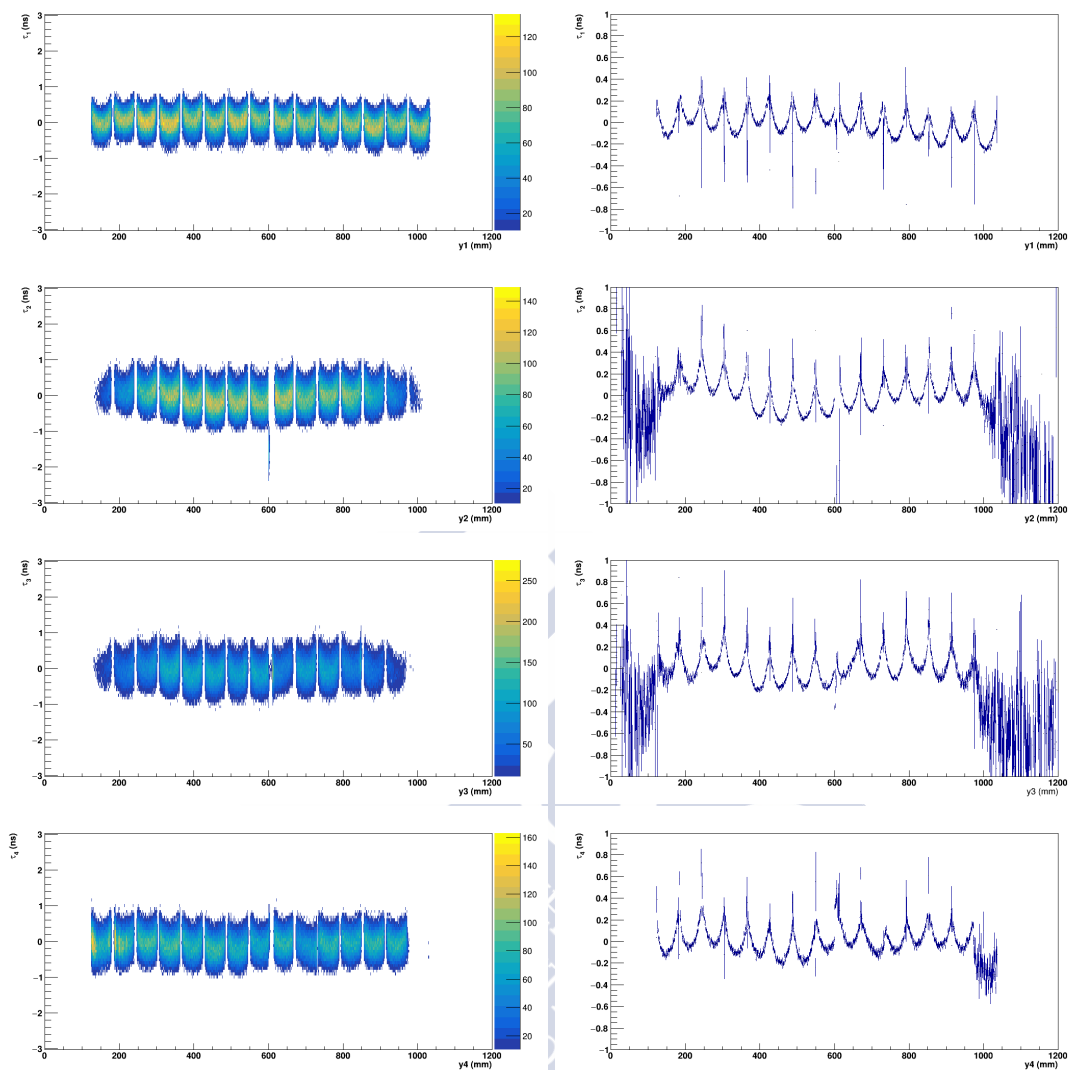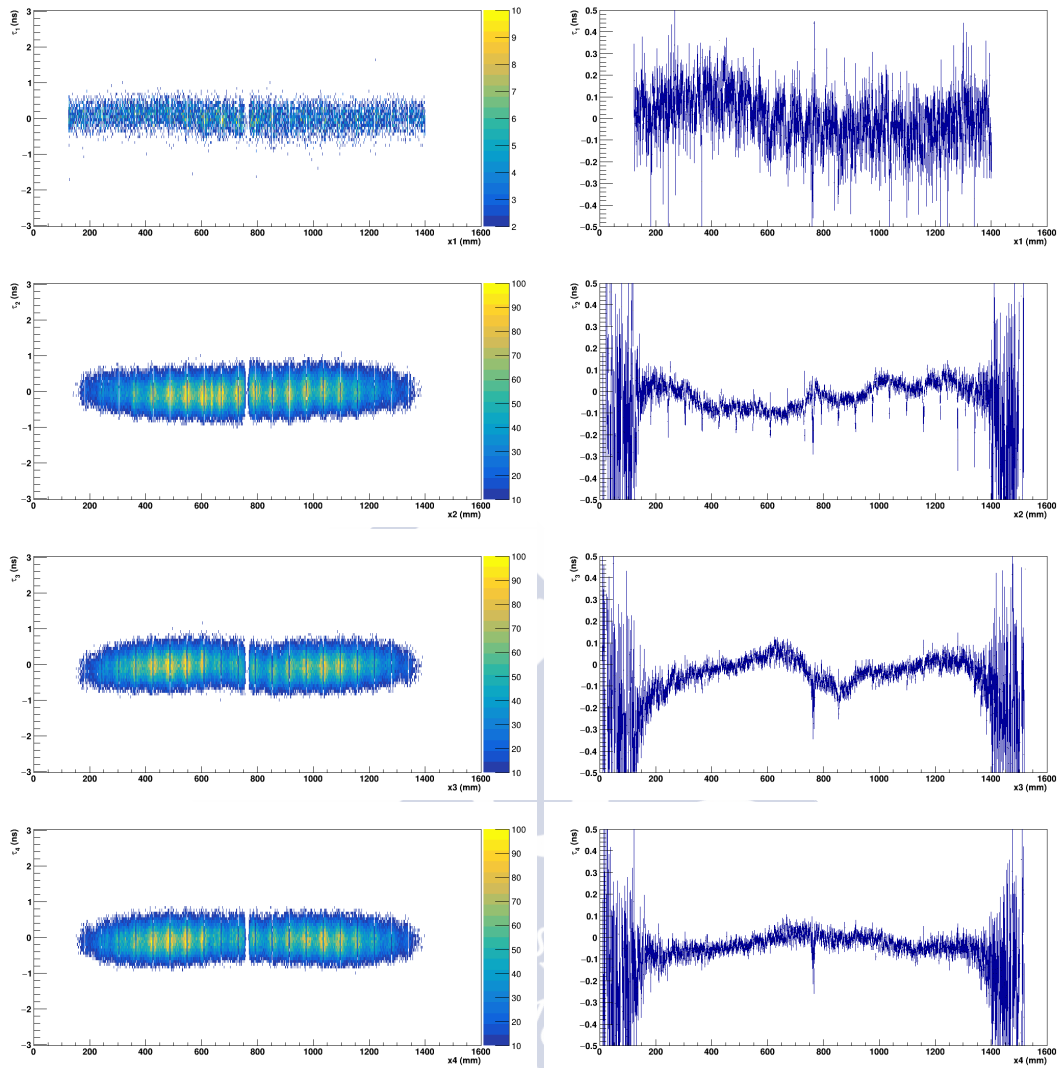
Fig. 6.7 Time correction as a function of the *y*-coordinate. On the left side are shown the histograms of τ with *y*. On the right side are the profiles of the histograms. There is a parabolic tendency of time with position.

The second problem is related to the nature of the algorithm. The reconstruction process is iterative. When the algorithm starts, the data file is opened and we apply already estimated corrections to the coordinates. After that, we use TimTrack iteratively to find the value of the residual, we add the new correction and perform a new reconstruction with a better corrected data. This works fine for not interdependent variables. The problem is that we are representing the residual of a variable against the variable that we need to correct. As a consequence, this iterative method is not able to eliminate these interdependencies. Also, the

Fig. 6.8 Time correction as a function of the *x*-coordinate. On the left side are shown the histograms of τ with x. On the right side are the profiles of the histograms. There is a parabolic tendency of time with position.

corrections are obtained using the profiles of the two dimensional histograms. This means that we correct with the mean value of the residual.

Third, the algorithm that transforms the raw information of the detector to hits can introduce some errors in the structure of data. As it was mentioned in Chapter 3, subsection 3.5.2, future improvements of this algorithm with the TimTrack method could be used.

These problems are all affecting the corrections in the spatial components. The TimTrack method by itself can not solve the structures seen in Fig.6.9 and Fig.6.10. However, after
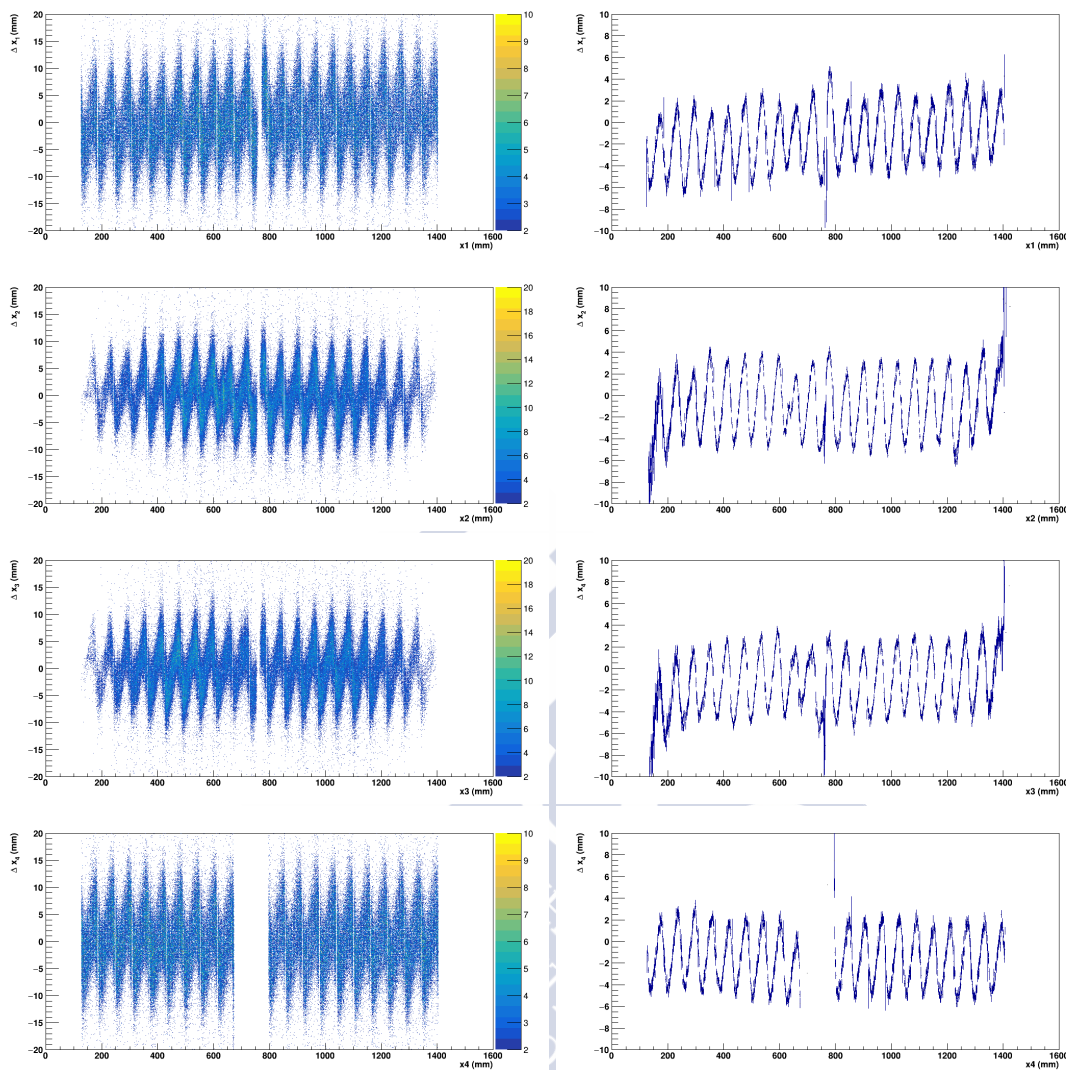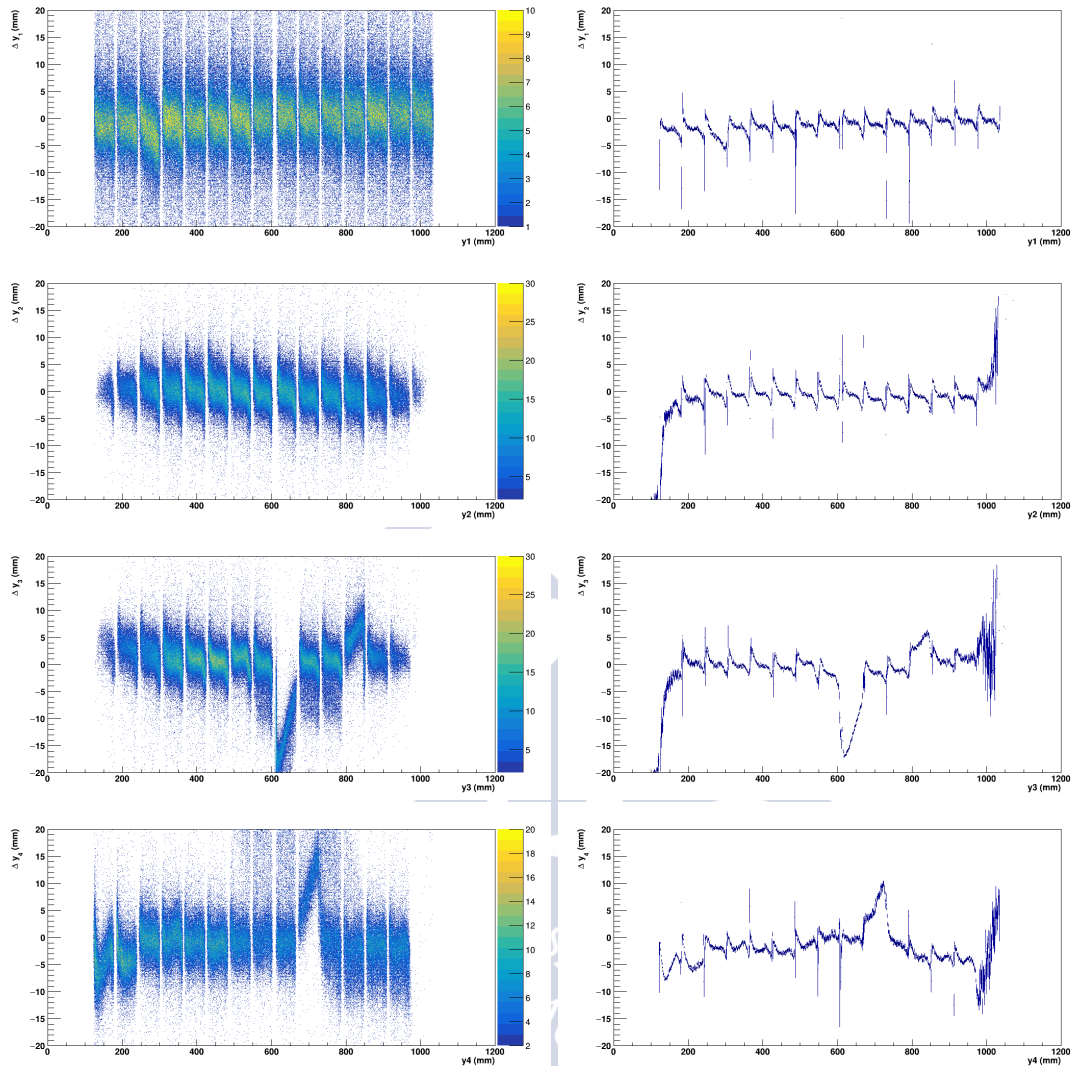
Fig. 6.9 On the left side is shown the residual Δx versus the x-coordinate of the hit for the four planes. On the right are the profiles of the histograms.

some additional software corrections the effect can be softer, but not completely removed. We present the software corrections in the next subsection 6.2.5.

6.2.5 Further software corrections

Previous subsections explained the different parts that conform our whole calibration algorithm. In particular, the last subsection devoted to the spatial corrections showed different problems in correcting the data by the use of an iterative process. To compensate or reduce the structures shown in Fig.6.9 and Fig.6.10 we introduced additional corrections to the data.

Fig. 6.10 On the left side is shown the residual Δy versus the y-coordinate of the hit for the four planes. On the right are the profiles of the histograms.

We propose here a phenomenological correction based on the experimental data. With this information, some mathematical functions are applied to the data.

In Fig.6.11 are represented the measured coordinates of the events in run number 31. It can be seen the problems with some strips at planes D1 and D2. In particular, the y_3-coordinate has an excess of statistics in two strips that could be classified as noise.

In the figures, the coordinates seem to be grouped in local structures corresponding to the different strips. Actually, this division is artificial and it is produced by the algorithm that converts the raw information into hits. In the figures can be also seen that the extremes of
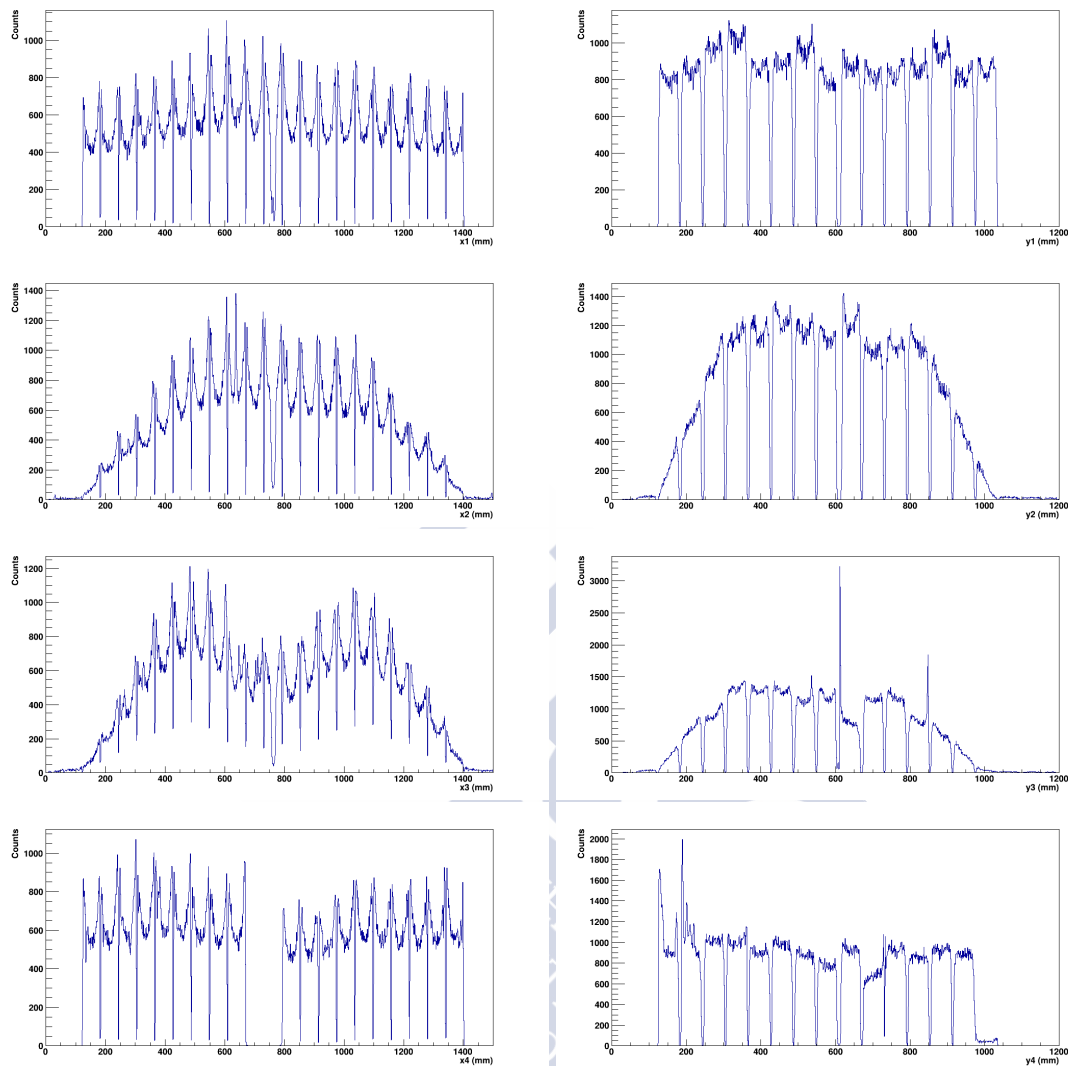
Fig. 6.11 (x, y) coordinates for the run number 31.

these groups are more populated than the central parts. This fact is also a consequence of the transformation algorithm.

One possible approach to the solution of the problem could be a correction of the data by the use of a mathematical function. This function will improve the representation of the spatial coordinates. The form of the function is unknown but must fulfil some conditions. First, the separation between these artificial groups should disappear. Second, the parabolic dependency must to be smoothed. This means that the correction function must trend to move data from the extremes of the groups and populate their central parts.

The method that we followed to find the form of those correction functions (one for x, another for y) was:

1. The centres of these artificial structures (x_c, y_c) were tabulated and stored in a text file.

2. For each group, we performed a fit of the data with a polynomial function of fifth degree.

3. We performed an average of the fit parameters of those functions.

As a result, we propose:

$$
\begin{aligned}
f(x) &= -4.93 \cdot 10^{-8} x_c^5 \\
g(y) &= -3.5 \cdot 10^{-5} y_c^3
\end{aligned}
\tag{6.5}
$$

where (x_c, y_c) is the position of the centre of the group. Once we have the correction functions we apply them to the experimental data. We read the run file and keep the measured values x_{m} and y_{m}. We then search in Fig.6.11 the group corresponding to those coordinates. After this, we apply the functions 6.5. The numbers given by the functions are the corrections that must be applied to the data:

$$
x_{\mathrm{corrected}} = x_{\mathrm{m}} - \delta_x
\tag{6.6}
$$

where δ_x is the value provided by the functions (6.5). The same procedure is applied to the y coordinate.

The representation of the corrected coordinates obtained by the use of the equations (6.5) are shown in Fig.6.12. It can be seen that the correction is not fully removing the separation between groups of data. However, the structure is more diffuse and the distribution of data inside the groups seems to be smoother that without the correction. The global effect of the corrections can be evaluated after the application of a reconstruction method.

Other correction methods can be proposed as well. For example, instead of a mathematical function we could define an algorithm based on the probability distributions and the bin content of the histograms. Future works could explore these kind of corrections. The functions that we proposed in equation (6.5) are a good first approximation to the purpose of the corrections and we use them in our reconstruction analysis algorithm.

Summary of the calibration process

This paragraph tries to summarise the whole calibration algorithm. The input of the process is the file that contains the information of the events. This file is the result of a transformation code that translates raw information of the detector into space and time coordinates.
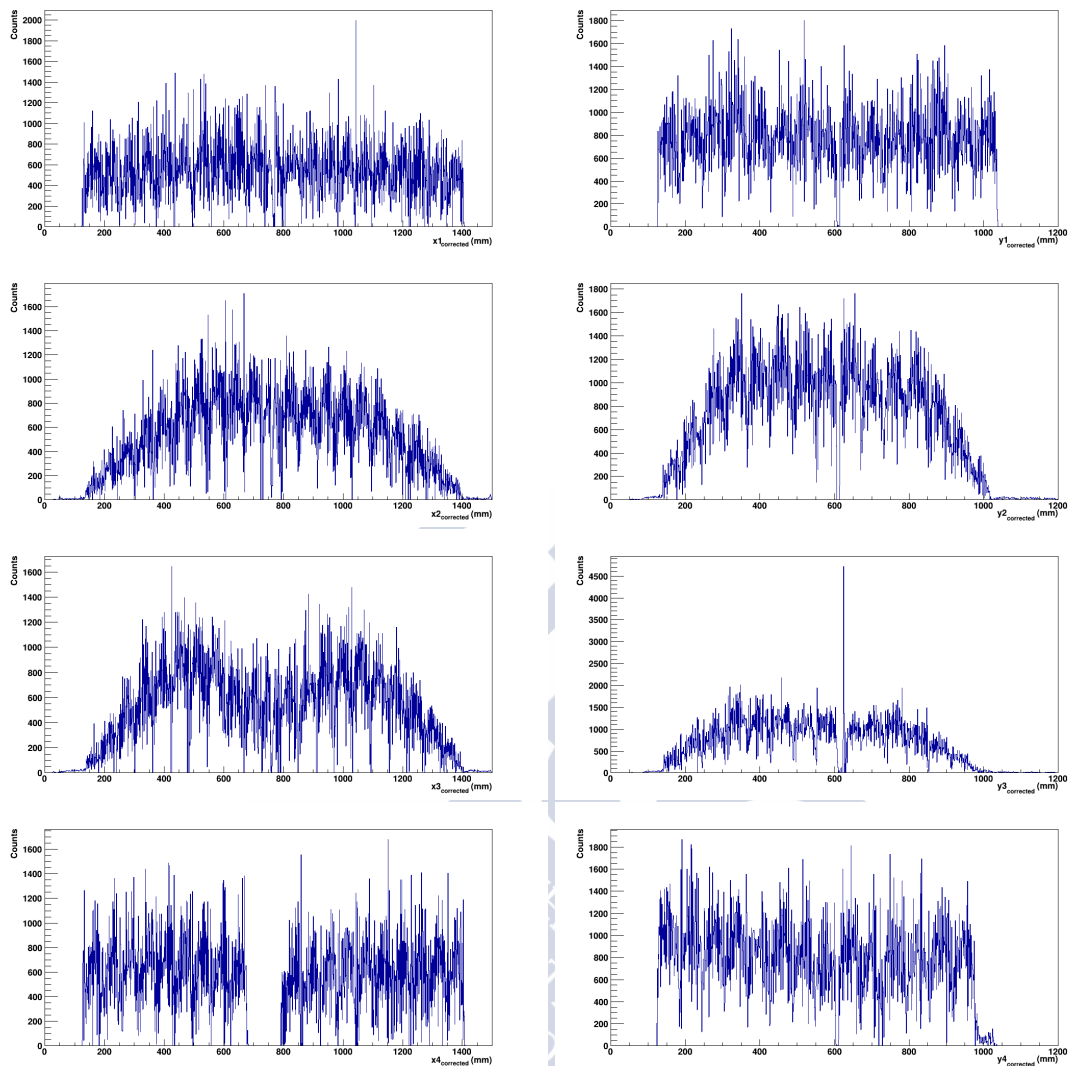
Fig. 6.12 Coordinates of run 31 after software corrections using equation (6.5).

During the process, this file is used several times inside an iterative calibration algorithm. As a standard, we used the TimTrack method to obtain the dynamical information and perform the tracking of the particles.

The sequence of steps is:

1. Synchronise both sides of the detectors.

2. Determine the global alignment constants.

3. Store the global constants in a calibration file.

4. Further corrections in spatial and time variables are made by the use of the distribution of the residuals. The values of the corrections are not stored in a text file, but in a Root file.

5. Store the corrections of the slewing effect in a Root file.

6. An additional correction of the spatial coordinates is made by the use of a polynomial function.

The calibration file (text format) and the Root files are the calibration parameters to be used on the different data inputs.

6.2.6 Reconstruction of tracks using corrected data

To see the effect of the corrections we perform a study of the trajectories of the particles. We reconstructed the tracks of the events corresponding to the run 31 by the use of the TimTrack method. The run 31 is empty, i.e., there are no material targets. Therefore, the trajectories of the particles are not deviated and we assume that all are straight.

The experimental data of run 31 are corrected with the previously estimated calibration parameters. Using the information of the trajectories it is possible to estimate some features of the detector, such as time and space resolution, after a reconstruction algorithm is applied.

The correction of the experimental data is done following these steps:

1. For each plane i we read the data: x_{im}, y_{im}, t_{im}, Q_{ui}, Q_{di}. Here, (x_{im}, y_{im}, t_{im}) are the measured position and time coordinates of the hit and Q_{ui}, Q_{di} are the charges of the layers up and down of the detection plane i.

2. We read the calibration file and read the corrections in time and space.

3. To find the value of the correction in a profile histogram, we only have to find the bin number corresponding occupied by the variable and extract the content of the bin.

4. The total correction is done as:

$$x_{\text{corr}} = x_{\text{im}} - \Delta x - \delta_x - \delta_{\text{soft}} \tag{6.7}$$

were Δx is the global displacement in x (taken from the calibration file), δ_x is the correction in x due to the position of the hit (obtained from the profiles of Fig.6.9), δ_{slewing} is the slewing correction (from profiles of Fig.6.5) and δ_{soft} is the correction made by software (equation (6.5)). In a similar way we correct the y variable. The

correction in time is similar. First, we determine the side of the detector where the hit was detected. From the calibration file we read the temporal shifts left-right and from the Root profiles in Figs.6.7 and 6.8 we extract the correction in time due to the position of the hit. Finally, the corrected time at plne i is obtained as:

$$t_{\text{corr}} = t_{im} - \tau_{il,r} - \delta_{txi} - \delta_{tyi} - \delta_{\text{slewing}} \tag{6.8}$$

where $\tau_{il,r}$ is the global correction left-right (from calibration file), δ_{txi} is the correction due to the x-coordinate (profiles of Fig.6.8) and δ_{tyi} is the correction due to the y-coordinate (profiles of Fig.6.7)

After we apply the calibration parameters to correct the experimental data, we perform the reconstruction of the trajectories using the TimTrack algorithm. As it is explained in Chapter 2, subsection 2.6.2, we use a non-linear model:

$$
\begin{aligned}
m_x &= X_0 + X' \cdot z \\
m_y &= Y_0 + Y' \cdot z \\
m_t &= T_0 + S \cdot k \cdot z
\end{aligned}
$$

where

$$k = \sqrt{1 + X'^2 + Y'^2}$$

With the reconstructed trajectories we performed the study of the spatial and temporal resolutions. These values are obtained using the distribution of the residuals in x, y, and t for the four planes.

In Fig.6.13 are shown the distributions of the residuals in time for each plane. To obtain the final value of the resolution, we performed a double Gaussian fit. First, we fitted the distribution with a Gaussian function without any restriction in data range. After this, we keep the sigma of this function and perform a second Gaussian fit. In this second fit we restricted the range of the data to 1.5 times sigma. With this, we obtain a Gaussian fit on the central part of the distribution, avoiding the possible influence of the tails of the distribution. The temporal resolution is the sigma of this second Gaussian fit.

The same study can be made for the spatial components. We used the same trajectories to extract the information about the residuals in x and y.

In Fig.6.14 are shown the distributions of the residuals of the spatial coordinates. As for the time resolution, the spatial resolutions are obtained after a double Gaussian fit technique.

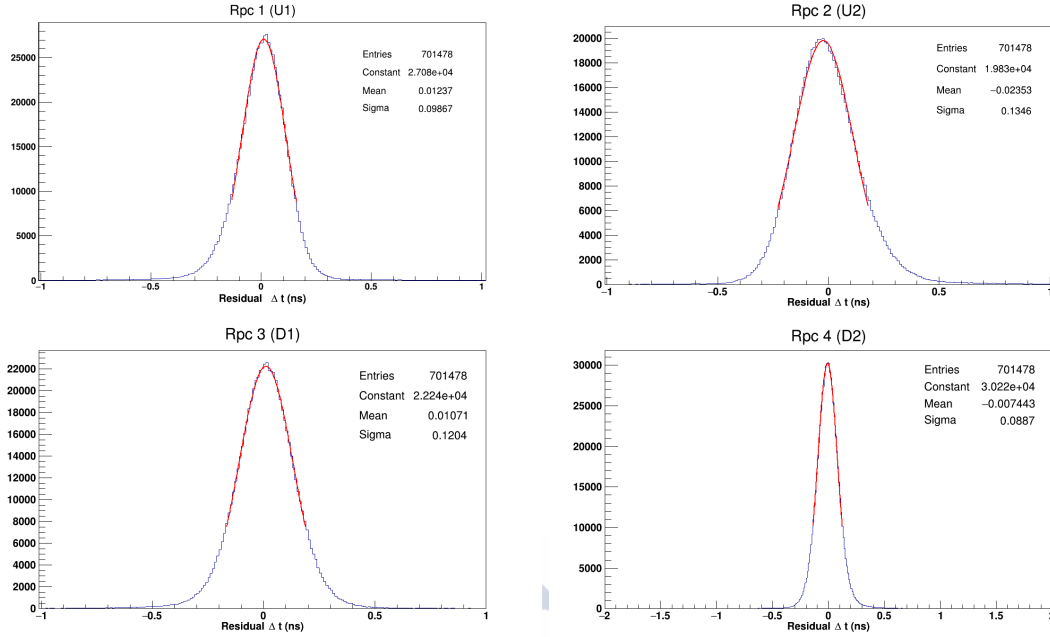We summarise in Table 6.3 the time and space resolutions for each individual plane.
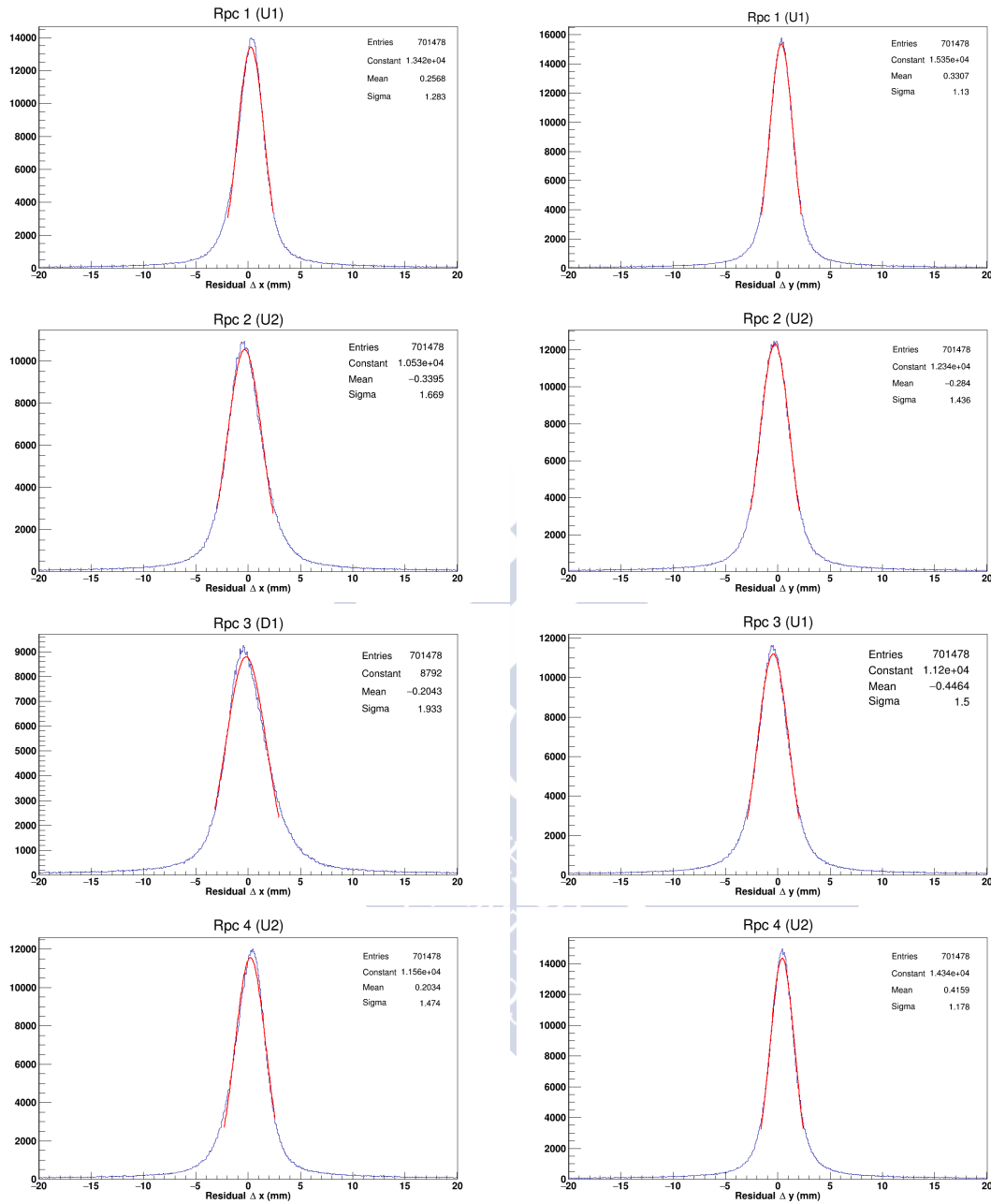
Fig. 6.13 Time resolution of the *muTT* planes after the use of a tracking algorithm. The values of the resolution are obtained after a double Gaussian fit of the distribution of time residuals.

Plane	σ_x (mm)	σ_y (mm)	σ_t (ps)
1	1.28(1)	1.13(2)	98.7(2)
2	1.67(3)	1.43(3)	134.6(3)
3	1.93(4)	1.50(3)	120.4(3)
4	1.47(3)	1.18(2)	88.7(1)

Table 6.3 Values of the spatial and temporal resolutions obtained after the TimTrack reconstruction method is applied. The first significant digits of the errors are in parenthesis.

The values of temporal and spatial resolutions provided by the manufacturer of the *muTT* are around:

$$\sigma_x = \sigma_y = 5 \; mm \qquad \sigma_t = 200 \; ps$$

for each RPC plane.

As we see, by the use of the TimTrack method we can perform tracking of particles with better temporal and spatial resolutions than the intrinsic of the *muTT*. The values that we obtained for the resolutions will be used in all the tracking analysis of experimental data instead of the intrinsic values provided by the manufacturer. The combined resolution of the

Fig. 6.14 Spatial resolution of the *muTT* planes after the use of a tracking algorithm. The values of the spatial resolutions are obtained after a double Gaussian fit of the distribution of residuals.

four RPC planes is given by:

$$\sigma_x = \sqrt{\sigma_{x1}^2 + \sigma_{x2}^2 + \sigma_{x3}^2 + \sigma_{x4}^2} \qquad (6.9)$$

and the same for the *x* and *t* coordinates. This equation provides total resolutions of:

$$\sigma_x = 3.2\,mm \qquad \sigma_y = 2.6\,mm \qquad \sigma_t = 224\,ps$$

These values correspond to the total resolution of our system.

To check the quality of the corrections we repeated the same analysis for the distributions of the spatial and temporal residuals with charge (slewing effect). With the TimTrack method on the corrected data we constructed similar histograms as in Figs.6.10 and 6.9.
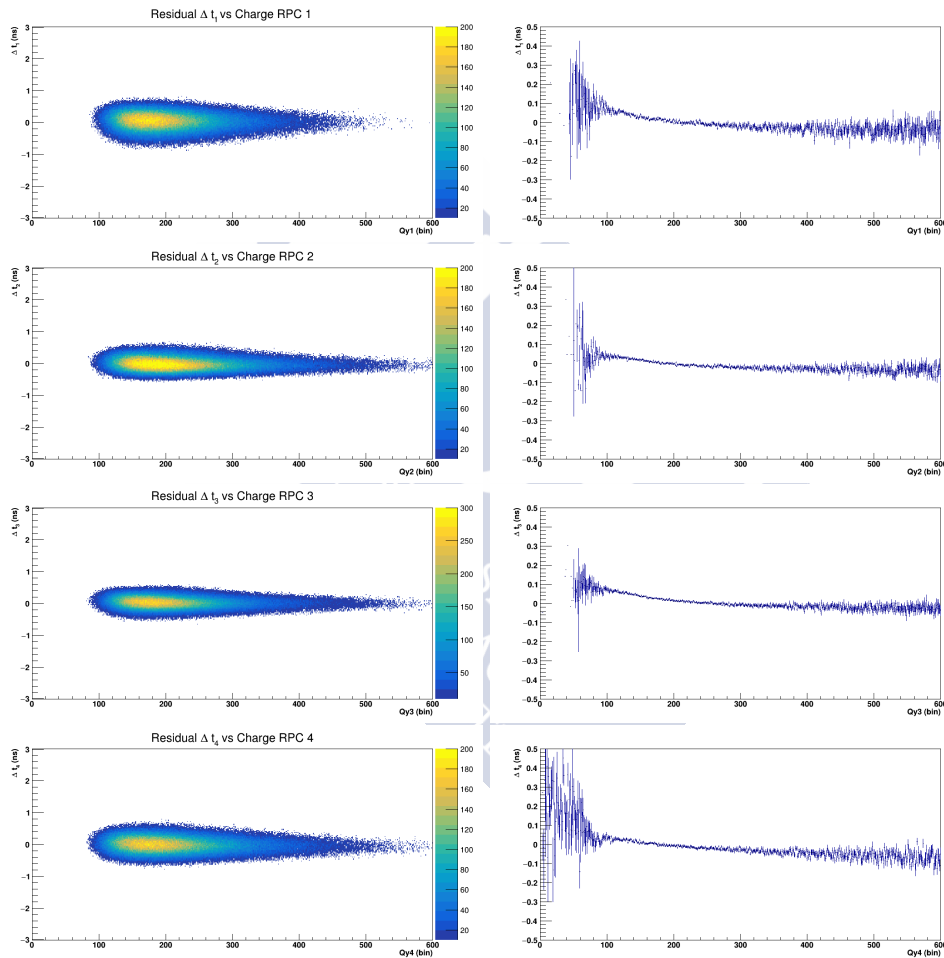
Fig. 6.15 Histogram of the time residuals as a function of charge after corrections to the data. On the right side are shown the profiles of the two dimensional histograms (left side)

As it can be seen in Fig.6.15 the histograms of the residual time versus the charge are smother. Since the correction is made using the profiles of the original histograms, we will find regions (specially in the low charge range) with a less strong correction. This effect can be seen in the profiles of the two dimensional histograms (right side of the Fig.6.15) Since

we have poor statistics in the very low charge region, the errors in the profile histograms are large. However, the effect is almost suppressed in the most populated region (from charge 100 to 300)

The same study was made for the residuals in *x* and *y*. The histograms of Δx, Δy versus the *x* and *y*-coordinates are shown in Figs.6.16 and 6.17.
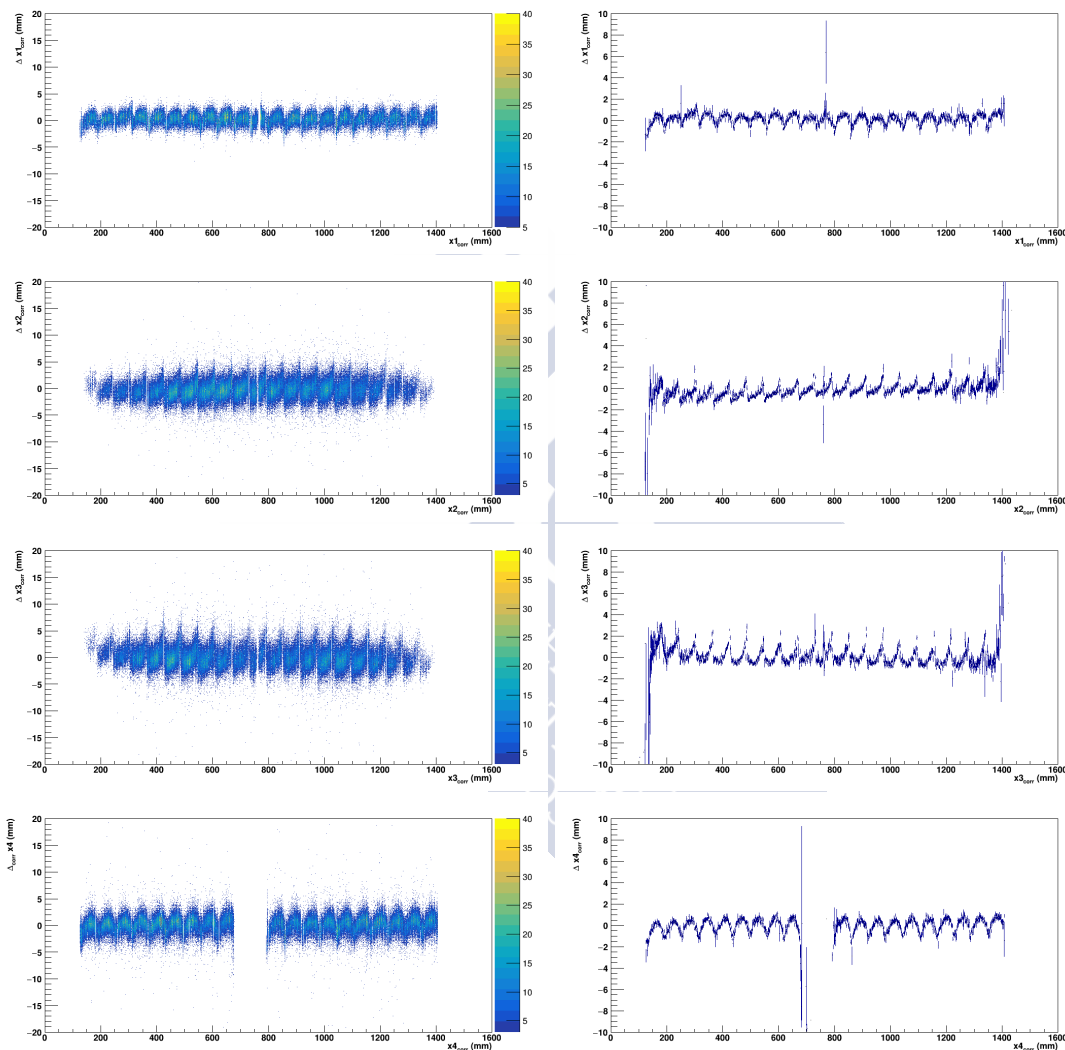
Fig. 6.16 Residuals of the *x*-coordinates versus *x* after corrections. On the left side are shown the two dimensional histograms using corrected variables. On the right side are the profiles of the two dimensional histograms.

The effects are not fully suppressed but the residuals are more centred around zero. We mentioned in 6.2.4 that these effects are complicated to suppress due to several problems, such as the iterative nature of the algorithm or the correction with profiles.
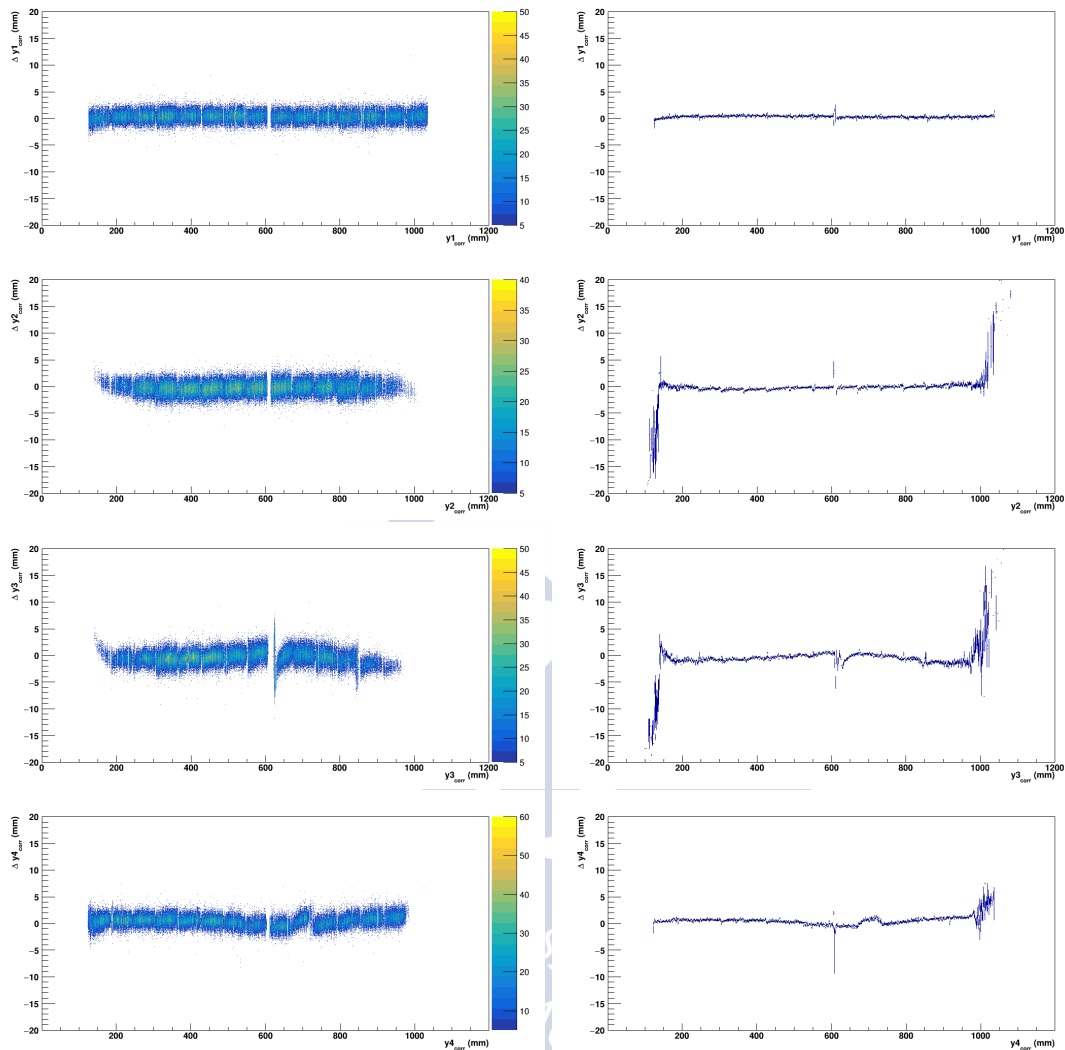
Fig. 6.17 Residuals of the *y*-coordinates versus *y* after corrections. On the left side are shown the two dimensional histograms using corrected variables. On the right side are the profiles of the two dimensional histograms.

From these figures we also notice that the problems with the regions in U1 and U2 planes are compensated after the application of the algorithm. Despite they are not perfectly corrected we find a much better structure of the data compared to the original in Fig.6.10. The effects in residuals are minimised and their contributions are oscillating around zero, in mean value. In general, we can confirm that the TimTrack method is a powerful tool to perform calibration and reconstruction analysis.

Chapter 7

Analysis of the experimental data

7.1 Introduction

The general structure of the *muTT* detector was introduced and described in previous chapters. We performed cosmic rays simulations using the RPC planes. These simulations were useful not only to understand the features of the RPC planes, but also to propose several physical observables relevant to the muon tomography process.

In Chapter 6 we already dealt with real data from the *muTT* detector. These data were used to perform the calibration of the RPC planes. To do that, we worked with an empty run (named run 31) together with a reconstruction algorithm. After this, we provided the global calibration constants of the *muTT* detector.

This chapter is the natural continuation of the calibration process. Here, we use the corrected real data to reconstruct shapes of different test materials.

The section 7.2 is a general description of the run number 27. This run is taken as a reference of developing the image algorithm in muon tomography. This run was already used in Chapter 5. In that chapter, we used the run 27 to check the reconstruction algorithms and to simulate some observables related to the scattering angle.

To construct the images of the test objects we used two independent algorithms: the Point Of Closest Approach (POCA) and TimTrack. The POCA algorithm is commonly used in other muon tomography works, as for example [47] or [43]. A more detailed description of the TimTrack algorithm is presented in Chapter 2.

The other part of the chapter is devoted to the material identification. We compare the behaviour of several physical observables to the simulations performed in Chapter 5. The combination of the reconstruction methods could provide information about the type of materials and not only about its shape.

7.2 Description of the run number 27

The *muTT* detector was checked and calibrated after the analysis of several test runs. Modifications in electronics or software acquisition were done based in these analysis. As a result, the detector was more stable and efficient.

The run number 27 was one of the first stable runs acquired by the *muTT* detector. The detector configuration for this run is shown in Fig.7.1. As it can be seen in the figure, it consisted on three bricks; Fe, Pb, and W, placed along the *x*-axis, as it we will be explained in more detail later.

General considerations

Some remarks are necessary before continuing with the chapter. First, we take into account that the tests of the reconstruction algorithms and other further analysis were done for the reduced structure of the *muTT* detector. This particular set-up consists on four RPC planes and an aluminium plate in the central part. The RPC planes were placed along the vertical axis positions indicated in Table 7.1.

Name	z (mm)
U1	0
U2	401.5
D1	1205.0
D2	1606.25

Table 7.1 Relative positions of the planes used in the reconstruction process. The origin of the *z*-coordinate is in U1.

As indicated in the table, the distances are given in millimetres and we set the origin of the vertical *z*-component at the U1 plane. In previous chapters we used a similar reference frame. This particular choice is not affecting the reconstruction algorithms. In this first part of the chapter we are interested on the reconstruction of the shapes of the materials. This problem is symmetric in the vertical component, and therefore the produced images are not affected by the position of the origin.

As it is shown in Fig.7.1, the origin of *x* and *y* coordinates is located at the top-right corner of the picture. The positive directions of the variables are indicated in the figure.

Technical description of the run

The run was registered with the *muTT* detector at the DIGAFER industrial hangar at O Porriño (Pontevedra) In this paragraph we provide the details of the the run number 27:
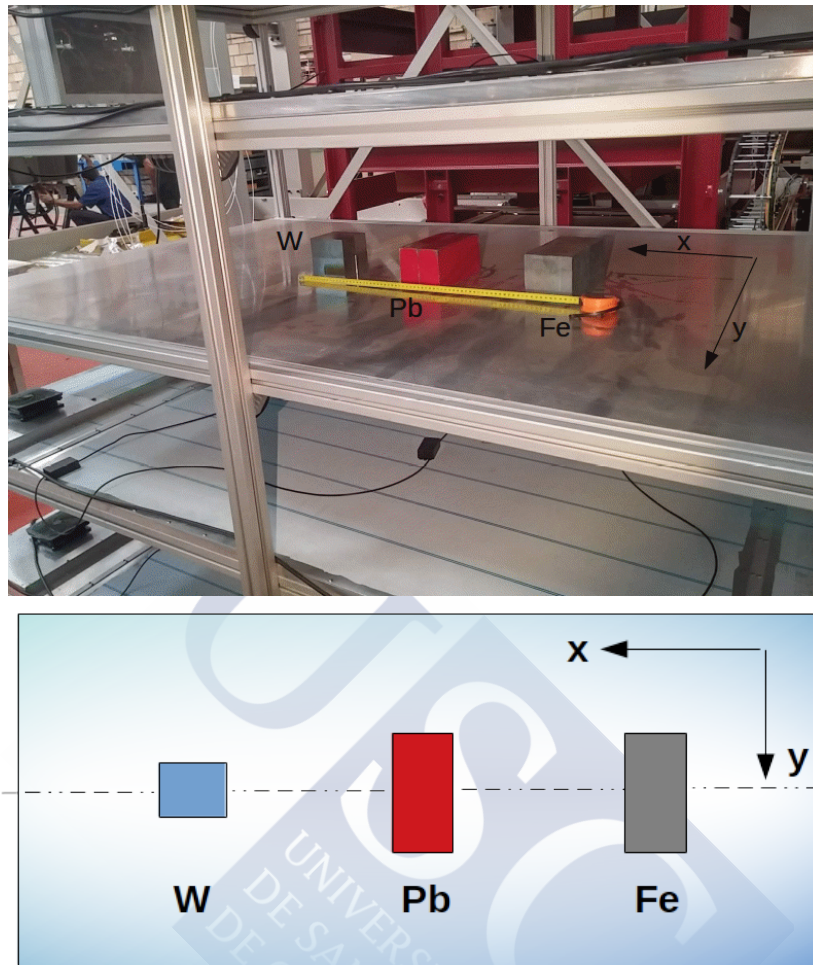
Fig. 7.1 Experimental configuration of the run number 27. Three bricks are placed along the *x*-axis. The origin of coordinates is placed at the top-right corner. The centres of the bricks are aligned with the *y*-axis.

- **Beginning**: September, 8th 2017 at 9:00 am.

- **Number of registered events**: 193k.

- **Trigger**: One and only one hit in U1 & U2 & D1 & D2.

- **Rate of registered events**: 5 Hz.

In Fig.7.1 it also can be seen the mechanical configuration of the run. On the aluminium plate are placed three different bricks: one is made of iron, another one is made of lead, and a third one is made of tungsten. The Fe and Pb bricks are of $10 \times 20 \times 10 \ cm^3$, while the W one is $10 \times 10 \times 10 \ cm^3$. The separation between the bricks is around 20 cm, and the Pb

brick was placed at the central part of the aluminium plate. The centres of the bricks are aligned with the *y*-axis at 63 cm to the border of the RPC.

7.3 Reconstruction of the targets

The reconstruction of the bricks of the run 27 was done based on two different methods. One is the Point Of Closest Approach (POCA) and the other is the TimTrack algorithm.

The POCA algorithm takes as input the trajectories before and after the interaction of the particle with the material medium. These trajectories are parametrised as straight lines. The method finds the point such as the distance between both lines is minimal. This point is the vertex of the interaction of the particle with the material.

One of the main advantages of the POCA method is that it can be easily programmed. The solution of the vertex is found only by geometrical arguments and the equations involved are quite simple. In Chapter 5 we presented the formulation of the POCA algorithm. The main disadvantage of the POCA algorithm is that it does not account for the correlations between variables. Therefore, in this analysis the POCA algorithm does not provide information about the uncertainties of the reconstructed vertex nor about the correlations between variables.

The TimTrack method was developed in Chapter 2. As it was mentioned, the algorithm is based on the least squares method and it provides information about the trajectory of a particle. Since these parameters are obtained after a statistical fit, we have information about the correlations between variables and the uncertainties of the parameters. TimTrack takes as input the experimental hits of the detectors and uses a model that describes them. As a result, we obtain kinematic parameters of the trajectories.

For the muon tomography reconstruction we use the supersaeta model, described in subsection 2.6.3. The model assumes that the incoming and outgoing trajectories have a common point. This point is the vertex of the interaction.

We programmed both algorithms and the vertices are represented in three dimensional histograms (Fig.7.2). The binning of the histograms was chosen to be $1 \times 1 \times 1 \ cm^3$. Therefore, we divide the space in cubes of 1 cm^3 each. These boxes are filled with the reconstructed vertices.

To obtain a cleaner image of the materials we can impose some conditions to the reconstruction of the vertices. Constraints in variables such as position, dispersion angle, or chi-squared can be applied. In the first reconstruction process of the run number 27 we imposed the condition that *a vertex is only reconstructed if the dispersion angle is greater than a value* θ_{cut}. The motivation of this cut is clear: the particles with low deviation have more uncertainty in the reconstruction and do correspond mainly to not-scattered muons.
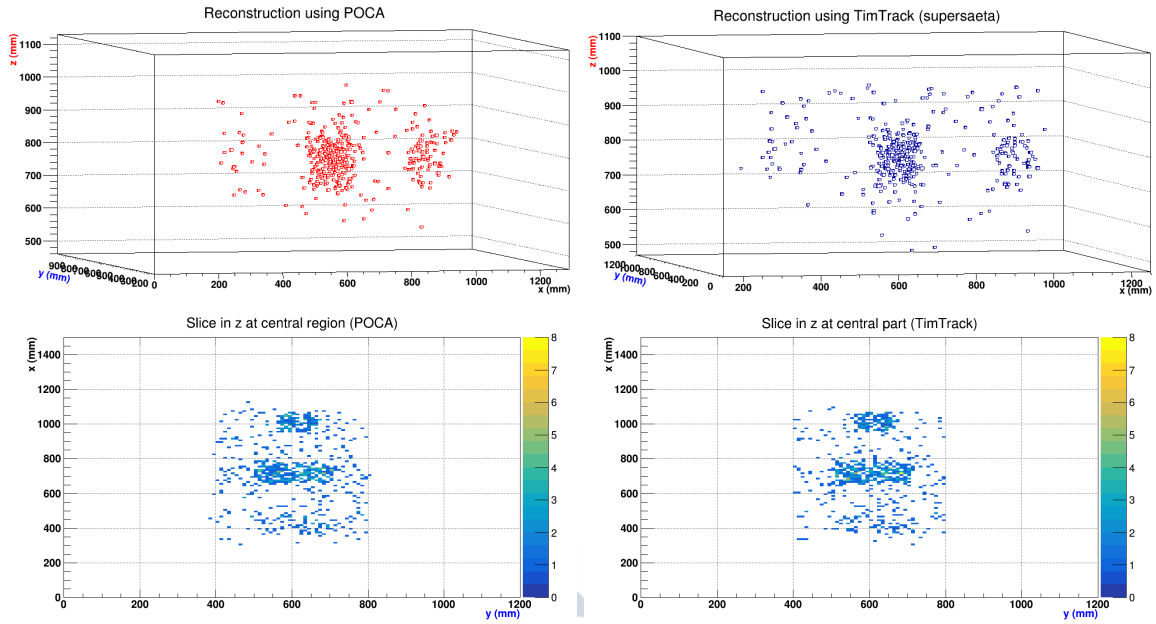
Fig. 7.2 Reconstruction of run 27 using POCA and TimTrack methods. For the TimTrack it was used a supersaeta model. The histograms are filled after a cut in angle $\theta_{ss} > 0.05$ rad. In the top of the figure are shown the reconstructed vertices, and in the bottom a two dimensional projection in the central region.

These particles do not contain relevant information of the Coulomb scattering, and therefore we suppress them.

The deviation angle is calculated using the directions of the incoming and outgoing trajectories:

$$\theta = \arccos\left(\vec{v} \cdot \vec{w}\right) \tag{7.1}$$

where $\vec{v}$ is the direction vector of the trajectory in the upper planes, and $\vec{w}$ is the direction vector in the lower planes.

The cut in dispersion angle was chosen to be:

$$\theta_{ss} > \theta_{cut} = 0.05 \text{ rad} \tag{7.2}$$

where θ_{ss} is the scattering angle obtained by the TimTrack (supersaeta) method.

In Fig.7.3 is shown the distribution of the angular deviation obtained with the TimTrack method. From this figure, we proposed the value of θ_{cut} mentioned before. The events with low deviation angle do not provide much information about the presence of heavy materials, and therefore we exclude them from the analysis. This value of θ_{cut} is the best compromise between cleaning the sample of vertical events and keeping enough statistics to perform a
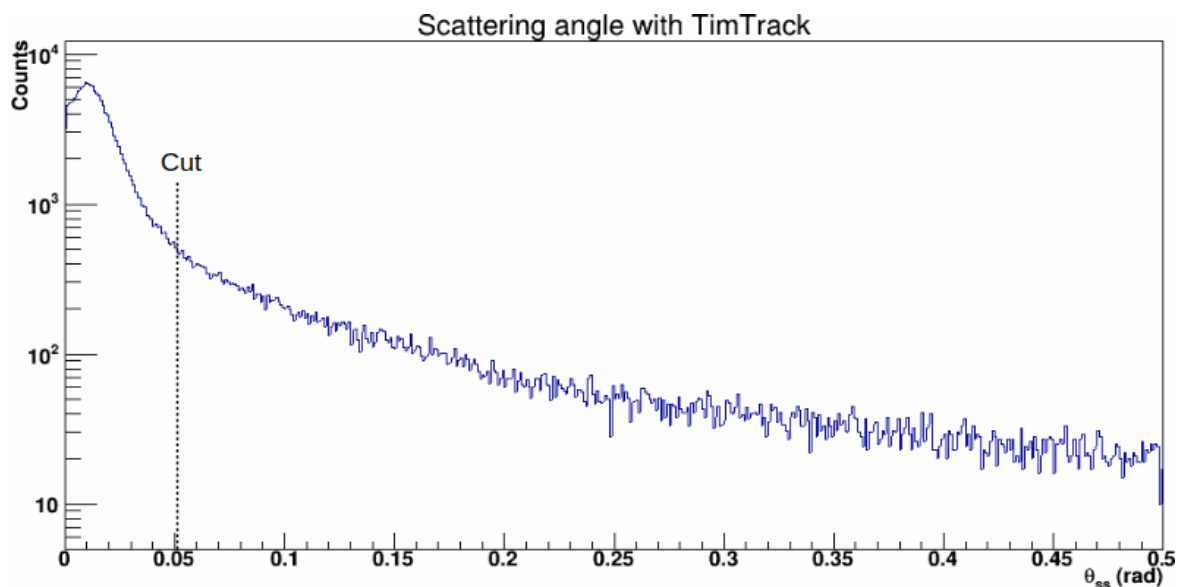
Fig. 7.3 Scattering angle with the TimTrack method. A value of 0.05 rad was chosen as a cut for the reconstructed events.

good vertex reconstruction. As we will see later in the study of the relative polar deviation, this value of θ_{cut} also provides clear structures of the observable.

Figure 7.2 shows the results of the reconstruction methods for run number 27 using POCA and TimTrack (supersaeta) with a θ_{cut} in deviation angle. As one can see, both algorithms provide a good representation of the shapes of the materials. At the bottom part of the figure are shown two dimensional histogram corresponding to the central part of the bricks. These slices of the volume space are 3 mm thick and all the vertices inside the slice are integrated. In these two dimensional histograms can be seen the reconstructed x and y-coordinates of the bricks.

We can conclude that the reconstruction algorithms provide reliable information of the run 27. First, the materials are separated around 20 cm to each other. Second, the position of the centres of the bricks are approximately the same as specified in the run 27 configuration. The exact position of the bricks are not known with that precision. Therefore, it is not possible to compare the reconstructed coordinates with the real positions. However, the information provided by the algorithms seems to be consistent to the real configuration.

As a result, we summarise that the bricks are centred at $x = 400$ mm (Fe), $x = 750$ mm (Pb), and $x = 1000$ mm (W). The dimensions of the bricks are 10×20 cm^2 (Fe and Pb) and 10×10 cm^2 (W).

In the next sections we compare the results of both algorithms. This comparison will provide additional information that will be used to obtain a more clean reconstructed image of the bricks. In addition, some observables could help to the identification of the material.

7.3.1   Comparison of the methods

In this subsection we compare the results of the reconstructed vertices of both methods, POCA and TimTrack.
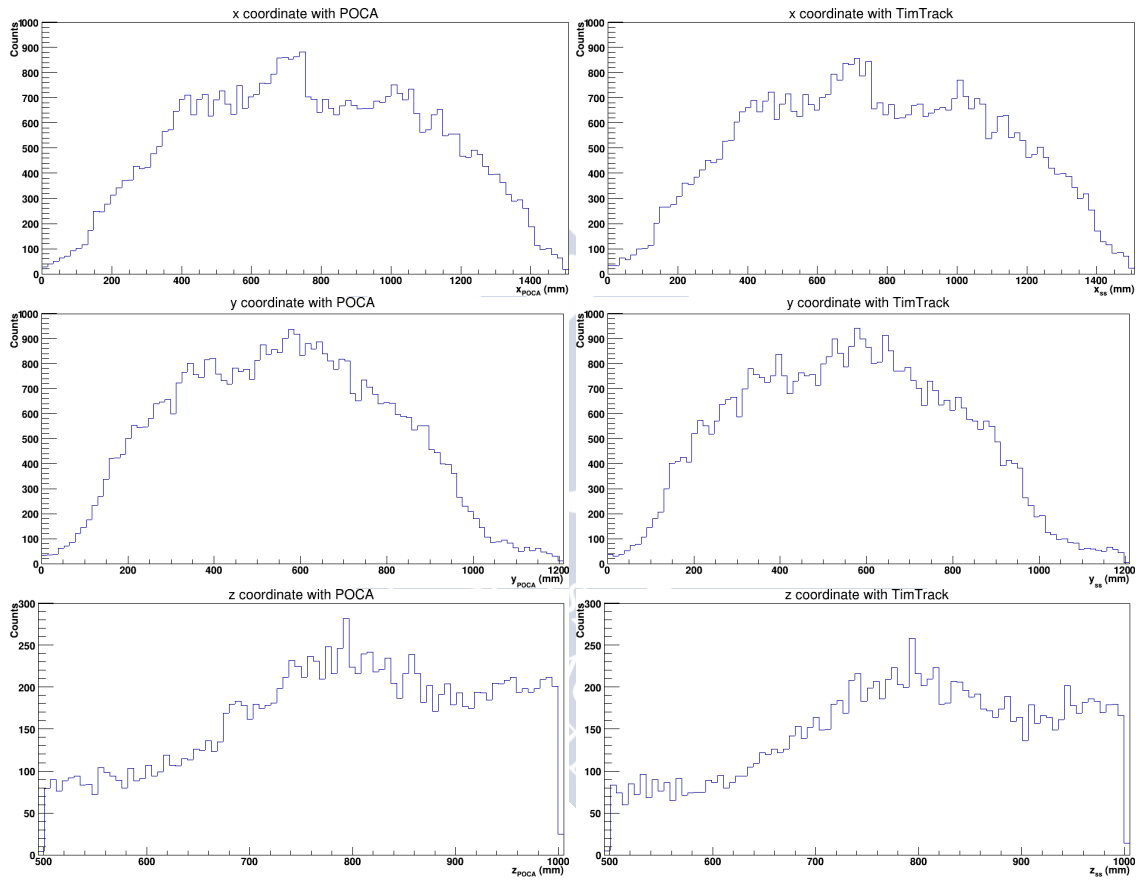

Fig. 7.4 Comparison of the coordinates (x, y, z) of the reconstructed vertices of run 27. The comparison is done between POCA (left) and TimTrack (right) The reconstructed vertices are constricted to a cut in angle $\theta_{ss} > 0.05$ rad.

In Fig.7.4 we show separately the coordinates of the reconstructed vertices with both methods. These values are the individual contributions of the vertices represented in Fig.7.2, and therefore they are already constricted in scattering angle: $\theta_{ss} > 0.05$ rad. In this figure, it is possible to distinguish three different regions in the x-coordinates. These peaks correspond to the bricks. In the middle we can clearly see the lead, and the iron is more diffuse than tungsten.
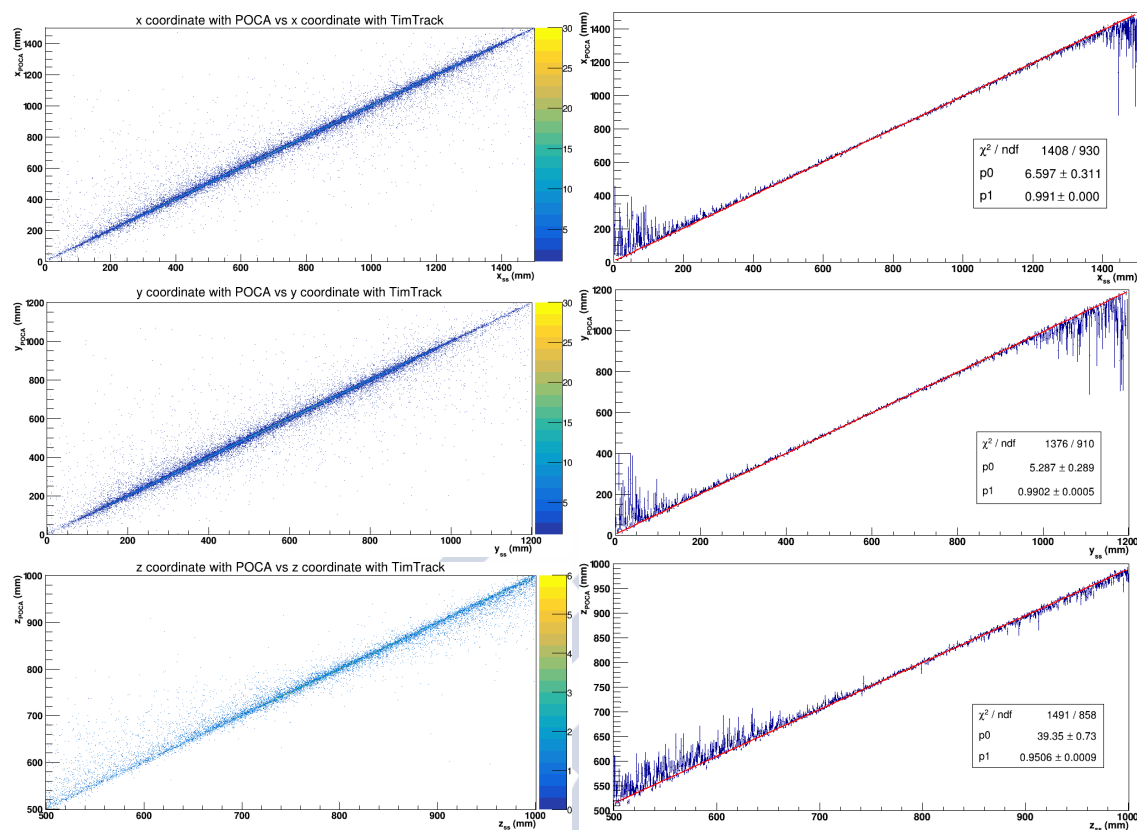
Fig. 7.5 Comparison of the reconstructed variables using POCA and TimTrack. The left side of the figure shows the two dimensional histograms of the POCA versus TimTrack variables. On the right side, a linear fit of the profiles of the histograms.

As it is shown in Fig.7.5 there is a strong linear correlation between the reconstructed variables using POCA and TimTrack. On the left side of the figure are shown the two dimensional histograms that represent the x_{poca} versus x_{ss} (the same for y and z) On the right side we perform a linear fit of the profiles of the histograms. The correlation coefficients are almost 1 for the x and y variables. However, the z component is more sensitive to the reconstruction method. This difference in the z-component is expected since the POCA method is more sensitive to the particles with small deviation angle. The difference in the reconstructed z-coordinate could be related to the scattering properties of the material. Therefore, we study the behaviour of this difference to check if it is related to the materials.

We defined the variable dz as the difference between the reconstructed z-coordinates:

$$dz = z_{\mathrm{poca}} - z_{\mathrm{ss}} \tag{7.3}$$

The representation of this variable is shown in Fig.7.6. As it can be seen, the global distribution of this variable is not symmetric.
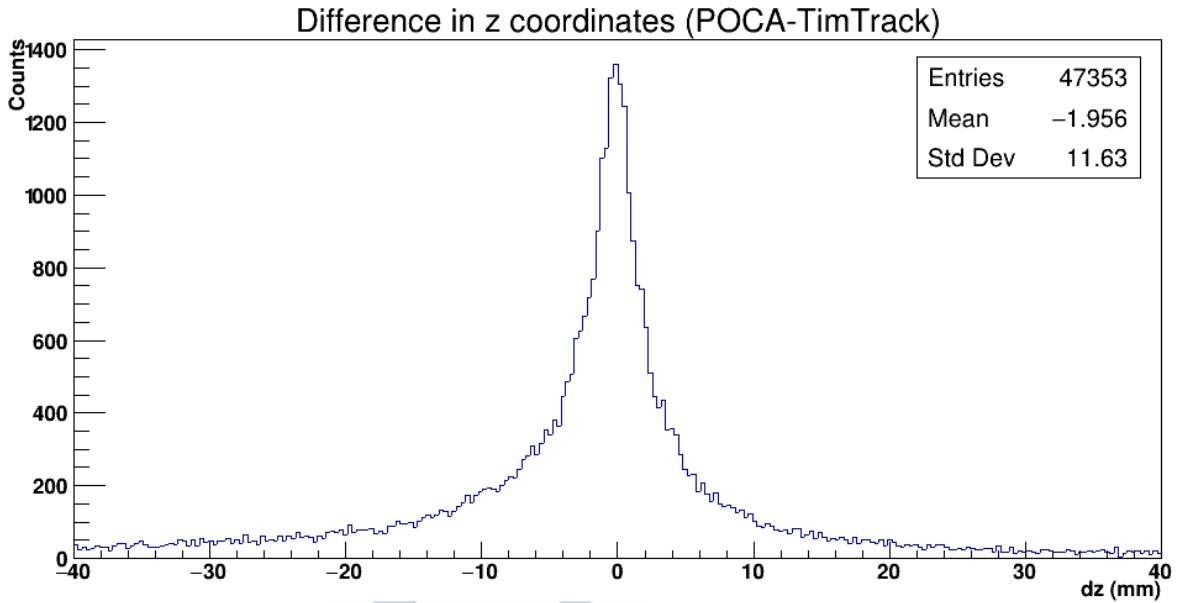
Fig. 7.6 Difference in z-coordinates, dz. The distribution of the variable is not symmetric. As the mean value is negative, typically the TimTrack algorithm reconstructs at higher z-coordinate, i.e., at the lower part of the brick.

As we see, typically the z value reconstructed with TimTrack is greater than the reconstructed with POCA. This means that the TimTrack algorithm trends to reconstruct vertices at the bottom part of the bricks. Since the supersaeta model used in TimTrack enforces a common point for the incoming and outgoing trajectories, it is more probable to find it at the bottom part of the brick.

The results shown in Fig.7.6 seem to be consistent with the simulations performed in Chapter 5. A further study of the difference in reconstructed vertices can be done. In the last column of Fig.7.7 are shown the dz values for each material. The selection of the bricks is done by the reconstructed position shown in Fig.7.2. This means that we do not know exactly the position of the brick and we could be considering vertices that are not inside the material. This fact produces an uncertainty in the individual contributions to the variable. As a consequence, we can only confirm that the results are consistent with the simulations, but the variable by itself does not provide further information about the material. Instead of using only the dz observable, we can consider the whole radial components (next study)

We performed a study to search a proper cut in the variable dz. From the distribution of the variable on air it is possible to use a cut to remove the contribution of the air vertices. We found a value of

$$dz < 10 \text{ mm} \tag{7.4}$$

to perform cuts in the reconstruction algorithm. This value was obtained after the systematic study of the influence of dz in other observables.

In Chapter 5 we proposed some physical observables that could be useful in the muon tomography. The next subsections study some of the observables for the materials of run number 27.

Study of the radial components

The radial vector is defined as the difference of the reconstructed coordinates using both methods:

$$dr = (dx, dy, dz) = (x_{poca} - x_{ss}, y_{poca} - y_{ss}, z_{poca} - z_{ss}) \tag{7.5}$$

As explained in Chapter 5, the lateral dispersion is related to the scattering angle and later to the material.
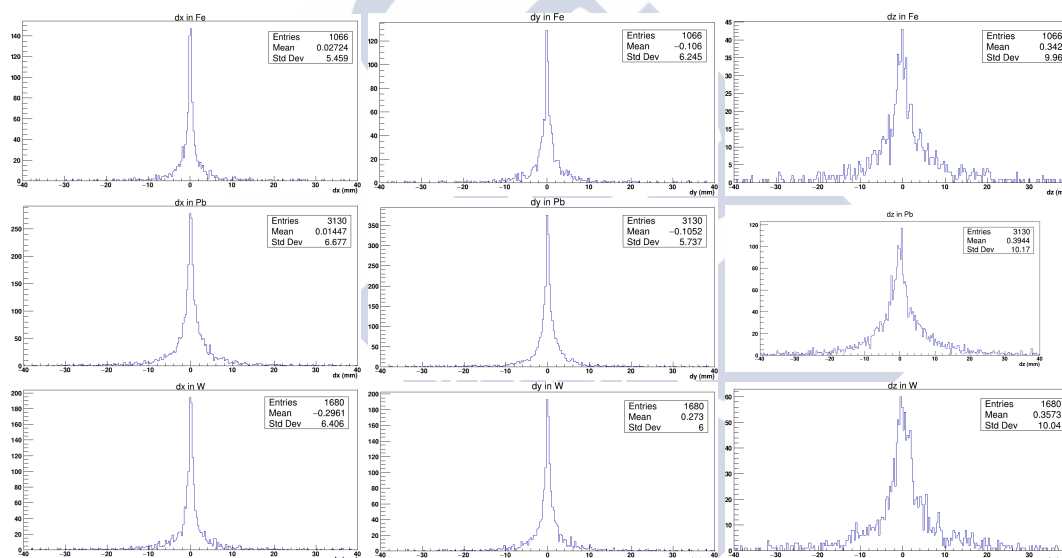
Fig. 7.7 Radial dispersion of the materials in run number 27. The rows are Fe, Pb, and W. The columns show the values of dx, dy, dz.

In Fig.7.7 are shown the values of the radial dispersions. If we compare this figure with the simulated values (as for example shown in Fig.5.8) the behaviour is very similar. As a consequence, the quantity $|dr|$ can be used as a discriminator for the reconstructed vertices. A large module of the vector means that both models reconstruct very separated vertices, and therefore there is an uncertainty in the position, as shown in Fig.5.6.

Study of the scattering properties

We performed the study of the scattering angle of the bricks in run 27. This angle is calculated as in equation (7.1) using the scalar product of the direction vectors. We present the results of this angle obtained with TimTrack (supersaeta) and POCA. The direction vectors in the POCA algorithm are obtained directly from the hits on the RPC planes:

$$\vec{v} = \left(\frac{x_2 - x_1}{z_2 - z_1}, \frac{y_2 - y_1}{z_2 - z_1}, 1 \right) \qquad \text{and} \qquad \vec{w} = \left(\frac{x_4 - x_3}{z_4 - z_3}, \frac{y_4 - y_3}{z_4 - z_3}, 1 \right) \qquad (7.6)$$

where (x_i, y_i, z_i) are the spatial components of the hit at plane i.

By contrast, the direction vectors in the TimTrack method are obtained after the fit of the dynamical parameters through a supersaeta model. For a full description of the equations and parameters see subsection 2.6.3.
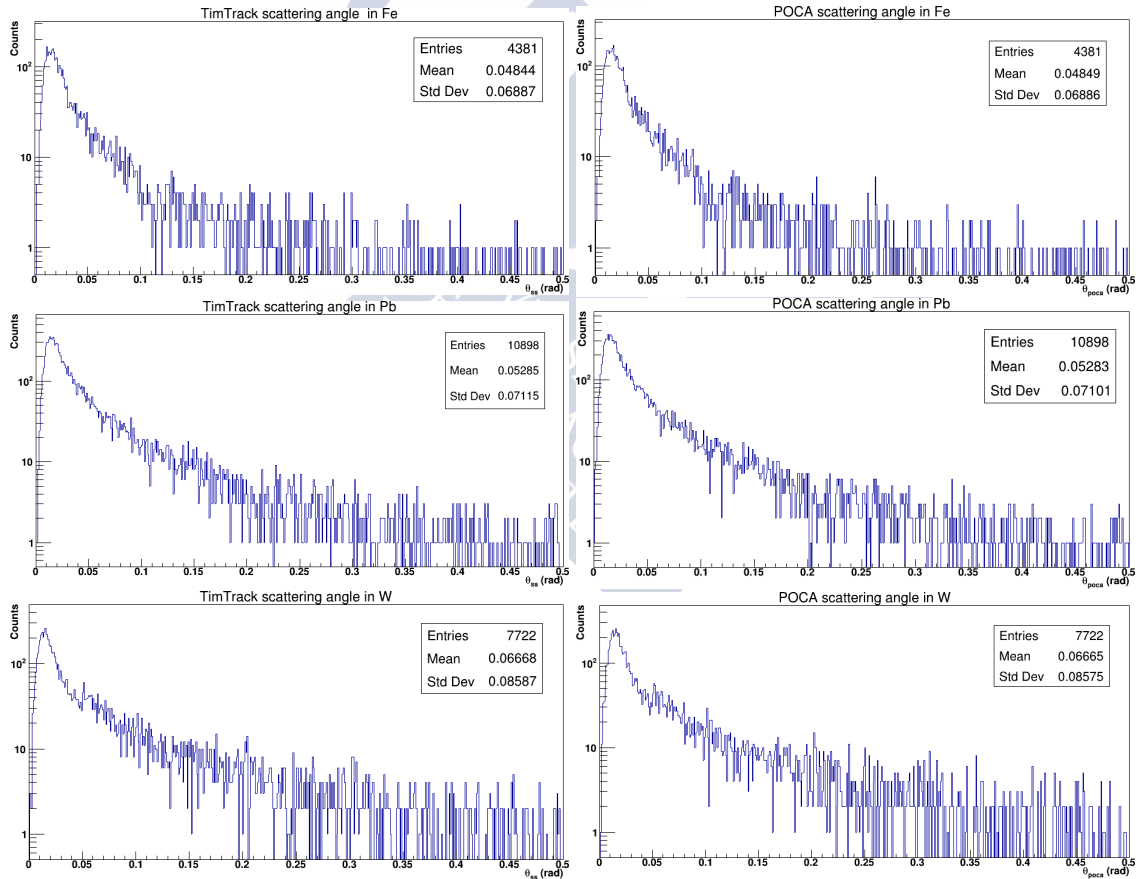
Fig. 7.8 Scattering angle of the materials in the run 27. The left column corresponds to the values with the TimTrack method (supersaeta model) and the right are the angles with POCA.

These angular values are consistent with the simulated structures in Chapter 4. The angle grows with density of the material. As already mentioned in the study of dz, the individual contributions can be affected by the presence of air outside the brick. This fact could change the shape of the distribution. Despite of that, as mentioned, the distributions seem to be consistent with the simulations.

From the angular distributions we also can see that the value $\theta_{cut} > 0.05$ for the angle suppresses the contribution of most of the vertical particles. Therefore, this value is not affecting the important contribution carried by the scattered muons.

Study of the relative polar deviation

In section 5.4.2 we defined the observable that relates the angular difference of both methods with the third component of the radial vector:

$$\frac{d\theta}{dz}.$$

(7.7)

We called this observable **relative polar deviation**. As it was shown in the simulations, this observable was sensitive to the material of the target.
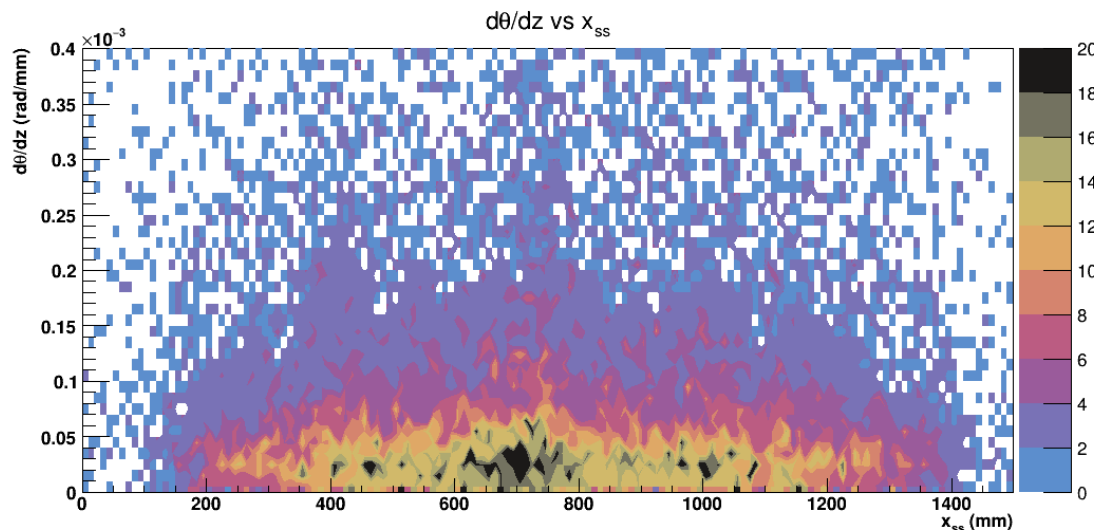
Fig. 7.9 Relative polar deviation as a function of the x_{ss}-coordinate. The observable was evaluated for the scattered particles above the $\theta_{cut} > 0.05$ rad. The three structures in the figure are the different materials, Fe (around $x = 400$ mm), Pb ($x = 700$ mm) and W ($x = 1000$ mm)

In Fig.7.9 we represent the relative polar deviation as a function of the x-coordinate reconstructed with the TimTrack method. We considered only the events above the angular

threshold $\theta_{cut} > 0.05$ rad. In the figure are clearly shown three different structures at around $x = 400$ mm, $x = 700$ mm, and $x = 1000$ mm. These zones correspond to the materials of the run 27. As it was mentioned in Chapter 5 the maximum value of the observable is related to the material. As it can be seen in the figure, the bricks seem to be part of *columns*. The height of this column depends on the material and energy of the incoming particle.

In the figure, lead has the most notable contribution. On one hand, it has a large scattering power. On the other, in this run the size of the brick is also contributing.

Other structure that can be distinguished in the figure corresponds to tungsten, located at around $x = 1000$ mm. The volume of the W brick is 1l, half of the volume of Pb and Fe. Despite of its volume, the structure can be seen more clear than the iron brick.

The iron has the smallest contribution to the relative polar deviation. The scattering properties of iron produce a lack of vertices inside the brick. Iron is therefore more diffuse compared to the others.

We present the evolution of the observable as a function of the cut in angle θ_{cut}. This study can confirm if the value of 0.05 rad provides a good description of the observable. In Fig.7.10 we represented $d\theta/dz$ as a function of x_{ss} for three different values of θ_{cut}: 0.06, 0.07, and 0.1.

The structure around the bricks is more diffuse as the cut in angle grows. Therefore, the choose of $\theta_{cut} = 0.05$ seems to be consistent with a good representation of the observable.

As a summary, this observable is a good candidate to the identification of materials in the muon tomography. In one hand, its behaviour is consistent with the simulations, and therefore it could provide information about the material. On the other hand, we proposed a cut in angle that suppresses the not-scattered muons and at the same time provides a good structure of this observable.

Study of the metric

The metric observable was introduced and defined by [47]. We performed a variation of their definition based on our two reconstruction algorithms, as described in equation (5.12):

$$\tilde{m}_{ij} = \frac{|dr|}{(\theta_i \tilde{p}_i) \cdot (\theta_j \tilde{p}_j)} \tag{7.8}$$

where we use the module of the radial deviation, $|dr|$. In this equation, θ are the angles and $\tilde{p}$ are the normalised momenta.

In Fig.7.11 are shown the different values of the logarithm of the metric observable. The density of vertices in the iron region is distorting the shape of the distribution. Some of the values in this iron region correspond to air. Together with the three materials is shown the
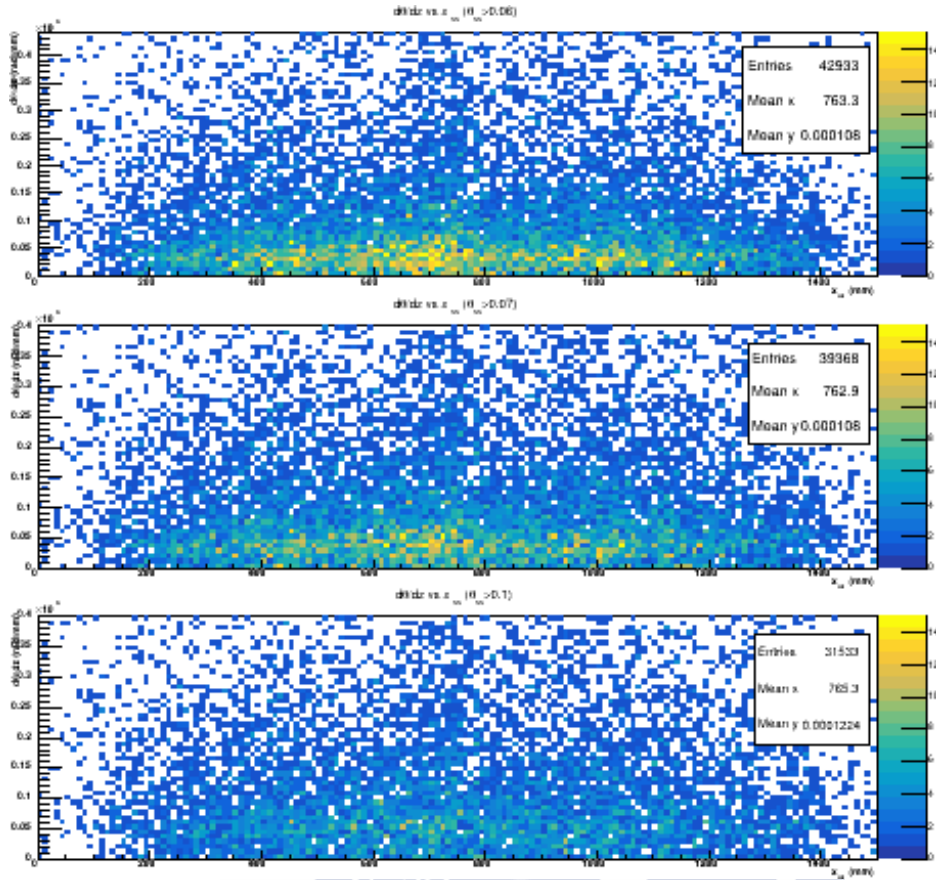
Fig. 7.10 Evolution of the $d\theta/dz$ observable as a function of θ_{cut}. The values of the cut are 0.06, 0.07, and 0.1 rad. The structures around the bricks disappear as the cut grows.

metric of the air. With this value it is possible to perform more cuts in the reconstructed vertices. As it can be seen, the air has the largest value of the metric. Therefore, if we impose a threshold in this observable, the reconstruction of the materials would be more clear since we are excluding the vertices corresponding to the air.

We have chosen the value

$$\ln\left(\tilde{m}_{ij}\right) < 10.0 \tag{7.9}$$

as the cut in metric. If we only consider the vertices with a metric below 10 we are suppressing a great amount of air vertices.

7.4   Reconstruction using the cuts in observables

The observables studied in the previous section can be used to make a better reconstruction of the run 27. The most relevant observables are related to the scattering angle and the metric
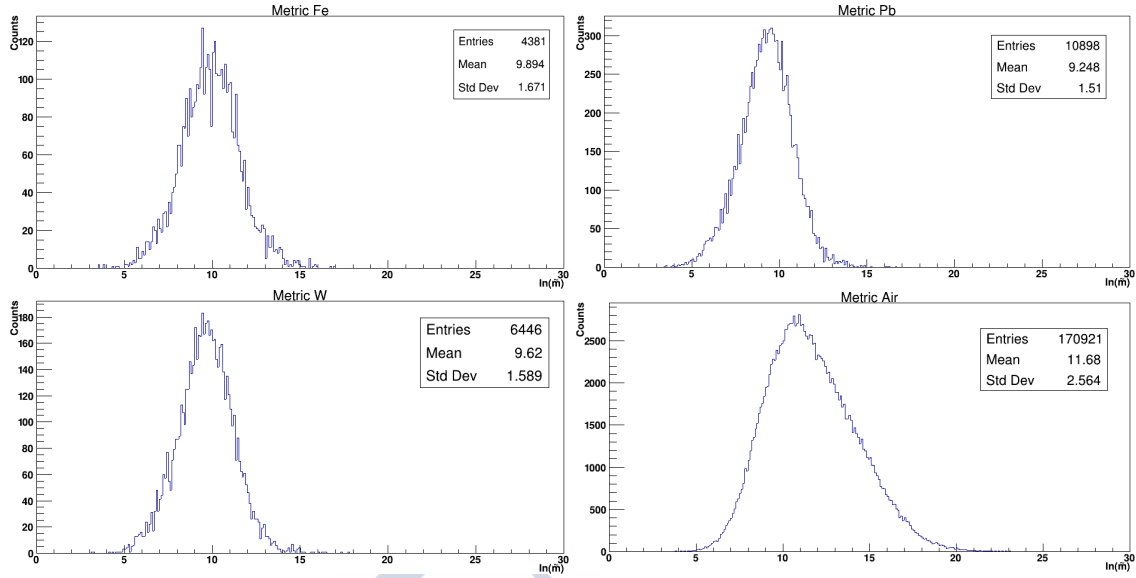
Fig. 7.11 Logarithm of the metric observable for the materials of run 27. The region of iron is more diffuse, and therefore the observable has a different shape.

of the material. In Fig.7.12 we show how the reconstruction using the TimTrack algorithm is evolving as we perform cuts in the variables. These three dimensional histograms correspond to the central part of the detector, i.e., the region between U2 and D1 planes. As it can be seen in the figure, when no cut is applied the vertices produce not a clear image of the bricks. After a first cut in angle we eliminate the muons with less information. A better image is formed. Finally, the image is even better after a cut in metric.

We can set the values of theta and metric to produce the best image as possible. From the studies performed in previous sections, we proposed the values:

$$\theta_{ss} > \theta_{cut} = 0.05 \text{ rad} \qquad dz < 10 \text{ mm} \qquad \ln\left(\tilde{m}_{ij}\right) < 10.0 \qquad (7.10)$$

for the cuts in angle, vertical difference in vertices, and value of the metric. The value of the metric is motivated by the Fig.7.11. We tried not be too much restrictive with the cuts, in particular with the metric. The values of the metric are very similar for all materials. The strategy is to eliminate the contribution of air as much as possible. The value $\ln\left(\tilde{m}_{ij}\right) < 10.0$ removes many vertices in air, but there are still others that can not be removed with this method.

The reconstruction of the run 27 after applying the cuts defined in equation (7.10) can be seen in Fig.7.13. The contribution of the vertices of air is considerably reduced.

The three bricks can be clearly distinguished. The cuts in angle and metric reduce the contribution of the background. The iron brick (left) is not well defined. This fact is due to
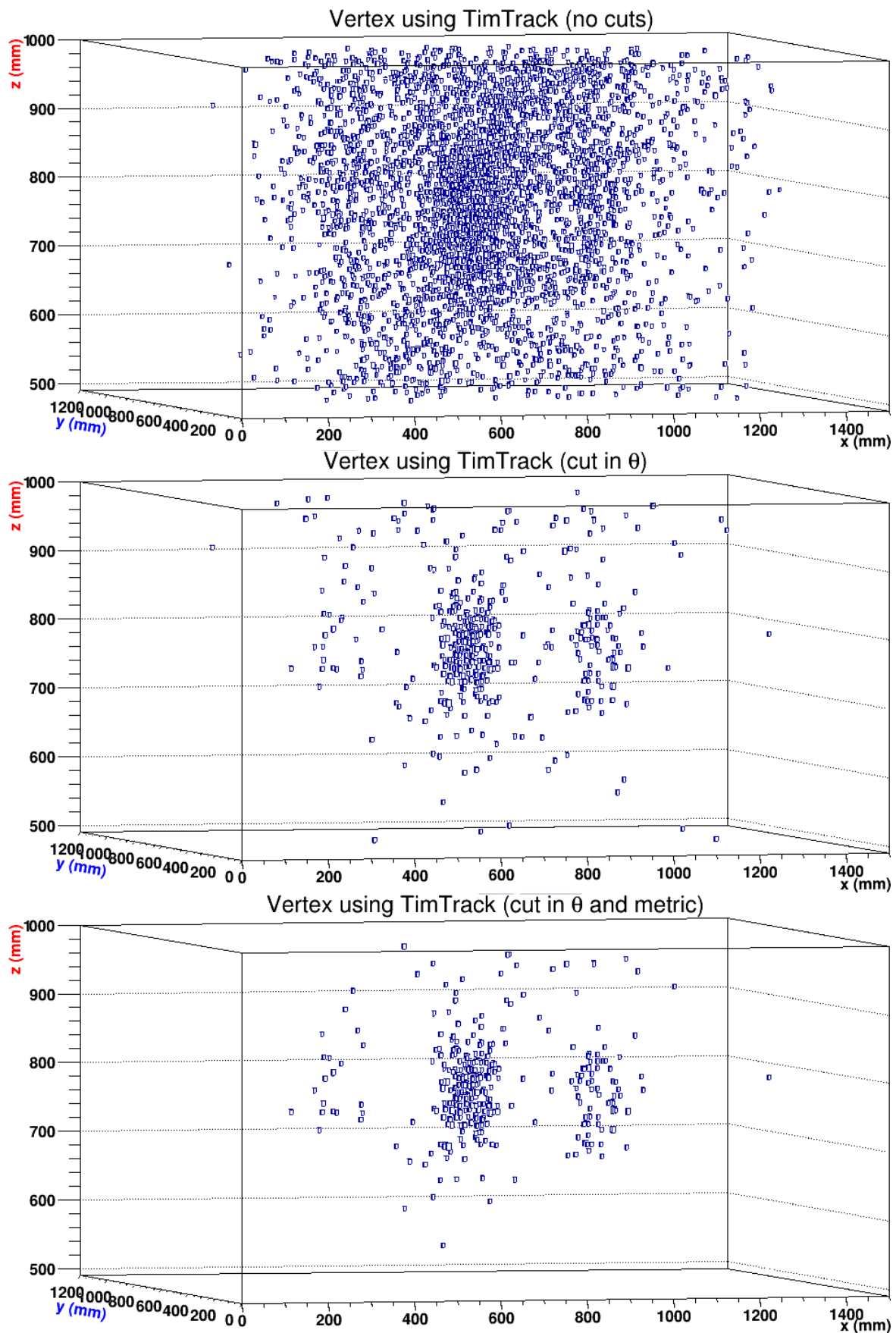
Fig. 7.12 Reconstruction of the run 27 after some cuts. The figures show the evolution of the images when no cut is applied (up), after a cut in theta (middle), and after a cut in theta and metric is applied (low).
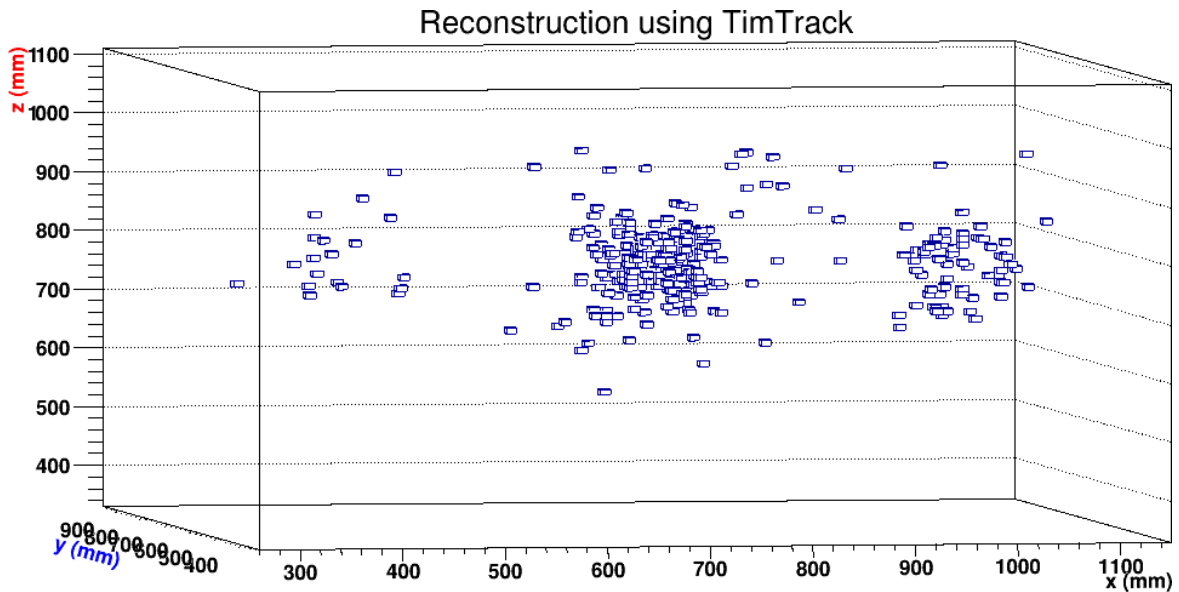
Fig. 7.13 Reconstruction of the run 27 after cuts in angle $\theta_{\mathrm{cut}} > 0.05$ rad, $dz < 10$ mm and in metric $\ln\left(\tilde{m}_{ij}\right) < 10.0$ as defined in equation (7.10) The three bricks can be clearly distinguished without background vertices.

the scattering properties of the material. The vertices in that region were a contribution to the air background and not to the interaction inside the brick.

As expected, the lead (centre) has the best image. The scattering properties and size of the brick provide a good vertex reconstruction without significant contribution to the background.

The tungsten brick (right) can be clearly seen, despite having half of the volume of iron.

In Fig.7.14 we show a two dimensional projection of the central part of the bricks. As it can be seen, the materials have a more clear shape and their dimensions are better defined. We can conclude that the reconstruction algorithms provide a reliable image of the run 27.

The relative polar deviation after the cuts in angle, vertical dispersion, and metric is shown in Fig.7.15. The three materials are clearly seen and the observable presents less background. The maximum values of the relative polar deviation can be determined.
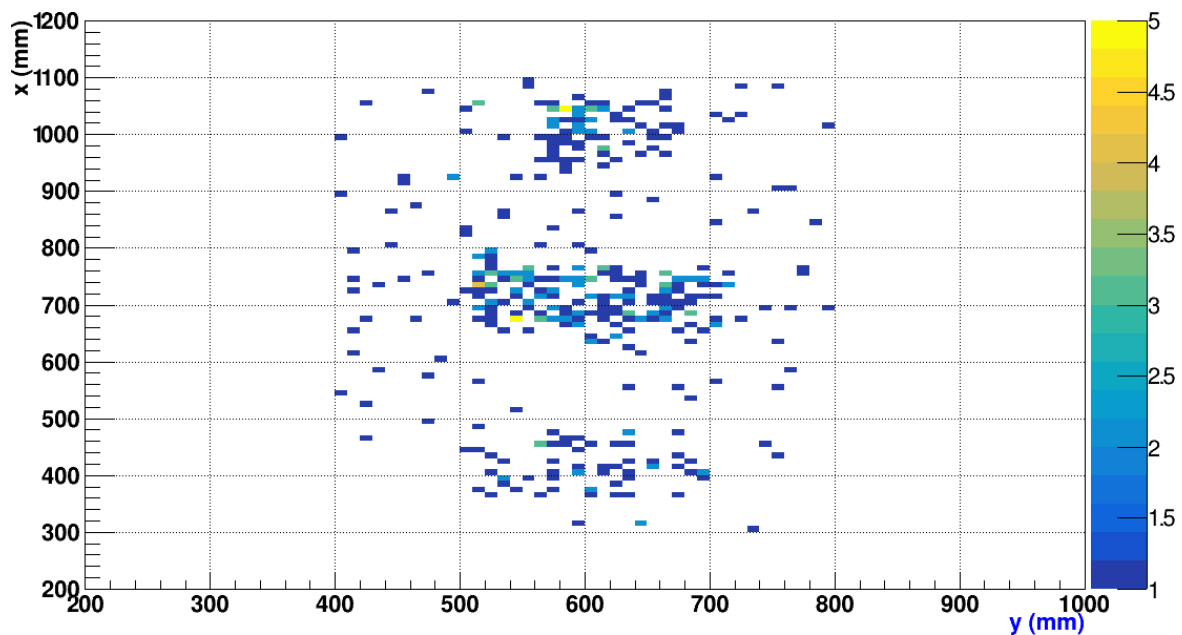
Fig. 7.14 Two dimensional projection of the central part of the bricks. The dimensions of the bricks can be clearly seen.
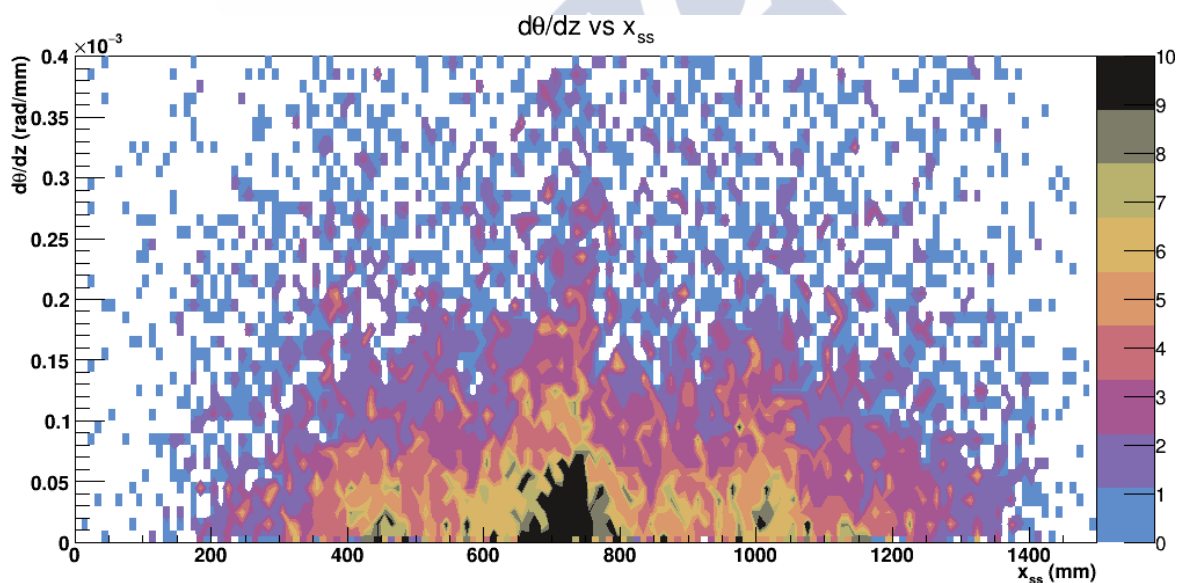

Fig. 7.15 Relative polar deviation, $d\theta/dz$ after the cuts in angle, dz, and metric. The three bricks are clearly seen: Fe (left), Pb (centre), and W (right).

In Table 7.2 is shown a summary of the reconstruction properties of the materials. In this table we present the centre of the bricks on the x-axis, the dimension of the bricks, and the value of the relative polar deviation. The dimensions are obtained from Fig.7.14 and the relative polar deviation from Fig.7.15.

Material	Centre x (mm)	l_x (mm)	l_y (mm)	$d\theta/dz \times 10^{-3}$ (rad/mm)
Fe	400	10	20	0.05
Pb	700	10	20	0.15
W	1000	10	10	0.15

Table 7.2 Summary of the reconstruction process. We present the centre at the *x*-axis, the values of the lengths along the *x* and *y* axis, obtained from Fig.7.14. The values of the relative polar deviation are obtained from Fig.7.15

These values can be compared to the obtained in Chapter 5. One of the problems of the method consists on the size of the bricks. As we can see in Table 7.2 the lead and tungsten present the same relative polar deviation. This is due to the size of the brick: we have 2l of lead and 1l of tungsten. The scattering properties of both bricks are similar, according to this observable.
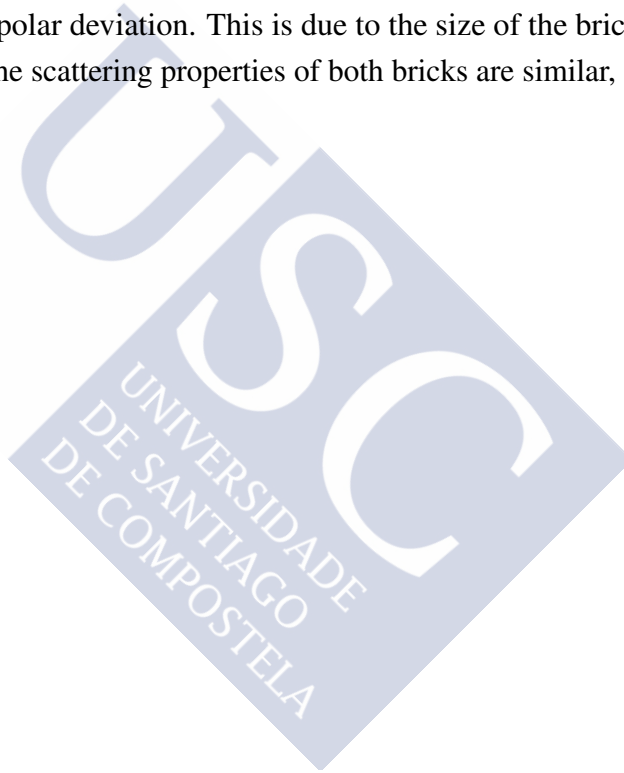
Chapter 8

Summary and conclusions

In this thesis I presented the development of the algorithms and software tools needed for the construction of a muon tomography system. This scanner should be able to analyse containers directly on the docks, in a short period of time, and in an efficient manner. The main task of the system is to detect the presence of radioactive or heavy Z materials inside the container.

The muon tomography system is based on the RPC detector technology. These gaseous detectors are widely used in particle physics and their features make them suitable for cosmic muon tomography purposes.

At the Particle Physics group of the Universidad de Santiago de Compostela (USC, Spain) we have a detector already devoted to the study of the cosmic radiation: the TRAGALDABAS detector, which is also based on the RPC technology and several simulation and analysis tools were already designed and tested in it.

A new model of RPC assembly, the *muTT* detector, is the technology of the muon tomography system used in this work. It was designed and constructed in LIP-Coimbra (Portugal) and it was transported to an industrial hangar in O Porriño (Pontevedra, Spain) where it was tested and finally added to the prototype of scanner.

The first block of the thesis was devoted to the implementation of the software environment. Inside this structure we can perform simulations of cosmic radiation and also study the features of our detectors and experimental set-ups.

We used the EnsarRoot framework as the standard software tool of the analysis and simulation tasks. The geometry and the basic data structure of TRAGALDABAS and *muTT* were included in EnsarRoot. I have developed also the analysis software tools which can be used either in TRAGALDABAS or in *muTT* systems.

One of the advantages of EnsarRoot is the possibility of reuse already made code. Most of the structure developed for TRAGALDABAS were easily re-adapted for the *muTT* detector.

As an example, I created an interface that allows EnsarRoot to use the CRY cosmic ray generator as an input for simulations.

I also proposed a cascade identification algorithm that can be used in the detectors of the same family as *muTT* or TRAGALDABAS. With this algorithm we can structure the information of events with high multiplicity.

The TimTrack algorithm is the standard particle reconstruction method for the *muTT* detector. I presented a description of the algorithm and some models that permit to calibrate the *muTT* detector and to perform the reconstruction of vertices in the muon tomography system.

Beside the simulation of the detectors, I simulated the behaviour of the cosmic radiation on materials. The Coulomb multiple scattering is the key process in understanding the muon tomography. The cosmic radiation at sea level is mainly composed by muons and electrons. The muons provide the most valuable information about the Coulomb multiple scattering. However, the electrons can be used as a first indicator of the presence of a heavy material. It could be possible, in a future implementation of the design, to study the contribution of the electromagnetic part of the cosmic radiation. This study could be used to decide whether the container is clean or needs a more detailed analysis.

Several physical observables were simulated within the EnsarRoot framework. I performed systematic studies of Monte Carlo simulations of quantities related to the scattering properties of materials. I proposed observables based on the scattering angle θ and their behaviour was studied for several energies of the cosmic muons over different materials.

For the study of the scattering properties of muons,the TimTrack method was used to perform the analysis of the muon trajectories. The simulation of the behaviour of physical observables related to the scattering properties of the materials was done.

Together with the TimTrack method we used the POCA (Point Of Closest Approach) algorithm as a reconstruction tool. Despite POCA only uses geometrical information, avoiding correlation between kinematic parameters, it is an optimal reconstruction tool. We proposed other physical observables based on the difference between POCA and TimTrack. These quantities can be used to obtain information about the test material.

The TimTrack method was used for the first time to perform calibrations of the *muTT* detector. Using this algorithm, there were found several systematic effects in the internal structure of the *muTT* detector. Several time and spatial correction were done after the analysis of the trajectories with the TimTrack method. As a result, after the application of TimTrack, we obtained better resolutions compared to the provided by the manufacturer. After reconstruction with TimTrack we obtain:

Plane	σ_x (mm)	σ_y (mm)	σ_t (ps)
1	1.28(1)	1.13(2)	98.7(2)
2	1.67(3)	1.43(3)	134.6(3)
3	1.93(4)	1.50(3)	120.4(3)
4	1.47(3)	1.18(2)	88.7(1)

These values are better compared to the original specifications.

After the calibration process I performed the reconstruction of some test materials. The reference run used in this work consisted on two pieces of $10 \times 20 \times 10$ cm^3, one of Fe and one of Pb, and a third piece of $10 \times 10 \times 10$ cm^3 made of W. The three bricks were placed in the scanning region of the *muTT* system. The result of the reconstruction can be seen in Fig.8.1

Both POCA and TimTrack methods provide good images of the bricks. Some new observables were proposed based on the difference in the reconstruction with both methods. The three more relevant are the difference in the third component of the reconstruction vertex, *dz*, the relative polar deviation $d\theta/dz$, and the metric $\tilde{m}$. These three observables are used to perform studies about the density of the materials.

As a conclusion, these observables were found to be good candidates to determine the density of the test material. Future studies can explore their use and maybe make possible a better material identification.

One of the possible research lines for future works could be the improvement in the reconstruction method using more time corrections. In particular, there is a parabolic dependency between the time-of-flight and the scattering angle. The form of the parable is different from one material to other.

Other possible study is related to the corrections in the position of the hit (x, y). The signal induced by a particle inside the detector has a dependency on the incident angle of the particle. This parallax effect between the entry point of the particle and the point where the signal was induced can be studied and better values of position can be obtained.

Improvements in the software analysis can be done by the implementation of pattern recognition algorithms. These tools could be used to obtain better images in a shorter period of time and to separate between *"clean"* and *"deeper analysis"* containers.

Since the shapes of the target materials are unknown, it could be possible to apply Principal Component Analysis (PCA) techniques to the reconstructed data. This analysis can provide information about the dispersion along the individual variables.
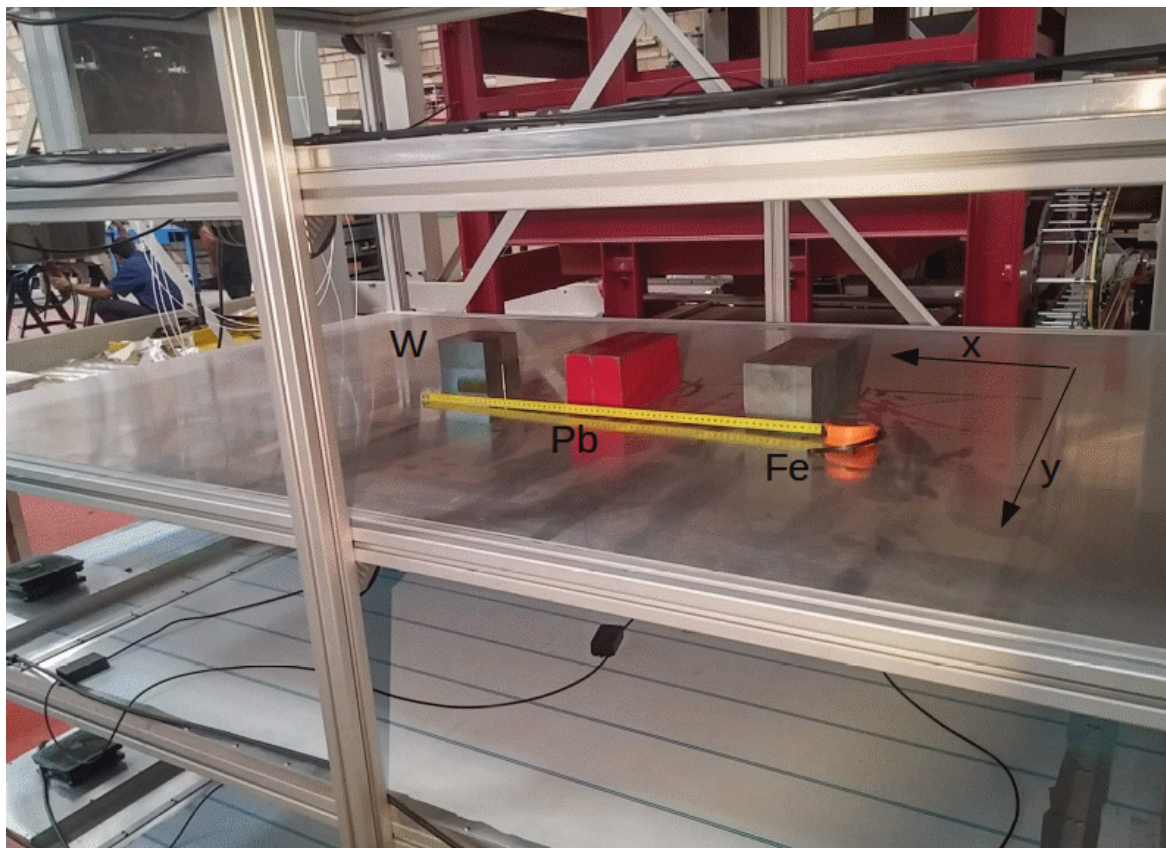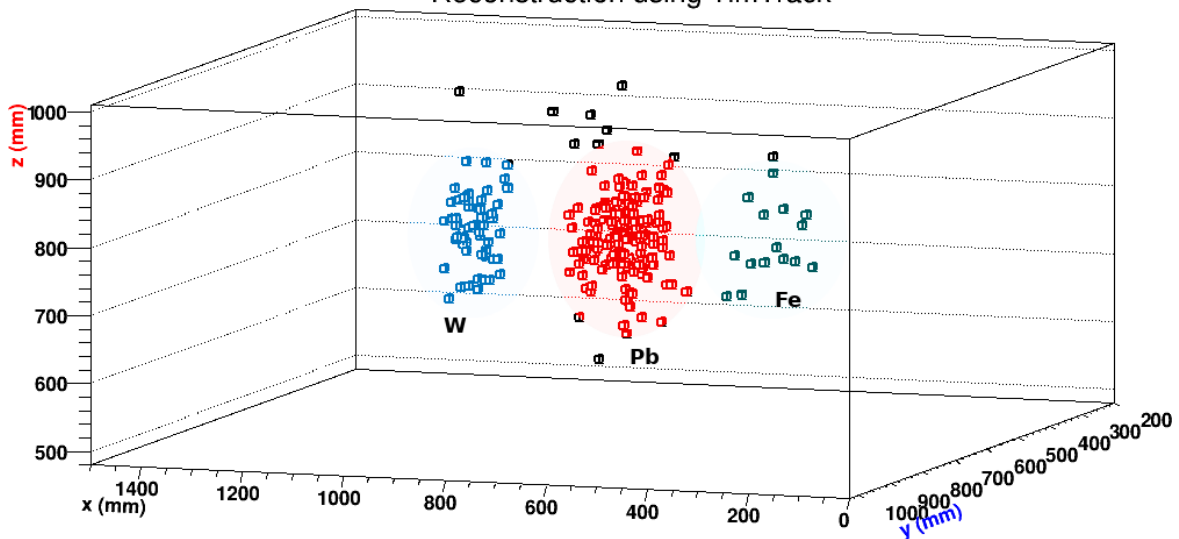
Fig. 8.1 Reconstruction of the run 27 using the TimTrack algorithm. There are three bricks made of Fe, Pb, and W that can be clearly seen.

The analysis of the data measured with the macro scanner must to be done. The prototype structure of the muon tomography system is placed at DIGAFER industrial hangar in O Porriño, Pontevedra. A picture of the scanner is seen in Fig.8.2.

Fig. 8.2 Prototype of the macro scanner structure. Two *muTT* detectors are placed above the container, and the other two are placed below. The four detectors move at the same time to cover the whole the surface of the container.

The calibration and analysis of data with this structure could provide information about the exposure time and the possible background contribution of the container and structure to the material identification.

Further studies of the structure can be done. The system with detectors placed up and down provides maximal resolution and acceptance if the targets are placed in the centre of the container. If the materials are placed outside this region, and specially on the corners, this set-up is not optimal. The possibility of placing detectors vertically at the lateral part of the scanner should be studied. Whit this new set-ups the lateral muons can also be detected, and further improvements in angular acceptance are possible.

Chapter 9

Resumen de la tesis en español

Introducción

En esta tesis hemos presentado diferentes algoritmos que serán de ayuda en la construcción y posterior desarrollo de un tomógrafo de muones.

La mejora en la seguridad de las fronteras para evitar el tráfico de sustancias radiactivas se ha convertido en una prioridad para los países de la Unión Europea. Se cree que gran parte del tráfico de estas sustancias tiene lugar mediante el uso de barcos mercantes. Estos barcos son capaces de transportar gran cantidad de contenedores siendo posible el paso de materiales que puedan comprometer a la seguridad nacional.

Actualmente, muchas líneas de investigación están orientadas al desarrollo de mecanismos de detección en las propias instalaciones portuarias. Puesto que cada día miles de contenedores deben ser descargados y despachados, es primordial que los dispositivos sean capaces de determinar la presencia de materiales radioactivos de la manera más rápida y eficaz posible. Cualquier pequeño retraso en la tramitación de mercancías puede llevar a un colapso en el tráfico del puerto.

Una de las maneras más evidentes y directas de realizar esta detección sería mediante el uso de rayos X. De esta manera, se procede a realizar una radiografía del contenedor y así se vería si este contiene o no materiales pesados. A pesar de que la radiación electromagnética de alta energía resulta ideal para esta misión por tener gran capacidad de dispersión ante materiales pesados, existen varios inconvenientes que hacen muy difícil su uso desde el punto de vista técnico. Por una parte existe la limitación dada por la superfice del contenedor. Sería muy costoso (tanto desde el punto de vista económico como desde el punto de vista físico) cubrir con fuentes de rayos X todo ese volumen manteniendo gran intensidad de radiación. Otro inconveniente tiene que ver con la naturuleza de la radiación. No es descartable el

tráfico de seres humanos u otros animales, por lo que los rayos X pueden ser perjudiciales para los seres vivos en el interior del contenedor.

Una alternativa es el uso de la radiación cósmica natural, principalmente compuesta por muones, electrones y fotones. Esta radiación constituye una fuente natural y permanente que además no resulta dañina para los seres vivos. Una de las limitaciones de usar radiación cósmica es la baja intensidad de partículas en comparación con las fuentes artificiales, como los rayos X o pequeños aceleradores de partículas.

Para poder analizar de manera eficiente y en corto periodo de tiempo el interior de un contenedor es necesario el uso de detectores y técnicas de reconstrucción e identificación optimizadas.

Rayos cósmicos

Los rayos cósmicos son una radiación natural de origenes variados (el Sol, nuestra galaxia y otras galaxias) que incide permanentemente sobre nuestra atmósfera. Está constituida por protones y otros núcleos atómicos llegando a alcanzar energías muy superiores a las que se consiguen en los mejores laboratorios de investigación en la Tierra. Dicha radiación colisiona a gran altitud con otros núcleos atómicos de la atmósfera produciendo grandes cantidades de los llamados rayos cósmicos secundarios, que son los que llegan hasta la superficie terrestre. Éstos, llegan a un ritmo de unos 180 por metro cuadrado y por segundo, siendo en su mayoría muones, una partícula inestable muy penetrante y electrones y fotones de alta energía, o rayos gamma. Estos llegan a alcanzar energías una 50 veces mayores que las de los rayos X mas energéticos.

Los rayos cósmicos mas abundantes y susceptibles de ser usados para la radiografía de grandes volúmenes son los muones; y casi todos ellos llegan próximos a la dirección vertical.

En el caso de los electrones y los rayos gamma, su distribución es mas uniforme, aunque los efectos que sufren dentro de un contenedor son muy diferentes a los sufridos por los muones y su posible uso requerirá otro tipo de estrategia en los algoritmos. Hay que tener en cuenta que casi un 30% de los muones que inciden tienen una energía inferior a 1 GeV y un 50% de ellos tienen una energía inferior a 2 GeV. Todos estos muones de menor energía serán los mas importantes para su uso en la radiografía de muones. Es conveniente, por lo tanto, optimizar nuestras simulaciones con rayos cósmicos de la forma más realista posible.

El algortimo de reconstrucción TimTrack

Uno de los aspectos más importantes en el análisis de datos experimentales es la reconstrucción de los eventos. El algoritmo TimTrack [31] está basado en el método de mínimos cuadrados [18] y es la herramienta de reconstrucción de referencia en los detectores de la familia *Trasgo*.

El objetivo de TimTrack es la determinación de los parámetros dinámicos de la partícula a partir de las medidas experimentales obtenidas de los detectores. Estos parámetros se agrupan en un vector llamado *saeta*:

$$\mathbf{s} = \left(X_0, X', Y_0, Y', T_0, S \right) \tag{9.1}$$

donde X_0, Y_0 son las coordenadas del inicio de la propapagión de la partícula, T_0 es el tiempo inicial respecto a un contador externo, X', Y' son las pendientes de la trayectoria según los ejes x e y, S es la llamada lentitud (inversa de la velocidad).

Una de las ventajas del método es que TimTrack es capaz de unificar todos los datos experimentales independientemente del tipo de detector que los mide. Si en un mismo experimento tenemos, por ejemplo, detectores RPC junto con plásticos centelleadores, TimTrack es capaz de manejarlos de manera conjunta.

La conexión entre el espacio de medidas experimentales y el espacio de los parámetros se realiza a través de un modelo $\mathbf{m}(\mathbf{s})$. Este modelo consta de tantas ecuaciones como datos experimentales n_d. Dependiendo del tipo de detector y del problema a tratar tendremos una forma del modelo u otra.

Los parámetros son obtenidos tras resolver la llamada ecuación *sea*:

$$\mathbf{s} = K^{-1} \cdot \mathbf{a} \tag{9.2}$$

donde en esta ecuación tenemos la inversa de la matriz K (llamada *matriz de configuración*) y el vector de datos reducido $\mathbf{a}$. Estos objetos se definen como:

$$
\begin{aligned}
K &= G' \cdot W \cdot G \\
\mathbf{a} &= G' \cdot W \cdot (\mathbf{d} - \mathbf{g_0}) \\
s_0 &= (\mathbf{d} - \mathbf{g}_0)' \cdot W \cdot (\mathbf{d} - \mathbf{g}_0)
\end{aligned}
$$

donde usamos la matriz jacobiana

$$G = \frac{\partial \mathbf{m(s)}}{\partial \mathbf{s}} = \begin{pmatrix} \frac{\partial m_1(s)}{\partial s_1} & \frac{\partial m_1(s)}{\partial s_2} & \cdots & \frac{\partial m_1(s)}{\partial s_{n_s}} \\ \frac{\partial m_2(s)}{\partial s_1} & \frac{\partial m_2(s)}{\partial s_2} & \cdots & \frac{\partial m_2(s)}{\partial s_{n_s}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial m_{n_d}(s)}{\partial s_1} & \frac{\partial m_{n_d}(s)}{\partial s_2} & \cdots & \frac{\partial m_{n_d}(s)}{\partial s_{n_s}} \end{pmatrix}$$

la matriz de pesos estadísticos W (en nuestro caso es siempre diagonal) y el término libre

$$\mathbf{g}_0(s) = \mathbf{m(s)} - G \cdot \mathbf{s}$$

Puesto que TimTrack obtiene los parámetros a través de un ajuste estadístico que tiene en cuenta las incertidumbre de las medidas, tenemos acceso a las correclaciones entre parámetros. Esto se hace a través de la inversa de la matriz de configuración. Es por ello que a esta matriz se la conozca como *matriz de error*.

En caso de que nuestro experimento tenga de manera conjunta varios detectores, los parámetros se obtienen resolviendo la ecuación *sea* general:

$$\mathbf{s} = \left(\sum_i^n K_i \right)^{-1} \cdot \left(\sum_i^n \mathbf{a}_i \right)$$

en el que cada detector es explicado por un modelo y por lo tanto tiene asociada una matriz K y un vector $\mathbf{a}$.

Dependiendo del tipo de modelo que tengamos, podemos distinguir entre TimTrack *lineal* o *no lineal*. El TimTrack de tipo no lineal se resuelve de manera iterativa. El algoritmo de resolución es el mismo que el de cualquier problema de análisis de varias variables: proponemos una solución inicial y la vamos modificando hasta encontrar el mínimo:

$$\mathbf{s}_i = K_{i-1}^{-1} \cdot \mathbf{a}_{i-1}$$

El proceso iterativo para cuando un criterio de convergencia es alcanzado. En nuestro caso, proponemos criterios basados en el módulo euclídeo de las saetas. Si la diferencia entre la nueva saeta y la usada en el anterior paso son muy poco diferentes en módulo, entonces paramos y decimos que el proceso ha convergido.

TimTrack puede ser usado tanto como herramienta de calibración como algoritmo de reconstrucción de vértices de interacción coulombiana. En esta tesis hemos creado algoritmos de calibración del detector *muTT* basados en TimTrack.

La reconstrucción de vértices ha sido realizada a través de un modelo llamado *supersaeta*. Este modelo está fundamentado en la idea de punto común entre trayectorias: establecemos las ecuaciones de la trayectoria incidente y saliente. La partícula cambia de trayectoria debido a la presencia de un medio material. El modelo encuentra las coordenadas del punto común por ambas trayectorias incidente y saliente. Ese punto es el vértice de interacción.

Resistive Plate Chambers (RPCs)

Las RPCs son detectores de tipo gaseoso ampliamente usados en los experimentos de física de partículas. Inicialmente han sido usados como contadores de partículas a la entrada de experimentos como ATLAS. Sin embargo, la mejora en sus diseños permiten que sean candidatos para tareas de reconstrucción de trayectorias. Actualmente se pueden obtener resoluciones temporales por debajo de los 200 ps, lo cual permite una buena identificación de trayectorias individuales.

En esta tesis hemos trabajado con detectores de la familia *Trasgo*. Estos detectores son del tipo RPC con múltiples volumenes activos y que están enfocados al estudio de rayos cósmicos. Por una parte, tenemos el detector TRAGALDABAS [4], que está en el departamento de física de partículas en la Universidad de Santiago de Compostela. Este detector trata de estudiar propiedades atmosféricas relacionadas con el flujo de rayos cósmicos. TRAGALDABAS ha sido construido tomando como base la electronica del experimento HADES (GSI, Alemania). Por otra parte, tenemos el detector *muTT*. Este nuevo modelo de RPC ha sido diseñado y construido en LIP-Coimbra (Portugal) y ha sido definido como detector de referencia del sistema de tomografía de muones. Ambos detectores pueden observarse en la Fig.9.1.

Así pues, disponemos de dos detectores del tipo Trasgo con los que hemos trabajado en paralelo. Por una parte, TRAGALDABAS tiene la zona de detección dividida en celdas rectangulares. La señal inducida por la partícula es procesada por la electrónica en función de la carga recogida en cada celda. Por otra parte, *muTT* está basado en tiras de hilos. Cada plano tiene dos planos de hilos, colocados perpendicularmente entre sí. Cada tira está formada por una colección de 25 hilos que miden la carga depositada. El juego superior de tiras mide, además de la carga depositada a derecha e izquierda de cada tira, el tiempo de vuelo de la señal. Tanto TRAGALDABAS como *muTT* proporcionan una medida (x, y, t), es decir, dan valores de posición de impacto y tiempo.

El desarrollo de las herramientas de simulación y reconstrucción han sido desarrolladas en paralelo para ambos detectores. Muchas de las propiedades (como por ejemplo la geometría) de TRAGALDABAS han sido simuladas dentro de la estructura de EnsarRoot [26]. Por
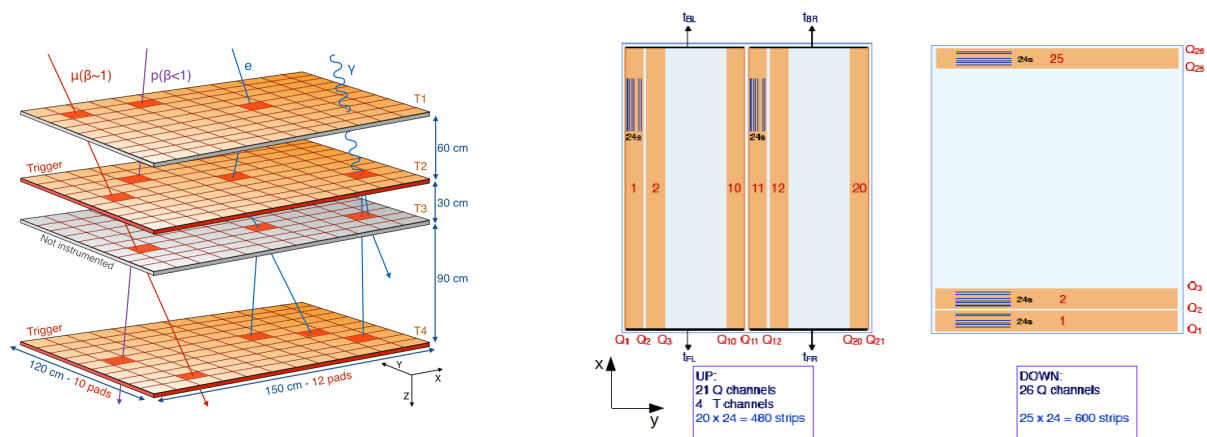
Fig. 9.1 Diferencia entre TRAGALDABAS (derecha) y *muTT*(izquierda). TRAGALDABAS está basado en zonas de detección rectangulares, mientras que *muTT* en tiras formadas por hilos.

lo tanto, muchas de las librerías y demás objetos de control de las estructuras de datos de TRAGALDABAS han sido readaptadas para ser usadas con el *muTT*.

El detector *muTT* fue trasladado hasta una nave industrial en O Porriño (Pontevedra) para ser probada antes de su instalación final en la estructura del tomógrafo. Esta disposición reducida permitió realizar estudios de calidad del detector. Defectos en la electrónica fueron corregidos y diferentes calibraciones (aliniamiento y desfase temporal) fueron aplicadas.

Fig. 9.2 Estructura reducida del *muTT*. En esta configuración se corrigieron defectos en la electrónica y se realizaron calibraciones en el detector previa instalación en la estructura final del tomógrafo.

La nomenclatura usada en el *muTT* identifica cada plano a través de una letra y un número. Por una parte, se diferencia entre *U* o *D* para decir si el plano es *up* o bien *down* (si está arriba o abajo tomando el centro como referencia). Por otra parte, el número indica el orden

de colocación del plano respecto al límite superior. Por lo tanto, disponemos de los planos U1, U2, D1 y D2.

Además de calibraciones y otros ajustes, esta estructura sirvió como banco de pruebas de nuestros algoritmos de reconstrucción de materiales.

Estructura general de las simulaciones

En todo proceso de construcción de estructuras y propuesta de observables físicos es imprescindible realizar una simulación previa de los mismos. Los experimentos de física nuclear y partículas cada vez tienden a ser más extensos en cuanto a componentes electrónicos y detectores. Las simulaciones son realizadas para hacer una representación lo más realista posible del experimento para poder realizar estudios previos a su construcción. Esto puede ahorrar tiempo y coste en la construcción y puesta en marcha del experimento.

Una simulación es un proceso en el que diferentes algoritmos son ejecutados para obtener un resultado lo más cercano posible al mundo real. Hoy en día las simulaciones por ordenador son unas herramientas muy potentes puesto que los avances tecnológicos permiten manejar grandes cantidades de datos en una escala de tiempo relativamente corta.

Uno de los algoritmos más usados en física para realizar simulaciones es el conocido como método Monte Carlo [40]. En principio, todo problema probabilístico puede resolverse mediante este método, que está basado en la resolución de ecuaciones diferenciales que sigue un estado inicial para pasar por unos estados intermedios y llegar a un estado final. Ese estado final es el que es analizado a través de sus variables estadísticas y en virtud de la ley de grandes números.

En esta tesis se realizaron simulaciones para el estudio de la dispersión coulombiana de la radiación cósmica en materiales pesados. Hemos supuesto que esta radiación está dividida en dos grandes partes: por un lado tenemos la componente muónica y por otro lado están los electrones y fotones. Esta división viene motivada por el hecho de que los fotones, en presencia de un medio, producen pares electrón-positrón que a su vez emiten fotones por frenado comenzando así una cadena de producción de pares y emisión de fotones. Esta cadena constituye una cascada electromagnética y nos referiremos tanto a los electrones como a los fotones con el nombre de parte electromagnética.

Cuando una partícula cargada (como lo es el muon) entra en un material esta sufre un proceso aleatorio de *zarandeo* provocado por la repulsión coulombiana de su carga con la de los átomos del material. Una respresentación esquemática de la dispersión de los muones como de las ya mencionadas cascadas electromagnéticas pueden ser vistas en la Fig.(9.3)
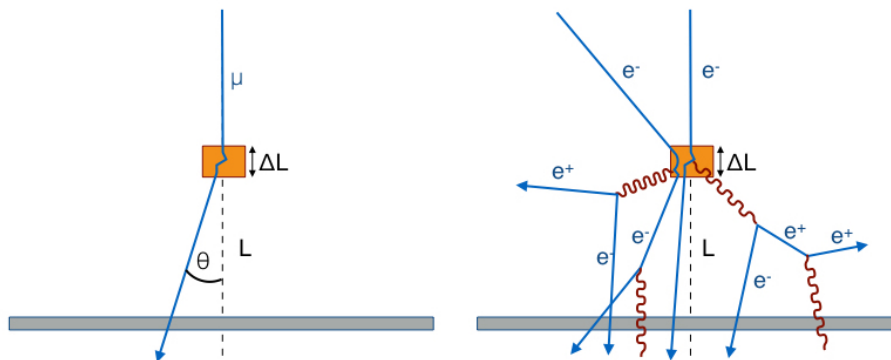
Fig. 9.3 Representación esquemática de la dispersión de la radiación cósmica en presencia de un material. A la izquierda tenemos la parte muónica y a la derecha la electromagnética, formada por fotones y electrones.

Para realizar las simulaciones hemos hecho uso del código EnsarRoot [26]. Este código es una variante oficial (o sabor) del código FairRoot [21] y supone una estructura óptima para realizar nuestras simulaciones al integrar herramientas de generación y análisis. La gran ventaja de EnsarRoot es que nos permite desarrollar nuestros detectores, realizar Monte Carlo y analizar los datos todo en un solo proceso sin necesidad de llamar a ejecutables externos. Además, ya que hereda todas las utilidades de Root y Geant, estas pueden ser usadas de manera nativa, por lo que la organización de los datos se realizar optimizando el uso de memoria computacional. Esta manera de trabajar agiliza notablemente la producción de nuevo código, ya que todas las librerias y utilidades ya creadas y probadas son reutilizadas.

Dentro de la estructura de EnsarRoot hemos simulado el comportamiento de detectores RPC como TRAGALDABAS [4]. Esta estructura geométrica ha sido reutilizada y adapatada a las particularidades del nuevo detector *muTT* para ser usado en el prototipo del tomógrafo de muones.

El observable físico sobre el que fundamentamos el análisis de los datos es el ángulo de dispersión θ de la partícula al salir del material. Sabemos que dicho ángulo se comporta de manera gaussiana en la parte central de su distribución, con desviación:

$$\theta_0 = \frac{13.6\,MeV}{\beta cp} Z\sqrt{x/X_0}\left[1 + 0.038\ln\left(x/X_0\right)\right] \tag{9.3}$$

siendo p el momento de la partícula incidente, $v = \beta c$ su velocidad y Z el número atómico del material atravesado. La cantidad X_0 es una magnitud característica del material llamada *longitud de radiación*, que se podría decir que es una "unidad natural" para casi todos los

procesos de naturaleza puramente electromagnética que tienen lugar dentro de un cierto material.

Puesto que estamos interesados únicamente en el comportamiento de las partículas dispersadas, hemos supuesto las partículas incidentes con dirección perpendicular perfectamente conocida y hemos analizado las características del ángulo de dispersión, en el caso del muón y de la dispersión y formación de secundarias en el caso de electrones y fotones. El rango de energías cinéticas de las partículas incidentes se muestran en la Tabla 9.1. En este estudio

Partícula	Energía (GeV)										
Muón	0.1	0.2	0.3	0.4	0.5	1.0	2.0	4.0	8.0	16.0	32.0
e, γ	0.1	0.2	0.3	0.4	0.5						

Table 9.1 Relación de partículas lanzadas y las correspondientes energías cinéticas usadas.

nos hemos limitado al análisis de la dispersión angular de los muones y a la producción de partículas secundarias por electrones en los diferentes blancos usando los puntos ideales de impacto y sin tener en cuenta la resolución de lectura del detector.

Sin embargo, los programas de simulación y análisis quedan listos para futuros estudios en los que se tenga en cuenta las incertidumbres ocasionadas por la "discretización" en la lectura de señales.

Simulación de observables físicos en *muTT*

La estructura reducida del *muTT*, mostrada en la Fig.9.2 nos permite realizar simulaciones sobre cantidades físicas relevantes en el sistema. Por una parte, nos sirve como bando de prueba de nuestros algoritmos de reconstrucción. Por otra, nos permite proponer observables físicos que pueden ayudar a la identificación de materiales pesados.

Para realizar estas tareas, hemos tomado como referencia el conjunto de datos etiquetados como *run 27*. La configuración experimental de este run consiste en tres ladrillos colocados en el centro del detector. Se trata de 2 litros de plomo, 2 litros de hierro y 1 litro de wolframio (Fig.9.4)

Esta configuración experimental fue simulada con EnsarRoot. A partir de los resultados de esa simulación podemos comprobar la eficacia de los métodos de reconstrucción. En particular, hemos propuesto dos métodos independientes: POCA (punto de máximo acercamiento) y TimTrack con modelo supersaeta. La diferencia entre las variables reconstruidas con ambos métodos son la base para nuestros observables físicos.
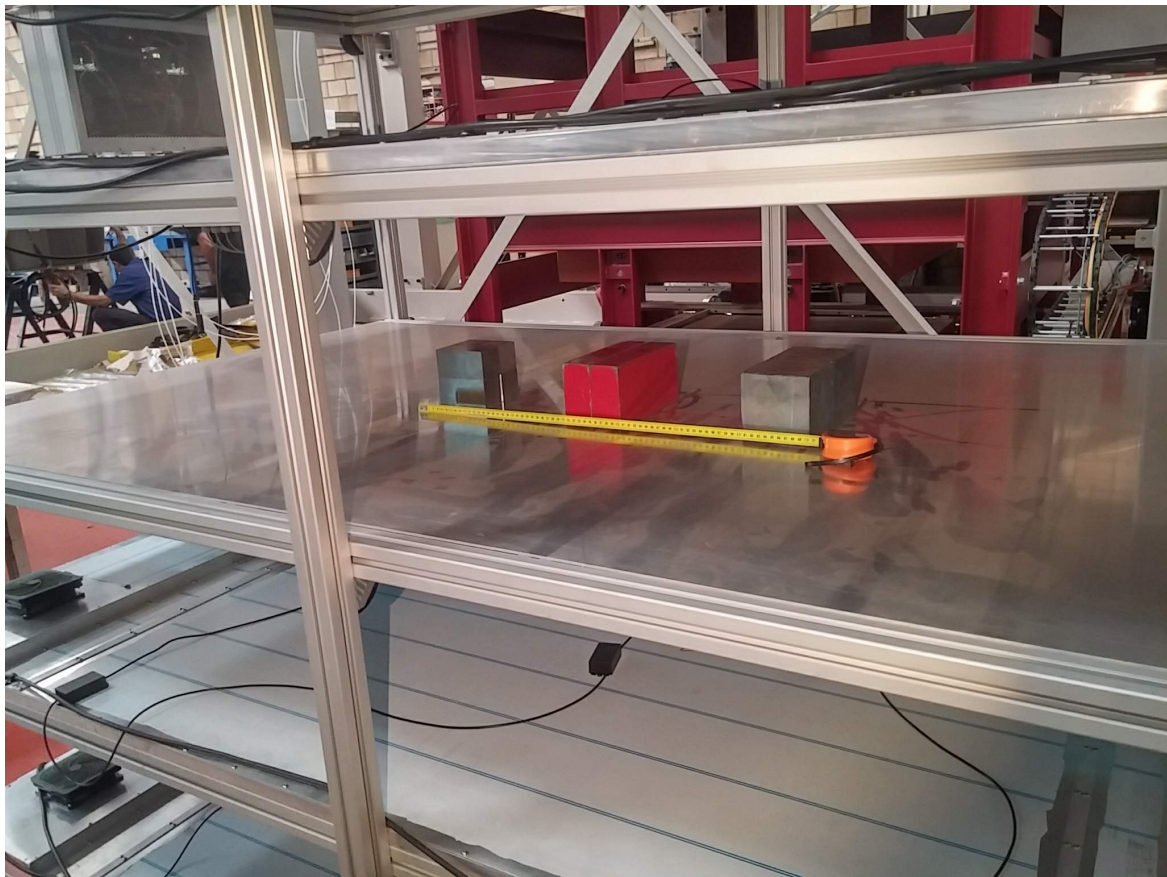
Fig. 9.4 Run número 27. Hay tres ladrillos: 2 litros de plomo (centro), dos litros de hierro (derecha) y un litro de wolframio (izquierda).

Nombre	Símbolo	Definición
Diferencia en vértices	dr	$(x_{\text{poca}} - x_{\text{ss}}, y_{\text{poca}} - y_{\text{ss}}, z_{\text{poca}} - z_{\text{ss}})$
Desviación polar	$d\theta$	$\theta_{\text{poca}} - \theta_{\text{ss}}$
Desviación polar relativa	$d\theta/dz$	$(\theta_{\text{poca}} - \theta_{\text{ss}})/(z_{\text{poca}} - z_{\text{ss}})$
Desviación esférica	$d\Omega$	$\arccos(\vec{v} \cdot \vec{w})$
Retardo temporal	dt	$t_4' - t_4$

Table 9.2 Definición de observables físicos propuestos para el estudio de materiales. Podemos diferenciar entre los basados en variables espaciales, variables angulares y variables temporales.

En la Tabla 9.2 puede encontrarse una relación de los observables estudiados. Podemos diferenciar entre observables basados en diferencias espaciales, diferencias angulares y retardo temporal.

Un apartado diferente merece la desviación polar relativa. Este observable compara diferencias angulares con diferencias en la componente vertical, z. Este observable veremos que es el que más información parece tener de cara a la identificación y separación de materiales pesados.

Calibración del detector *muTT*

El método de reconstrucción TimTrack ha sido usado por primera vez como herramienta de calibración en un detector tipo RPC. Mediante la reconstrucción de las trayectorias hemos podido establecer constantes globales en la calibración del *muTT*. Constantes como alineamiento o desplazamientos temporales han sido estudiados a través de TimTrack.

La variable clave en todo el proceso ha sido el residuo, definido como:

$$\Delta_x = x_{\text{rec}} - x_{\text{m}}$$

donde x_{rec} es una variable reconstruida y x_{m} es el valor medido. Se define exactamente igual para cualquier variable, sea espacial o temporal: siempre es la diferencia entre lo que se reconstruye y lo que se mide.

Calibración temporal

Mediante un proceso iterativo es posible obtener las constantes globales de calibración para cada plano. El *muTT* tiene dos partes diferenciadas, derecha e izquierda. Como la electrónica es diferente en cada parte, existe un desfase temporal entre ellas que debe ser corregido. El algoritmo es iterativo, y a cada paso se estudia el residuo temporal por plano. El valor medio de la distribución es usada como corrección. Como resultado, damos los valores de desfase τ para cada plano:

Plano	τ_{il} (ns)	τ_{ir} (ns)
1	0	0.126887
2	0.571072	2.16441
3	31.5284	31.9045
4	-4.66287	-4.93934

Table 9.3 Valores de los desfases temporales globales por plano. Se asume que la parte izquierda del primer plano es el valor de referencia para el resto de tiempos.

Calibración espacial

El mismo procedimiento se lleva a cabo para el caso de las correcciones espaciales. Estas constantes espaciales son consideradas como alineamiento globlal de los planos, es decir, aparecen por el hecho de que los cuatro planos no están perfectamente alineados. Estos números nos dan idea de cuánto deben desplazarse los planos para tener un alineado correcto:

Plano	Δx (mm)	Δy (mm)
1	-2.62171	-0.369186
2	1.00447	0.630083
3	2.64132	-1.28205
4	-4.04285	1.38945

Table 9.4 Valores de las constantes de alineamiento por plano.

Los valores globales de alineamiento y desfases temporales son guardados en un fichero de calibración para ser usado cada vez que se quiera analizar un evento.

Correcciones por slewing

El efecto slewing es puramente electrónico y está relacionado con la amplitud de la señal recogida por la electrónica. En efecto, la amplitud de la señal tiene que ver con la cantidad de carga depositada. Toda señal que esté discriminada por un umbral va a sufrir un desfase temporal entre el inicio del pulso y el momento de cruzar el umbral. Las señales más intensas tardan menos en llegar, por lo tanto, a mayor carga, menor desfase temporal. Esto se conoce como slewing effect, y ha de ser introducido como corrección temporal debido a la carga medida por nuestra RPC.

Correcciones temporales debido a la posición del impacto

Después de las correcciones globales pueden realizarse correcciones finas en espacio y tiempo. Nuestro detector *muTT* está compuesto por una serie de tiras de hilos. A través de TimTrack podemos acceder a la estructura interna del detector y realizar calibraciones hilo a hilo. Puesto que no todos los hilos tienen la misma longitud la señal tarda diferente tiempo dependiendo del hilo en el que se detecta la inducción. Es por ello por lo que se pueden realizar este tipo de correcciones o ajustes finos al tiempo global anteriormente estimado.

Para estimar la corrección debida a la posición hacemos uso de TimTrack para estudiar el comportamiento del residuo temporal en función de la coordenada *y* (el juego de hilos que mide el tiempo está colocado a lo largo de ese eje)
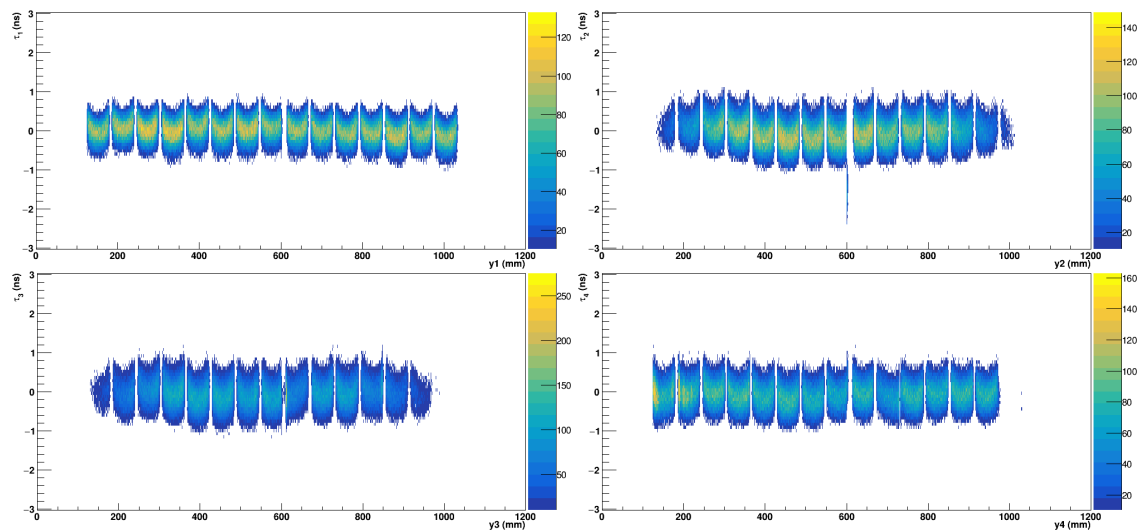
Fig. 9.5 Corrección temporal debida a la posición *y* del impacto para cada plano. Con TimTrack tenemos acceso a la estructura interna del detector y podemos realizar correcciones hilo a hilo.

Correcciones espaciales debido a la posición del impacto

De manera análoga a las correcciones temporales podemos realizar correcciones dependiendo de la coordenada espacial del impacto.

Como se puede observar, hay una serie de estructuras que aparecen en estas figuras. La manera de corregir las variables espaciales se hace a través de los perfiles de dichos histogramas.

Correcciones adicionales por software

Además de las correcciones anteriormente realizadas se propone el uso de funciones matemáticas como correcciones adicionales. Estas funciones matemáticas se obtienen a partir de los datos experimentales, mediante un ajuste de las estructuras. De esta manera, se trata de conseguir un comportamiento lo más suave posible de las variables.

Las funciones que se proponen son potencias de orden cinco y orden orden, obtenidas de manera fenomenológica.
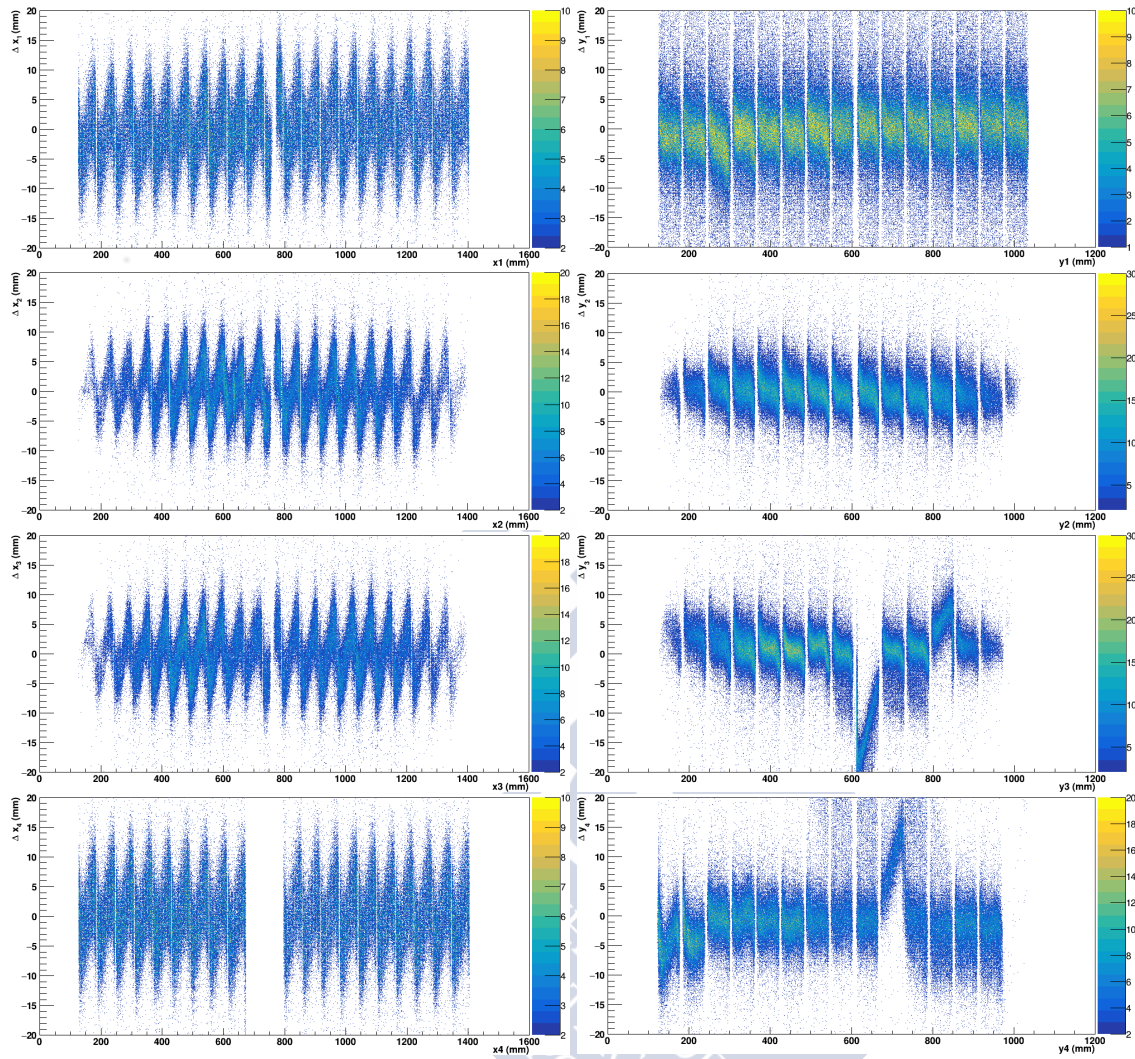
Fig. 9.6 Correcciones espaciales debido a las coordenadas espaciales del impacto.

Resumen de correcciones

Todas las correcciones deben ser aplicadas a los datos experimentales originales. La manera de hacerlo sería, una vez obtenida cada contribución individual:

$$
\begin{aligned}
x_{\text{corr}} &= x_{\text{im}} - \Delta x - \delta_x - \delta_{\text{soft}} \\
y_{\text{corr}} &= y_{\text{im}} - \Delta y - \delta_y - \delta_{\text{soft}} \\
t_{\text{corr}} &= t_{im} - \tau_{il,r} - \delta_{txi} - \delta_{tyi} - \delta_{\text{slewing}}
\end{aligned}
$$

donde a cada valor medido en el plano i, x_{im}, se hacen correcciones de desplazamiento global (Δ_x), correcciones debidas a la coordenada que ocupa (δ_x) y correcciones adicionales por software (δ_{soft}) En caso de las coordenadas temporales tenemos una corrección global

dependiendo de si el impacto ocurre en la parte derecha o izquierda del detector ($\tau_{\mathrm{il,r}}$), una corrección debida a la posición que ocupa (δ_{txi} y δ_{tyi}) y una corrección por la cantidad de carga de la señal ($\delta_{\mathrm{slewing}}$)

Reconstrucción de trazas con datos corregidos

Una vez realizadas las correcciones anteriores sobre los datos experimentales es posible realizar la reconstrucción de los eventos. El estudio de las trayectorias reconstruidas puede ayudar a determinar diferentes características intrínsecas del detector.

Usando un juego de datos sin blancos, es decir, sin materiales que dispersen los muones, es posible hacer una estimación de las resoluciones espaciales y temporales por plano. Para ello, hacemos un histograma de la distribución de residuos espaciales y temporales por plano. A partir de esas distribuciones, estimamos las desviaciones de las zonas centrales. De esta manera, damos valores de las diferentes resoluciones:

Plano	σ_x (mm)	σ_y (mm)	σ_t (ps)
1	1.28(1)	1.13(2)	98.7(2)
2	1.67(3)	1.43(3)	134.6(3)
3	1.93(4)	1.50(3)	120.4(3)
4	1.47(3)	1.18(2)	88.7(1)

Table 9.5 Valores de las resoluciones espaciales y temporales por plano. Los valores han sido obtenidos como la desviación en la parte central de la distribución de residuos.

Los resultados obtenidos son significativamente mejores que los proporcionados por el fabricante. Esto quiere decir que con nuestro algoritmo de reconstrucción podemos alcanzar resoluciones mucho mejores que las inicialmente esperadas.

Análisis de los datos experimentales

En esta tesis se realizó el análisis de los datos experimentales contenidos en el run número 27. Este run contine tres bloques de diferentes materiales y tamaños. Por un lado, tenemos dos litros de plomo situados en el centro del detector. Flanqueando el ladrillo central hay dos litros de hierro y un litro de wolframio.

Para realizar las imágenes de los bloques se usaron dos algoritmos. Por una parte, el método POCA y por otra el TimTrack con modelo supersaeta.

Ambos métodos proporcionan resultados muy parecidos en cuanto a imagen se refiere, así que tomamos como referencia el método TimTrack para el resto del análisis.

A los métodos de reconstrucción hay que añadir una serie de cortes en las variables que van a ayudar a limpiar las imágenes finales. En particular, se realizaron cortes en el ángulo de dispersión, en la diferencia dz y en el observable *métrica*[47]:

$$\tilde{m}_{ij} = \frac{|dr|}{(\theta_i \tilde{p}_i) \cdot (\theta_j \tilde{p}_j)}$$

donde en esta ecuación $|dr|$ es el módulo del vector desplazamiento (definido como diferencia entre métodos POCA y TimTrack), θ_i y θ_j son los ángulos de las trayectorias en las partes superior e inferior (entrada y salida) y $\tilde{p}$ son los momentos normalizados.
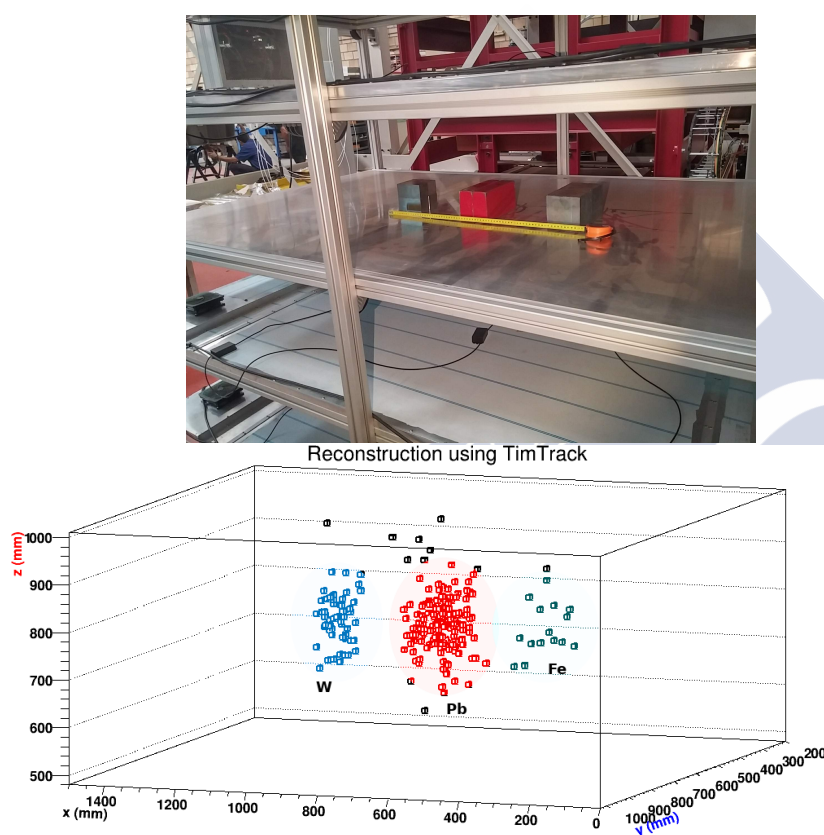
Fig. 9.7 Reconstrucción de los bloques del run 27. Hay tres ladrillos diferentes: Fe, Pb y W. En la parte superior de la figura pude verse una foto de la configuración real del run 27.

En la Fig.9.7 pueden apreciarse los tres ladrillos reconstruidos y comparados con la forma real del run 27.

Como vemos, después de realizar los cortes oportunos la imagen es bastante clara y hay vértices que corresponden a aire que son eliminados de la muestra.

Proponemos, además, el observable $d\theta/dz$ (desviación polar relativa) como el candidato para la identificación de los materiales.
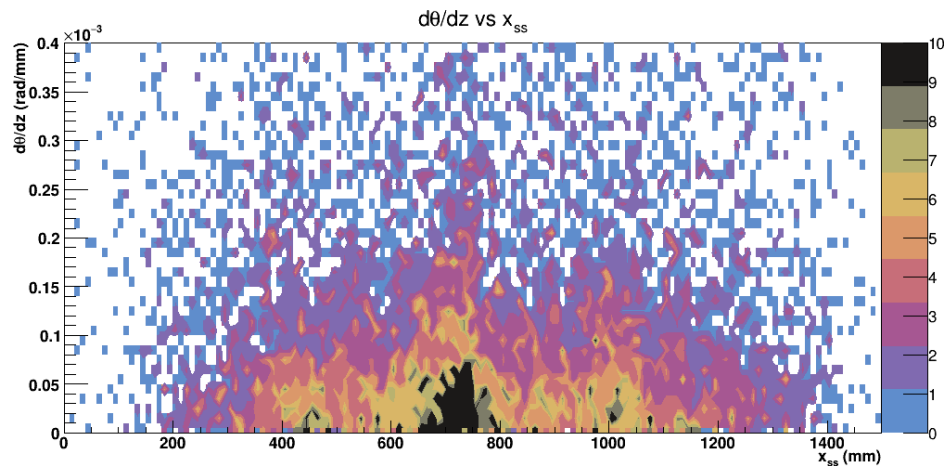
Fig. 9.8 Desviación polar relativa en función de la coordenada reconstruida x_{ss} para el run 27. Como puede observarse, hay tres estructuras bien definidas y que dependen de la naturaleza del material.

En la Fig.9.8 puede verse la representación del observable *desviación polar relativa* en función de la coordenada reconstruida x_{ss}. Como puede observarse, hay tres estructuras bien definidas que corresponden a los tres materiales presentes en el run 27. La región alrededor de los 400 mm corresponde al hierro, en la parte central está el plomo y sobre los 1000 mm se encuentra el wolframio. Como podemos observar, hay una dependencia del observable con el material. A pesar de que el ladrillo de wolframio es la mitad que el de hierro puede notarse que la desviación polar relativa es mayor.

Como conclusión, el observable que corresponde a la desviación polar relativa puede ser buen candidato para la identificación de materiales pesados.

References

[1] T. Wulf. Beobachtungen über Strahlung hoher Durchdringungsfähigkeit auf dem Eiffelturm. *Physikalische Zeitschrschift*, 11:811–813, 1910.

[2] V.F. Hess. Über beobachtungen der durchdringenden Strahlung bei sieben Freiballonfahrten. *Physikalische Zeitschrschift*, 13:1084–1091, 1912.

[3] The pierre auger cosmic ray observatory. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 798:172 – 213, 2015.

[4] J.A. Garzón et al. TRAGALDABAS. first results on cosmic ray studies and their relation with the solar activity, the earth magnetic field and the atmospheric properties. *XXV ECRS 2016 Proceedings - eConf TBA*, 2017.

[5] D. García. PhD thesis, Universidad de Santiago de Compostela, Departamento de Física de Partículas.

[6] I. Garrigós. PhD thesis, Universidad de Santiago de Compostela, Departamento de Física de Partículas.

[7] C. Patrignani et al. Review of Particle Physics. *Chin. Phys.*, C40(10):100001, 2016.

[8] T.K. Gaisser, R. Engel, and E. Resconi. *Cosmic Rays and Particle Physics*. Cambridge University Press, 2 edition, 2016.

[9] A.M. Galper et al. The PAMELA experiment: a decade of cosmic ray physics in space. *Journal of Physics: Conference Series*, 798(1):012033, 2017.

[10] IceCube Observatory. https://icecube.wisc.edu/.

[11] S. Chakraborty and I. Izaguirre. Star-forming galaxies as the origin of IceCube neutrinos: Reconciliation with Fermi-LAT gamma rays. https://arxiv.org/abs/hep-ph/9609357, 2016.

[12] L. Cremonesi. Cosmic-ray muon tomography for anti-terrorism applications. Master's thesis, University College London, 2011.

[13] Gran Telescopio Milimétrico Alfonso Serrano. http://www.lmtgtm.org.

[14] Yangbajing International Cosmic Ray Observatory in Tibet. http://www.ircip.cn/bbx/999722-999726.html?id=26645&newsid=639942.

[15] P. Greider. *Cosmic rays at Earth: Researcher's Reference Manual and Data Book*. Elsevier, 2001.

[16] M. Abbrescia, L. Benussi, D. Piccolo, S. Bianco, M. Ferrini, S. Muhammad, L. Passamonti, D. Pierluigi, D. Piccolo, F. Primavera, A. Russo, and G. Saviano. Eco-friendly gas mixtures for resistive plate chambers based on tetrafluoropropene and helium. *Journal of Instrumentation*, 11(08):P08019, 2016.

[17] L.W. Alvarez, J.A. Anderson, F. El Bedwei, J. Burkhard, A. Fakhry, A. Girgis, A. Goneid, F. Hassan, D. Iverson, G. Lynch, Z. Miligy, A. Hilmy Moussa, M.-S., and L. Yazolino. Search for hidden chambers in the pyramids. *Science*, 167(3919):832–839, 1970.

[18] G. Audi, W.G. Davies, and G.E. Lee-Whiting. A method of determining the relative importance of particular data on selected parameters in the least-squares analysis of experimental data. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 249(2):443–450, 1986.

[19] P. Baesso, D. Cussans, C. Thomay, J.J. Velthuis, J. Burns, C. Steer, and S. Quillin. A high resolution resistive plate chamber tracking system developed for cosmic ray muon tomography. *Journal of Instrumentation*, 8(08):P08006, 2013.

[20] D. Belver, P. Cabanelas, D. Domínguez, J.A. Garzón, G. Kornakov, and M.F. Morales. TRASGO: A proposal for a timing RPCs based detector for analyzing cosmic ray air showers. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 661:S163 – S167, 2012.

[21] D. Bertini, M. A-Turany, I. Koenig, and F. Uhlig. The fair simulation and analysis framework. *Journal of Physics: Conference Series*, 119(3):032011, 2008.

[22] H.A. Bethe. Molière's theory of multiple scattering. *Phys. Rev.*, 89:1256–1266, 1953.

[23] A. Blanco, P. Fonte, L. Lopes, P. Martins, J. Michel, M. Palka, M. Kajetanowicz, G. Korcyl, M. Traxler, and R. Marques. Toftracker: gaseous detector with bidimensional tracking and time-of-flight capabilities. *Journal of Instrumentation*, 7(11):P11012, 2012.

[24] T.T. Böhlen, F. Cerutti, M.P.W. Chin, A. Fassò, A. Ferrari, P.G. Ortega, A. Mairani, P.R. Sala, G. Smirnov, and V. Vlachoudis. The FLUKA code: Developments and challenges for high energy and medical applications. *Nuclear Data Sheets*, 120:211–214, 2014.

[25] R. Brun and F. Rademakers. ROOT - An object oriented data analysis framework. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 389(1):81–86, 1997.

[26] P. Cabanelas, H. Alvarez-Pol, E. Galiana, Y. González-Rozas, and SATNuRSE workpackage of ENSAR2. EnsarRoot: The framework for simulation and data analysis for ENSAR. *Journal of Physics: Conference Series*, 1024(1):012038, 2018.

[27] R. Cardarelli, R. Santonico, A. Di Biagio, and A. Lucci. Progress in resistive plate coun-ters. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 263(1):20–25, 1988.

[28] G. Cattani and the RPC group. The resistive plate chambers of the atlas experiment: performance studies. *Journal of Physics: Conference Series*, 280(1):012001, 2011.

[29] G. Chiodini, M.R. Coluccia, E. Gorini, F. Grancagnolo, and M. Primavera. Laser beam characterization of the ATLAS RPC gas mixture. *Nuclear Physics B - Proceedings Supplements*, 172:284 – 288, 2007. Proceedings of the 10th Topical Seminar on Innovative Particle and Radiation Detectors.

[30] I. Fröhlich et al. A general purpose trigger and readout board for hades and fair-experiments. *IEEE Transactions on Nuclear Science*, 55(1):59–66, Feb 2008.

[31] J.A. Garzón, P. Cabanelas, D. González-Díaz, and T. Kurtukian-Nieto. TimTrack: A new concept for the tracking of charged particles with timing detectors. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 629, 02 2011.

[32] J.A. Garzón and P. Cabanelas. TimTrack: A matrix formalism for a fast time and track reconstruction with timing detectors. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 661(Supplement 1):S210–S213, 2012.

[33] S. Agostinelli et al. Geant4, a simulation toolkit. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 506(3):250–303, 2003.

[34] J. Allison et al. Recent developments in Geant4. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 835:186–225, 2016.

[35] E.P. George. Cosmic rays measure overburden of tunnel. *Commonwealth Engineer*, pages 455–457, 1955.

[36] D.E. Groom, N.V. Mokhov, and S.I. Striganov. Muon stopping power and range tables 10 mev–100 tev. *Atomic Data and Nuclear Data Tables*, 78(2):183–356, 2001.

[37] C.F. Hagmann, D.C. Lange, and D.J. Wright. Cosmic-ray shower generator (CRY) for Monte Carlo transport codes. *2007 IEEE Nuclear Science Symposium Conference Record*, 2:1143–1146, 2007.

[38] N. Kume, H. Miyadera, C.L. Morris, J. Bacon, K.N. Borozdin, J.M. Durham, K. Fuzita, E. Guardincerri, M. Izumi, K. Nakayama, M. Saltus, T. Sugita, K. Takakura, and K. Yoshioka. Muon trackers for imaging a nuclear reactor. *Journal of Instrumentation*, 11(09):P09008, 2016.

[39] J. Marteau, C. Bruno, D. Gibertand, J.-C. Ianigro, K. Jourde, B. Kergosien, and P. Rolland. Muon tomography applied to active volcanoes. *PoS*, PhotoDet2015, 2016.

[40] N. Metropolis and S. Ulam. The Monte Carlo Method. *Journal of the American Statistical Association*, 44(247):335–341, 1949.

[41] F. James and M. Roos. Minuit - a system for function minimization and analysis of the parameter errors and correlations. *Computer Physics Communications*, 10(6):343 – 367, 1975.

[42] K. Morishima, M. Kuno, A. Nishio, et al. Discovery of a big void in Khufu's pyramid by observation of cosmic-ray muons. *Nature*, 552:386–390, December 2017.

[43] S. Riggi, V. Antonuccio-Delogu, M. Bandieramonte, U. Becciani, A. Costa, P. La Rocca, P. Massimino, C. Petta, C. Pistagna, F. Riggi, E. Sciacca, and F. Vitello. Muon tomography imaging algorithms for nuclear threat detection inside large volume containers with the muon portal detector. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 728(Supplement C):59–68, 2013.

[44] D. Rossi and H. Simon. A closed-circuit gas recycling system for rpc detectors. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 661:S230 – S233, 2012. X. Workshop on Resistive Plate Chambers and Related Detectors (RPC 2010).

[45] R. Santonico and R. Cardarelli. Development of resistive plate counters. *Nuclear Instruments and Methods in Physics Research*, 187(2):377–380, 1981.

[46] L.J. Schultz, G.S. Blanpied, K.N. Borozdin, A.M. Fraser, N.W. Hengartner, A.V. Klimenko, C.L. Morris, C. Orum, and M.J. Sossong. Statistical reconstruction for cosmic ray muon tomography. *IEEE Transactions on Image Processing*, 16(8):1985–1993, 2007.

[47] C. Thomay, J.J. Velthuis, P. Baesso, D. Cussans, P.A.W. Morris, C. Steer, et al. A binned clustering algorithm to detect high-Z material using cosmic muons. *Journal of Instrumentation*, 8(10):P10013, 2013.

[48] C. Thomay, J. Velthuis, T. Poffley, P. Baesso, D. Cussans, and L. Frazão. Passive 3d imaging of nuclear waste containers with muon scattering tomography. *Journal of Instrumentation*, 11(03):P03008, 2016.

[49] A. Blanco, J.J. Blanco, et al. TRAGALDABAS: a new RPC based detector for the regular study of cosmic rays. *Journal of Instrumentation*, 9(09):C09027, 2014.

[50] B.C. Wilson and G. Adam. A Monte Carlo model for the absorption and flux distributions of light in tissue. *Medical Physics*, 10(6):824–830, 1983.

[51] P. Cabanelas. *Software development and performance analysis of the HADES Resistive Plate Chamber time-of-flight detector at GSI*. PhD thesis, Universidad de Santiago de Compostela, Departamento de Física de Partículas, 2011.

[52] D. Belver. *The front-end electronics of the HADES timing RPCs wall, design, development and performances analysis*. PhD thesis, Universidad de Santiago de Compostela, Departamento de Física de Partículas, 2010.

[53] D. González-Díaz. *Research and Developments on Timing RPCs. Application to the ESTRELA Detector of the HADES experiment at GSI*. PhD thesis, Universidad de Santiago de Compostela, Departamento de Física de Partículas, 2006.

[54] G. Kornakov. *New advances and developments on the RPC tof wall of the HADES experiment at GSI*. PhD thesis, Universidad de Santiago de Compostela, Departamento de Física de Partículas, 2012.

[55] C. Lippmann. *Detector physics of Resistive Plate Chambers*. PhD thesis, Johann Wolfgang Goethe-Universität Frankfurt am Main, 2003.

[56] L. Schultz. *Cosmic ray muon radiography*. PhD thesis, Portland State University, Electrical and Computer Engineering, 2003.

[57] F. James. *Statistical Methods in Experimental Physics*. World Scientific, 2006.

[58] L. Lyons. *Statistics for nuclear and particle physics*. Cambridge University Press, 1986.

[59] D. Peña. *Análisis de datos multivariantes*. McGraw-Hill, 2002.

[60] S. Tavernier. *Experimental Techniques in Nuclear and Particle Physics*. Springer-Verlag Berlin Heidelberg, 2010.

[61] H. Álvarez Pol. Física computacional avanzada. Lecture notes at USC, 2017.

[62] HIDRONAV research. Macro scanner project. http://www.hidronav.es/page4.html.

[63] LIP-Coimbra. http://www.lip.pt/index.php?section=about&poles&page=coimbra.

[64] E. Winsberg. Computer simulations in science. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, summer 2018 edition, 2018.

Appendix A

RPC geometry source code

In this appendix we show the source code of a RPC detector as the used in the TRAGALD-ABAS experiment, **tragaldabas.C**:

```cpp
#include <iomanip>
#include <iostream>
#include "TGeoManager.h"
#include "TMath.h"

// Create Matrix Unity
TGeoRotation *fGlobalRot = new TGeoRotation();

// Create a null translation
TGeoTranslation *fGlobalTrans = new TGeoTranslation();
TGeoRotation *fRefRot = NULL;

TGeoManager*  gGeoMan  = NULL;

Double_t fThetaX = 0.;
Double_t fThetaY = 0.;
Double_t fThetaZ = 0.;
Double_t fPhi   = 0.;
Double_t fTheta = 0.;
Double_t fPsi   = 0.;
Double_t fX = 0.;
Double_t fY = 0.;
```

```
Double_t fZ = 0.;
Bool_t fLocalTrans = kFALSE;
Bool_t fLabTrans = kFALSE;


TGeoCombiTrans* GetGlobalPosition(TGeoCombiTrans *fRef);


// -------    Main function for creating the geo -------------------
void tragaldabas(const char* geoTag = "")
{


fGlobalTrans->SetTranslation(0.0,0.0,0.0);


// Usage: select the geoTag when calling the macro for a new CALIFA geo
// Possible geoTag values:
// geoTag            - Short description
// 0                 - Initial Tragaldabas basic geometry



// -------    Load media from media file  -----------------------
FairGeoLoader*    geoLoad = new FairGeoLoader("TGeo","FairGeoLoader");
FairGeoInterface* geoFace = geoLoad->getGeoInterface();
TString geoPath = gSystem->Getenv("VMCWORKDIR");
TString medFile = geoPath + "/tragaldabas/geometry/media_ensar.geo";
geoFace->setMediaFile(medFile);
geoFace->readMedia();
gGeoMan = gGeoManager;
//----------------------------------------------


// -------    Geometry file name (output)    -------------------
TString geoFileName = geoPath + "/tragaldabas/geometry/tragaldabas_";
geoFileName = geoFileName + geoTag + ".geo.root";
// --------------------------------------------------------


// ----------    Get and create the required media    ------------
FairGeoMedia*   geoMedia = geoFace->getMedia();
FairGeoBuilder* geoBuild = geoLoad->getGeoBuilder();
```

```cpp
FairGeoMedium* mAir       = geoMedia->getMedium("Air");
if ( ! mAir ) Fatal("Main", "FairMedium Air not found");
geoBuild->createMedium(mAir);
TGeoMedium* pAirMedium = gGeoMan->GetMedium("Air");
if ( ! pAirMedium ) Fatal("Main", "Medium Air not found");

FairGeoMedium* mAluminium      = geoMedia->getMedium("aluminium");
if ( ! mAluminium ) Fatal("Main", "FairMedium Aluminium not found");
geoBuild->createMedium(mAluminium);
TGeoMedium* pAluminiumMedium = gGeoMan->GetMedium("aluminium");
if ( ! pAluminiumMedium ) Fatal("Main", "Medium Aluminium not found");

FairGeoMedium* mCopper = geoMedia->getMedium("copper");
if ( ! mCopper ) Fatal("Main", "FairMedium Copper not found");
geoBuild->createMedium(mCopper);
TGeoMedium* pCopperMedium = gGeoMan->GetMedium("copper");
if ( ! pCopperMedium ) Fatal("Main", "Medium Copper not found");

FairGeoMedium* mCar = geoMedia->getMedium("CarbonFibre");
if ( ! mCar ) Fatal("Main", "FairMedium CarbonFibre not found");
geoBuild->createMedium(mCar);
TGeoMedium* pCarbonFibreMedium = gGeoMan->GetMedium("CarbonFibre");
if ( ! pCarbonFibreMedium ) Fatal("Main", "Medium CarbonFibre not found");

FairGeoMedium* mMethacrylate =
geoMedia->getMedium("methacrylate");
if ( ! mMethacrylate ) Fatal("Main", "FairMedium Methacrylate not found");
geoBuild->createMedium(mMethacrylate);
TGeoMedium* pMethacrylateMedium = gGeoMan->GetMedium("methacrylate");
if ( ! pMethacrylateMedium ) Fatal("Main",
"Medium Methacrylate not found");

FairGeoMedium* mRPCGlass = geoMedia->getMedium("RPCglass");
if ( ! mRPCGlass ) Fatal("Main",
"FairMedium RPCGlass not found");
```

```
geoBuild->createMedium(mRPCGlass);
TGeoMedium* pRPCGlassMedium = gGeoMan->GetMedium("RPCglass");
if ( ! pRPCGlassMedium ) Fatal("Main", "Medium RPCGlass not found");


FairGeoMedium* mRPCGasR134A = geoMedia->getMedium("RPCgas");
if ( ! mRPCGasR134A ) Fatal("Main", "FairMedium RPCGasR134A not found");
geoBuild->createMedium(mRPCGasR134A);
TGeoMedium* pRPCGasR134AMedium = gGeoMan->GetMedium("RPCgas");
if ( ! pRPCGasR134AMedium ) Fatal("Main", "Medium RPCGasR134A not found");


FairGeoMedium* mTfl = geoMedia->getMedium("Tefflon");
if ( ! mTfl ) Fatal("Main", "FairMedium Tefflon not found");
geoBuild->createMedium(mTfl);
TGeoMedium* pWrappingMedium = gGeoMan->GetMedium("Tefflon");
if ( ! pWrappingMedium ) Fatal("Main", "Medium Tefflon not found");
// -----------------------------------------------------------


// -------------    Create geometry and top volume  --------------------
gGeoMan = (TGeoManager*)gROOT->FindObject("FAIRGeom");
gGeoMan->SetName("TragaldabasGeom");
TGeoVolume* top = new TGeoVolumeAssembly("TRA_World");
gGeoMan->SetTopVolume(top);
// -------------------------------------------------


//WORLD
//TGeoVolume *pAWorld  =  top;


// Defintion of the Mother Volume (cube of 2.0 meters side)
Double_t length = 400.;
TGeoShape *pTragaShape = new TGeoBBox("Traga_boxOut",
length/2.0,length/2.0,length/2.0);
TGeoVolume* pTragaWorld = new TGeoVolume("TragaWorld",pTragaShape, pAirMedium);


TGeoCombiTrans *t0 = new TGeoCombiTrans();
TGeoCombiTrans *pGlobalc = GetGlobalPosition(t0);
```

```
// add the cube (pTragaWorld) as Mother Volume
top->AddNode(pTragaWorld, 0, pGlobalc);

//MODULE KeepIn Volume, 1 cm larger in each dimension than aluminum box
TGeoVolume *RPC_KeepInVolume=gGeoManager->MakeBox("RPC_KeepInVolume", pAirMedium,
166.0/2,129.5/2,3.6/2);
RPC_KeepInVolume->SetVisibility(kFALSE); //invisible KIV

TGeoRotation *rotUni = new TGeoRotation(); //unitary rotation

pTragaWorld->AddNode(RPC_KeepInVolume,0,new TGeoCombiTrans(0.,0.,-186.0,rotUni));
pTragaWorld->AddNode(RPC_KeepInVolume,1,new TGeoCombiTrans(0.,0.,-133.5,rotUni));
pTragaWorld->AddNode(RPC_KeepInVolume,2,new TGeoCombiTrans(0.,0.,-95.6,rotUni));
pTragaWorld->AddNode(RPC_KeepInVolume,3,new TGeoCombiTrans(0.,0.,-11.9,rotUni));

//Aluminum box containing the whole chamber, 3mm thick, laterals 5mm thick
TGeoVolume *RPC_AluBox=gGeoManager->MakeBox("RPC_AluBox", pAluminiumMedium,
165.0/2,128.5/2,2.6/2);
RPC_KeepInVolume->AddNode(RPC_AluBox,0,new TGeoCombiTrans(0.,0.,0.,rotUni));
RPC_AluBox->SetVisibility(kTRUE); //invisible KIV
RPC_AluBox->SetLineColor(kRed);

TGeoVolume *RPC_AluInBox=gGeoManager->MakeBox("RPC_AluInBox", pAirMedium,
164.0/2,127.5/2,2.0/2);
RPC_AluBox->AddNode(RPC_AluInBox,0,new TGeoCombiTrans(0.,0.,0,rotUni));
RPC_AluInBox->SetVisibility(kFALSE); //invisible interior volume (air)

//Foam (box) between aluminium box and copper PCB
TGeoVolume *RPC_FoamBox=gGeoManager->MakeBox("RPC_FoamBox", pAirMedium,
163.0/2,126.5/2,0.87/2);
RPC_AluInBox->AddNode(RPC_FoamBox,0,new TGeoCombiTrans(0.,0.,0.565,rotUni));
RPC_FoamBox->SetLineColor(kYellow);

//copper (box) between foam box and the PCB
//NOTE: It is not divided at this stage. It is also NOT the sensitive
//material, but signals are going to be taken from the gas, dividing it
```

```
//with the proper copper-pad segmentation.
TGeoVolume *RPC_CopperBox=gGeoManager->MakeBox("RPC_CopperBox", pCopperMedium,
163.0/2,126.5/2,0.003/2);
RPC_AluInBox->AddNode(RPC_CopperBox,0,new TGeoCombiTrans(0.,0.,0.1285,rotUni));
RPC_CopperBox->SetLineColor(kRed);

//PCB (box) between copper and the metacrilate
TGeoVolume *RPC_PCBBox=gGeoManager->MakeBox("RPC_PCBBox", pCarbonFibreMedium,
163.0/2,126.5/2,0.157/2);
RPC_AluInBox->AddNode(RPC_PCBBox,0,new TGeoCombiTrans(0.,0.,0.0485,rotUni));
RPC_PCBBox->SetLineColor(kGreen);


//Metacrilate box containing gas and glass
TGeoVolume *RPC_MetaBox=gGeoManager->MakeBox("RPC_MetaBox", pMethacrylateMedium,
163.0/2,126.5/2,0.97/2);
RPC_AluInBox->AddNode(RPC_MetaBox,0,new TGeoCombiTrans(0.,0.,-0.515,rotUni));
RPC_MetaBox->SetLineColor(kWhite);

TGeoVolume *RPC_MetaInBox=gGeoManager->MakeBox("RPC_MetaInBox", pRPCGasR134AMedium,
160.0/2,123.5/2,0.77/2);
RPC_MetaBox->AddNode(RPC_MetaInBox,0,new TGeoCombiTrans(0.,0.,0.,rotUni));
RPC_MetaInBox->SetVisibility(kFALSE); //invisible interior volume (gas)

//Glass planes in the gas (medium of RPC_MetaInBox volume)
TGeoVolume *RPC_GlassBox=gGeoManager->MakeBox("RPC_GlassBox", pRPCGlassMedium,
155.5/2,122.5/2,0.19/2);
RPC_MetaInBox->AddNode(RPC_GlassBox,0,new TGeoCombiTrans(0.,0.,-0.29,rotUni));
RPC_MetaInBox->AddNode(RPC_GlassBox,1,new TGeoCombiTrans(0.,0.,0.,rotUni));
RPC_MetaInBox->AddNode(RPC_GlassBox,2,new TGeoCombiTrans(0.,0.,0.29,rotUni));
RPC_GlassBox->SetLineColor(kBlue);

//Gas (planes) already in the gas (sensitive volumes where ionization takes place)
TGeoVolume *RPC_GasBox=gGeoManager->MakeBox("RPC_GasBox", pRPCGasR134AMedium,
151.2/2,121.0/2,0.1/2);
RPC_MetaInBox->AddNode(RPC_GasBox,0,new TGeoCombiTrans(0.,0.,-0.145,rotUni));
```

```
RPC_MetaInBox->AddNode(RPC_GasBox,1,new TGeoCombiTrans(0.,0.,0.145,rotUni));
RPC_GasBox->SetVisibility(kFALSE); //invisible interior volume (gas)

// --------------   Finish -------------------------------------
gGeoMan->CloseGeometry();
gGeoMan->CheckOverlaps(0.001);
gGeoMan->PrintOverlaps();
gGeoMan->Test();

TFile* geoFile = new TFile(geoFileName, "RECREATE");
top->Write();
geoFile->Close();
// -----------------------------------------------------------
}
//-------------------------------------------------------------

// -------   Global positioning  ------------------------------
TGeoCombiTrans* GetGlobalPosition(TGeoCombiTrans *fRef)
{
if (fLocalTrans == kTRUE ) {

if ( ( fThetaX == 0 )  && ( fThetaY==0 )  && ( fThetaZ == 0 )
&&
( fX == 0 ) && ( fY == 0 ) && ( fZ == 0 )
)  return fRef;

// X axis
Double_t xAxis[3] = { 1. , 0. , 0. };
Double_t yAxis[3] = { 0. , 1. , 0. };
Double_t zAxis[3] = { 0. , 0. , 1. };
// Reference Rotation
fRefRot = fRef->GetRotation();

if (fRefRot) {
Double_t mX[3] = {0.,0.,0.};
Double_t mY[3] = {0.,0.,0.};
```

```
Double_t mZ[3] = {0.,0.,0.};


fRefRot->LocalToMasterVect(xAxis,mX);
fRefRot->LocalToMasterVect(yAxis,mY);
fRefRot->LocalToMasterVect(zAxis,mZ);


Double_t a[4]={ mX[0],mX[1],mX[2], fThetaX };
Double_t b[4]={ mY[0],mY[1],mY[2], fThetaY };
Double_t c[4]={ mZ[0],mZ[1],mZ[2], fThetaZ };


ROOT::Math::AxisAngle aX(a,a+4);
ROOT::Math::AxisAngle aY(b,b+4);
ROOT::Math::AxisAngle aZ(c,c+4);


ROOT::Math::Rotation3D fMatX( aX );
ROOT::Math::Rotation3D fMatY( aY );
ROOT::Math::Rotation3D fMatZ( aZ );


ROOT::Math::Rotation3D  fRotXYZ = (fMatZ * (fMatY * fMatX));

//cout << fRotXYZ << endl;

Double_t fRotable[9]={0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0};
fRotXYZ.GetComponents(
fRotable[0],fRotable[3],fRotable[6],
fRotable[1],fRotable[4],fRotable[7],
fRotable[2],fRotable[5],fRotable[8]
);
TGeoRotation *pRot = new TGeoRotation();
pRot->SetMatrix(fRotable);
TGeoCombiTrans *pTmp = new TGeoCombiTrans(*fGlobalTrans,*pRot);

TGeoRotation rot_id;
rot_id.SetAngles(0.0,0.0,0.0);

TGeoCombiTrans c1;
```

```
c1.SetRotation(rot_id);
const Double_t *t = pTmp->GetTranslation();
c1.SetTranslation(t[0],t[1],t[2]);

TGeoCombiTrans c2;
c2.SetRotation(rot_id);
const Double_t *tt = fRefRot->GetTranslation();
c2.SetTranslation(tt[0],tt[1],tt[2]);

TGeoCombiTrans cc = c1 * c2 ;

TGeoCombiTrans c3;
c3.SetRotation(pTmp->GetRotation());
TGeoCombiTrans c4;
c4.SetRotation(fRefRot);

TGeoCombiTrans ccc = c3 * c4;

TGeoCombiTrans pGlobal;
pGlobal.SetRotation(ccc.GetRotation());
const Double_t *allt = cc.GetTranslation();
pGlobal.SetTranslation(allt[0],allt[1],allt[2]);

return  ( new TGeoCombiTrans( pGlobal ) );

}else{
cout << "-E- R3BDetector::GetGlobalPosition() \
No. Ref. Transformation defined ! " << endl;
cout << "-E- R3BDetector::GetGlobalPosition() \
cannot create Local Transformation " << endl;
return NULL;
} //! fRefRot

} else {
// Lab Transf.
if ( ( fPhi == 0 )  && ( fTheta==0 )  && ( fPsi == 0 )
```

```
&&
( fX == 0 ) && ( fY == 0 ) && ( fZ == 0 )
)  return fRef;

return ( new TGeoCombiTrans(*fGlobalTrans,*fGlobalRot) );
}
}
```

Appendix B

Complementary figures to the systematic studies of physical observables

In this appendix we show complementary figures corresponding to systematic studies of different observables presented along the thesis. In particular, we show the studies of the relative polar deviation $d\theta/dz$ and the $d\Omega$ observable.

The figures were done for the materials of interest (Fe, Pb, and W) Each material had different thickness (1 cm, 5 cm, 10 cm, and 20 cm) and were used vertical muons of different energies. The energy ranges are labelled as *low* (from 0.1 GeV to 1.0 GeV), *intermediate* (from 1 GeV to 3 GeV), and *high* (from 3 GeV to 6 GeV) following the definition of Table 5.2.

B.1   Study of the integrated $d\theta/dz$

The integrated relative polar deviation, $d\theta/dz$, is the sum of all the events contained in an interval Δx. This interval is the length of the brick, and therefore we show the total contribution inside the material.

Integrated $d\theta/dz$ for low energies (from 0.1 GeV to 1 GeV)

Fe

Pb

W
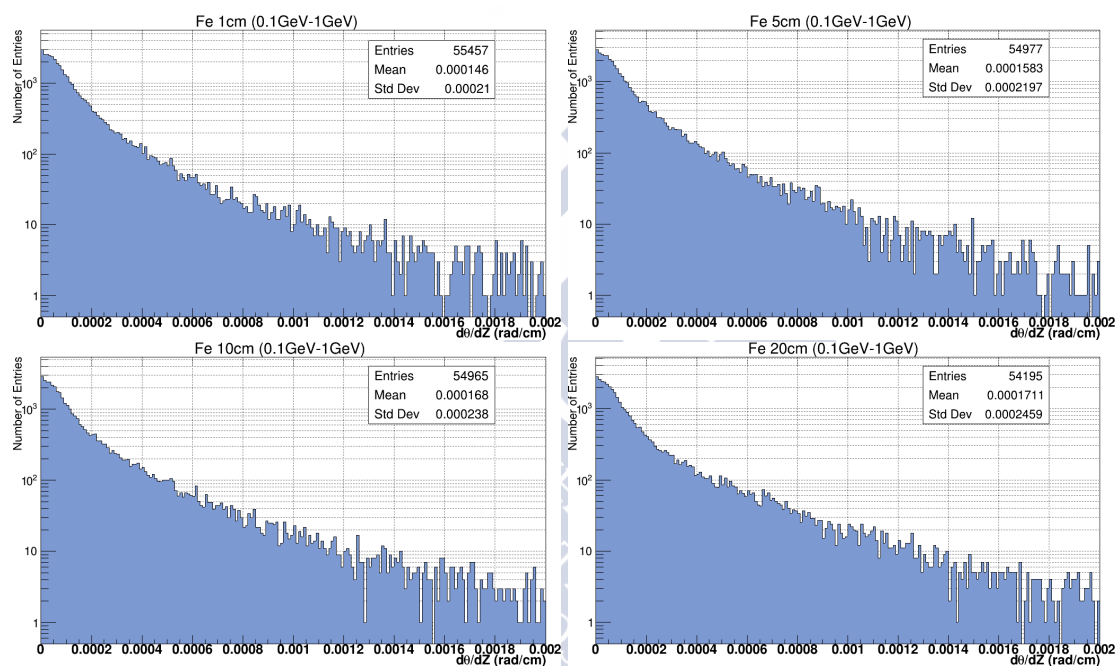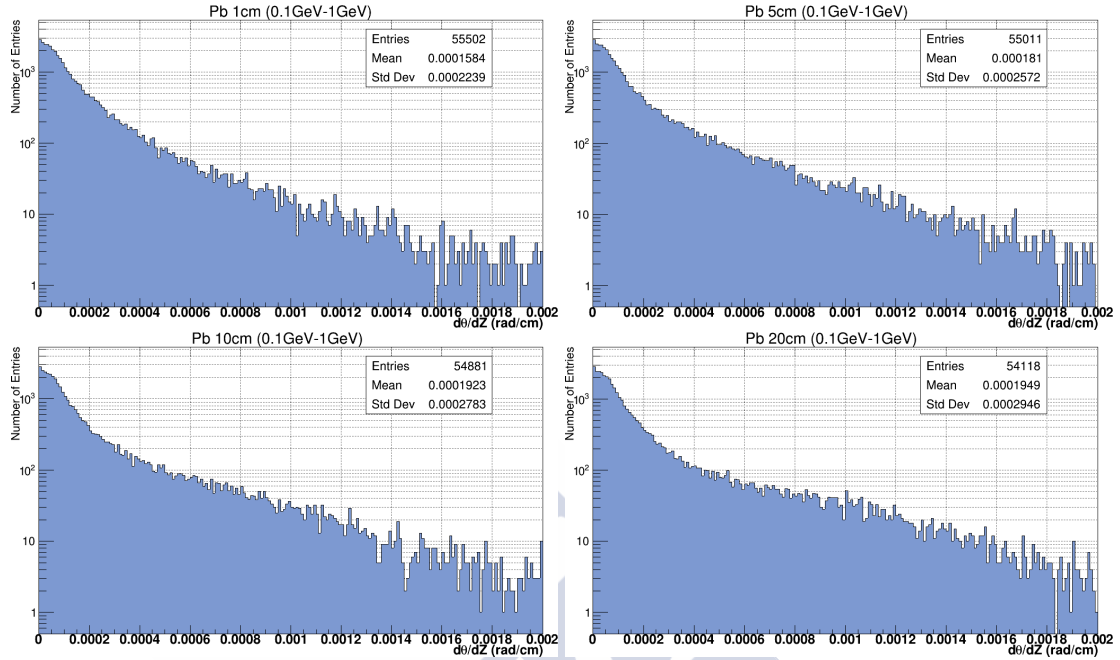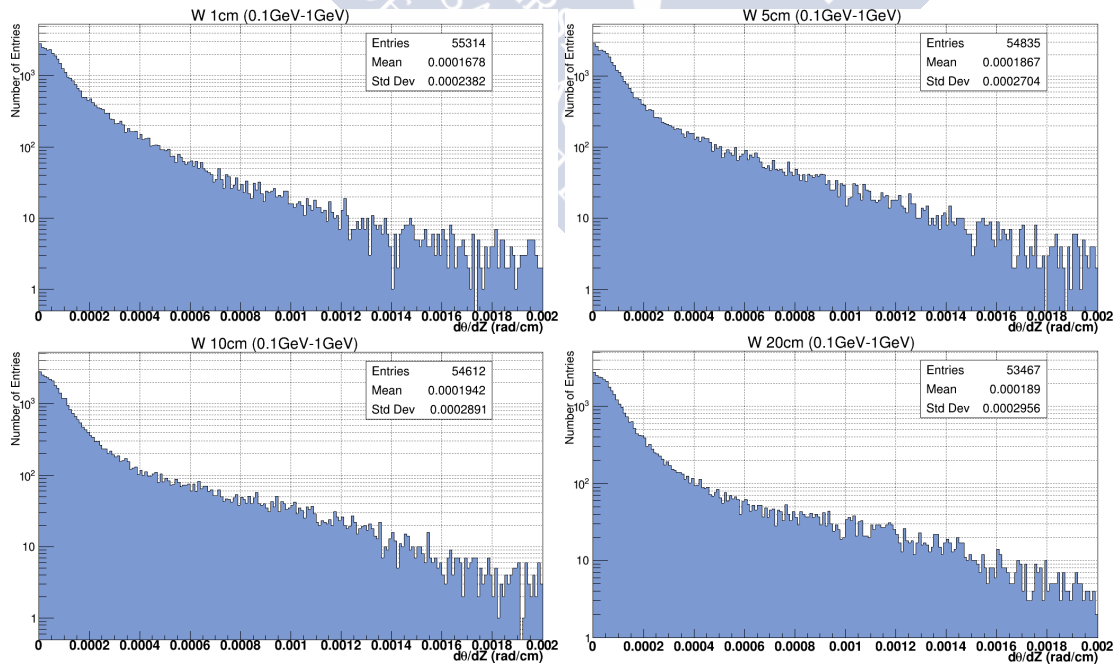
Integrated $d\theta/dz$ for intermediate energies (from 1 GeV to 3 GeV)

Fe

Pb

W
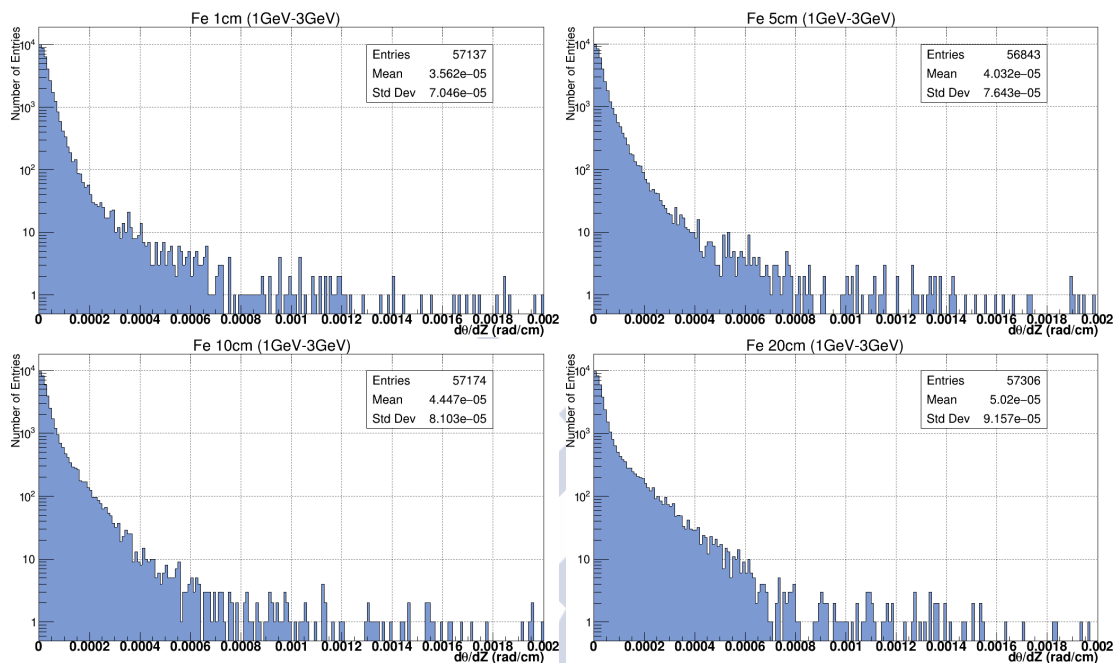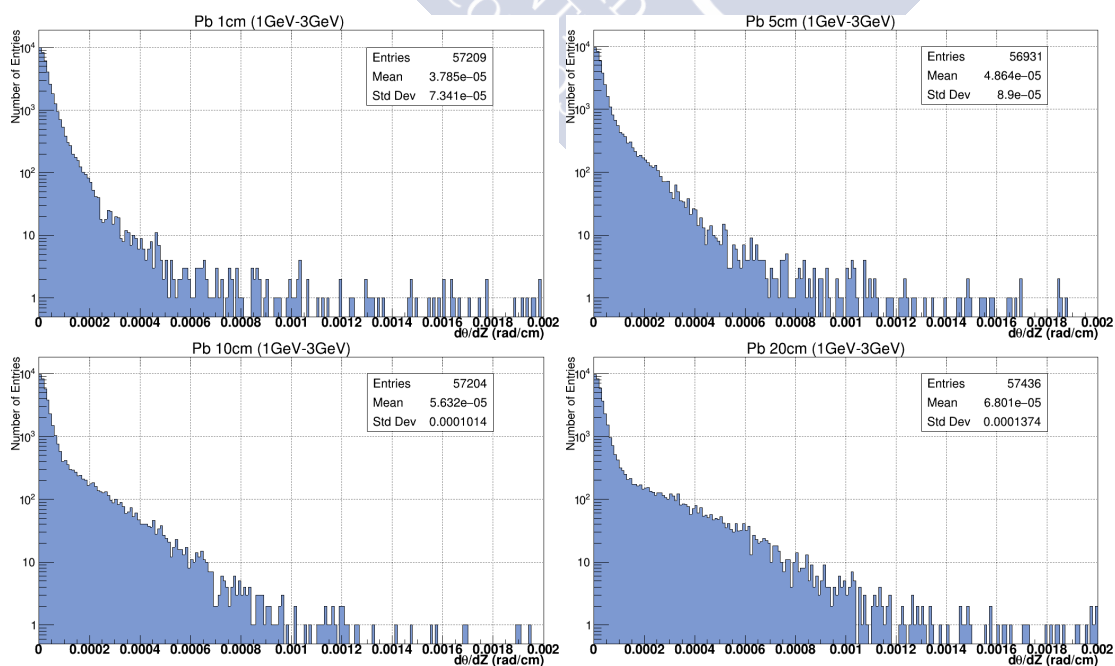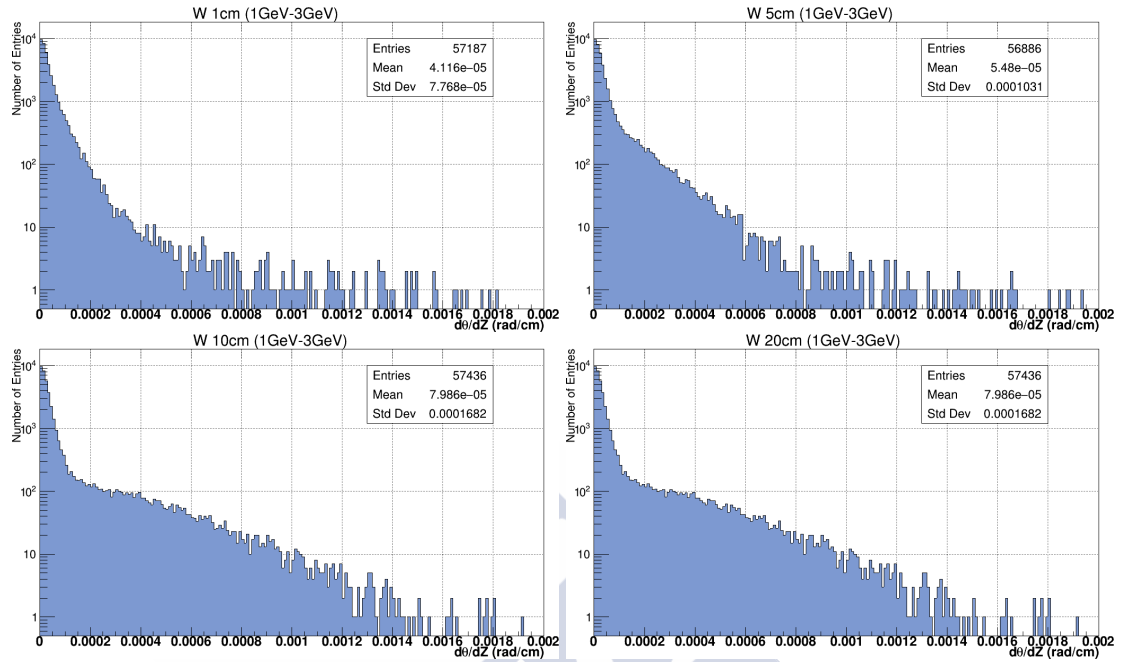
Integrated $d\theta/dz$ for high energies (from 3 GeV to 6 GeV)

Fe
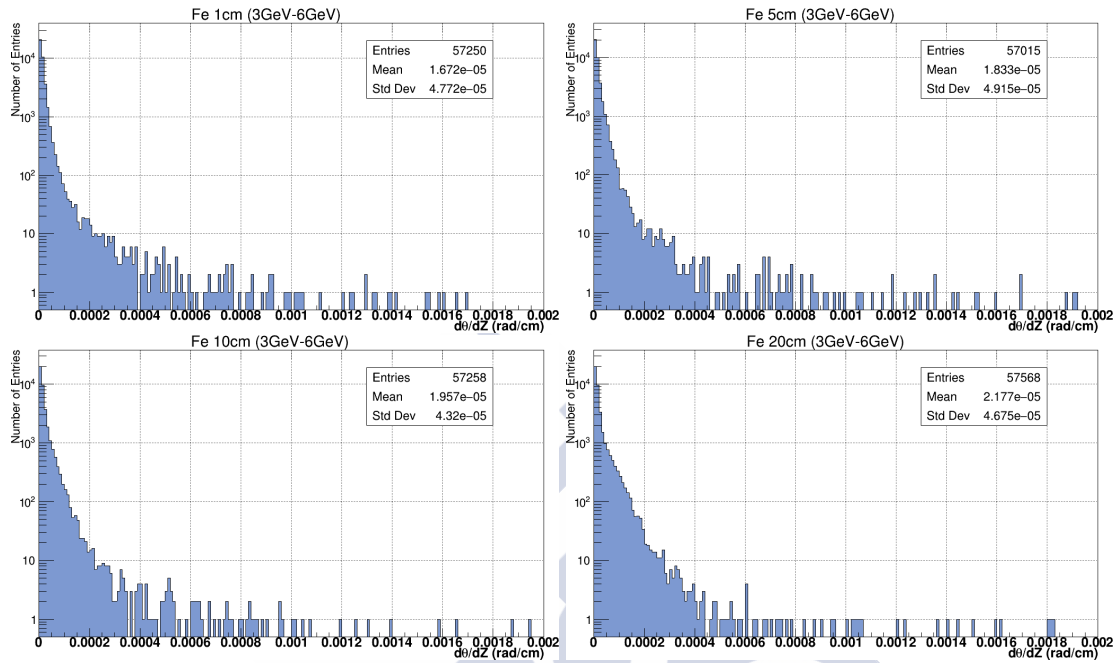
Pb
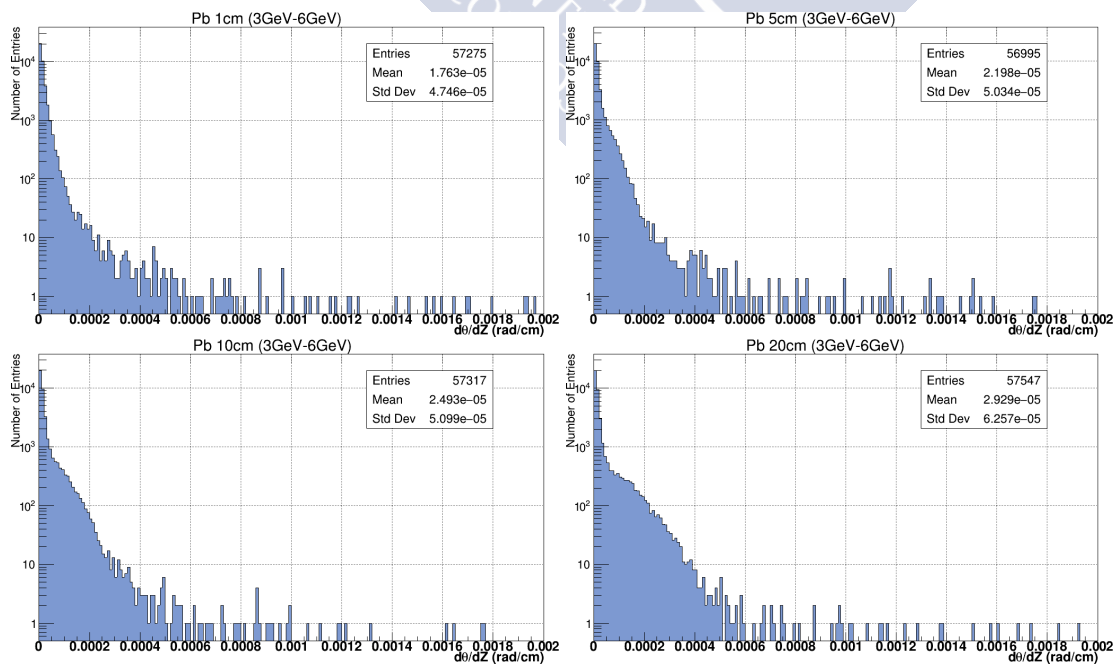
W
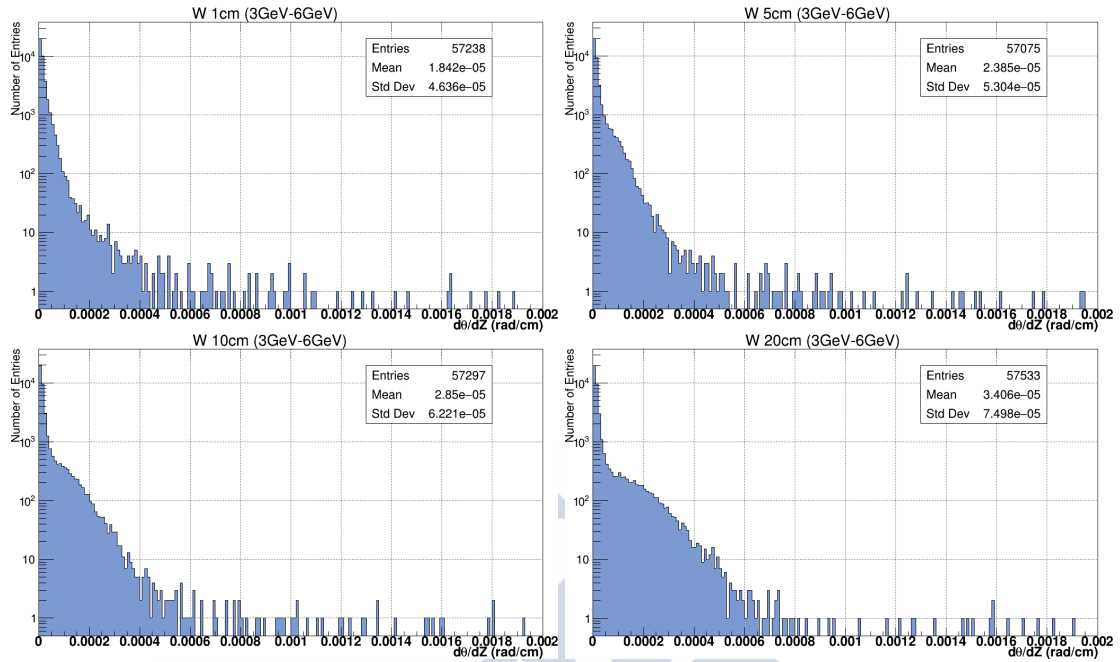
B.2   Study of $d\Omega$

The $d\Omega$ observable is defined as the relative angle between the scattered trajectories obtained with POCA and TimTrack. This angle is calculated using the scalar product of both vectors:

$$d\Omega = \arccos\left(\vec{v} \cdot \vec{w}\right)$$

where $\vec{v}$ is the direction vector of the TimTrack and $\vec{w}$ is the direction vector obtained with POCA.

In the first part of the section, the observable is represented as a function of the incident angle θ. The second part shows the individual histograms of the observable for each material, thickness, and muon energy.

$d\Omega$ vs θ low energies (from 0.1 GeV to 1 GeV)

Fe

Pb

W
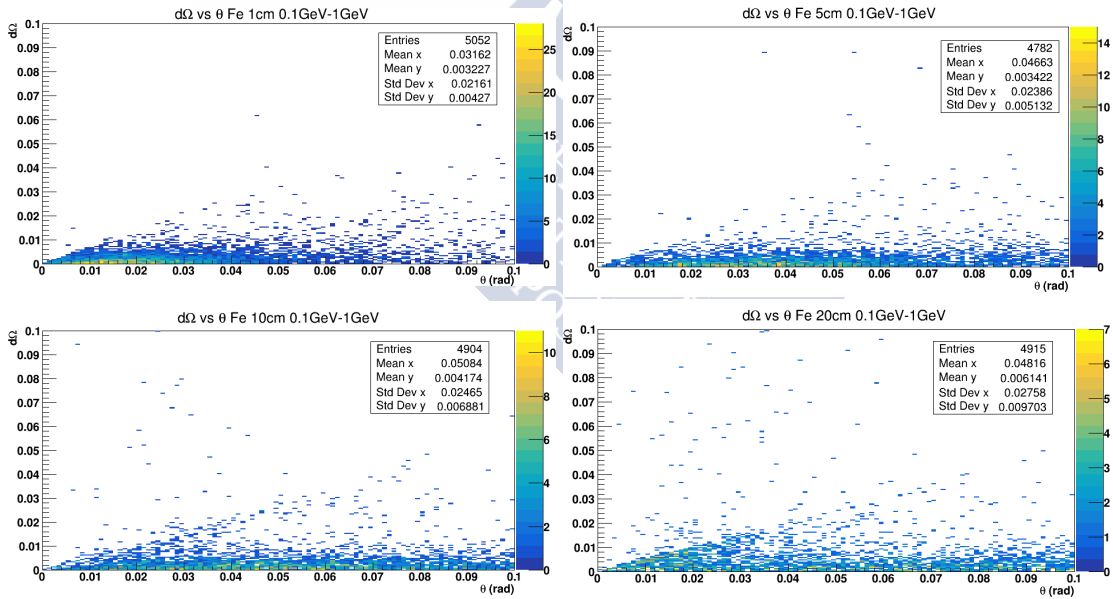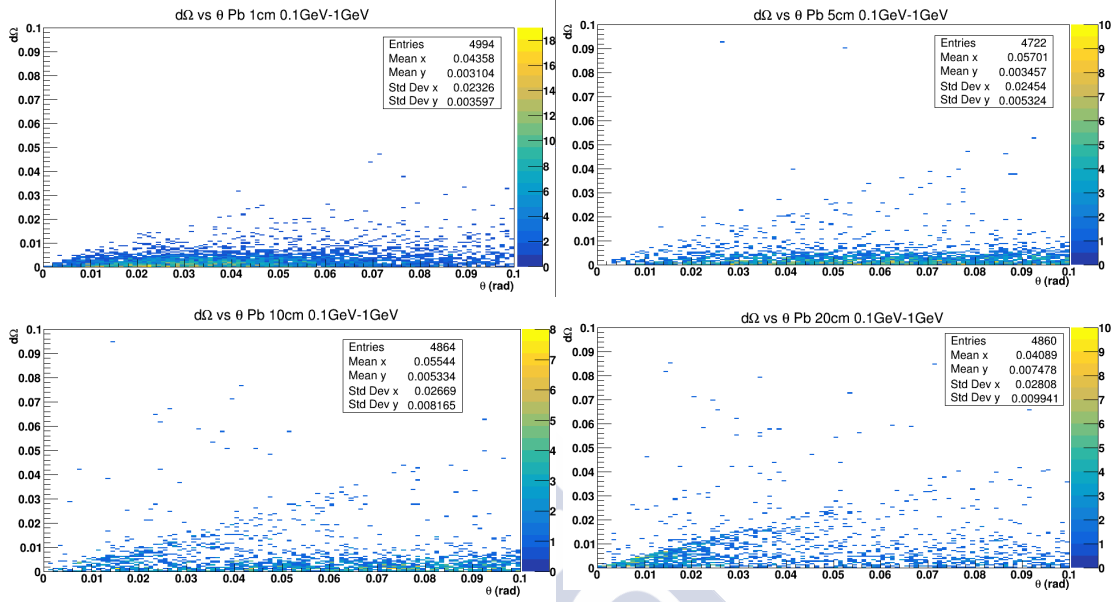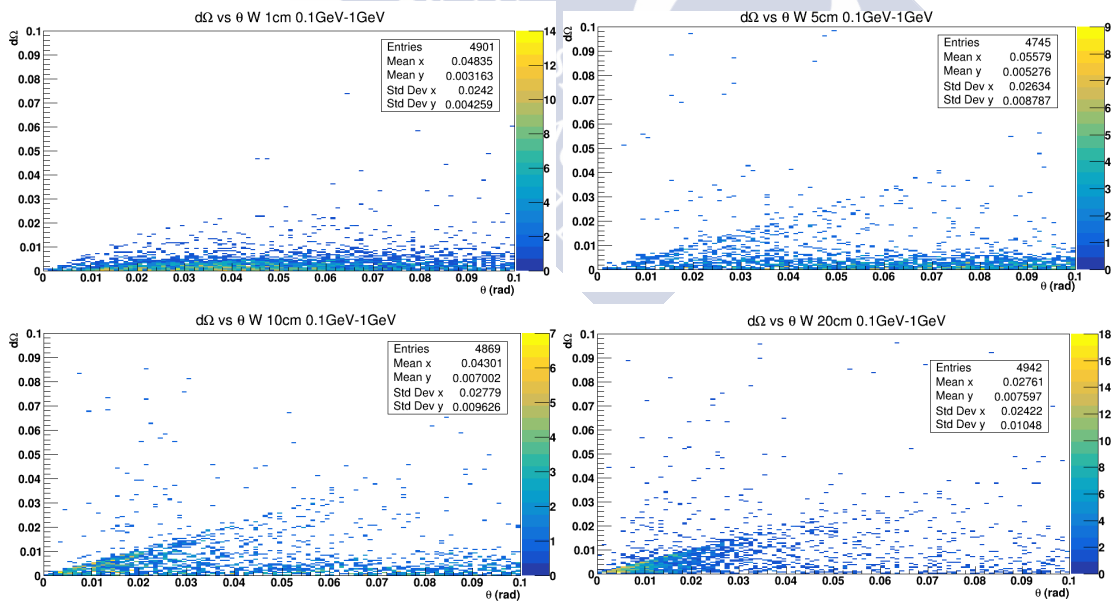
$d\Omega$ vs θ intermediate energies (from 1 GeV to 3 GeV)

Fe

Pb

W
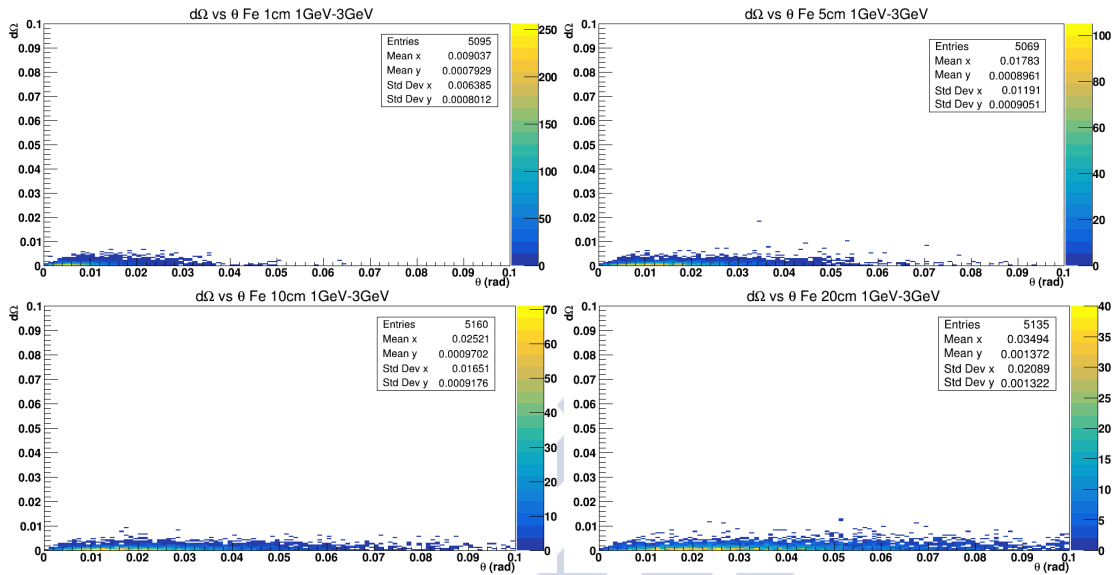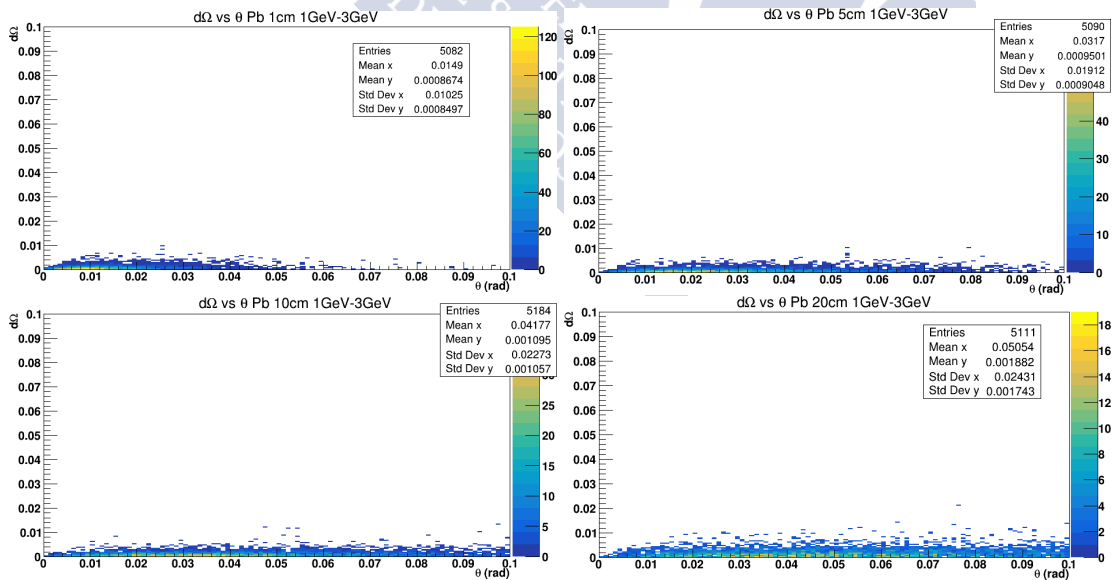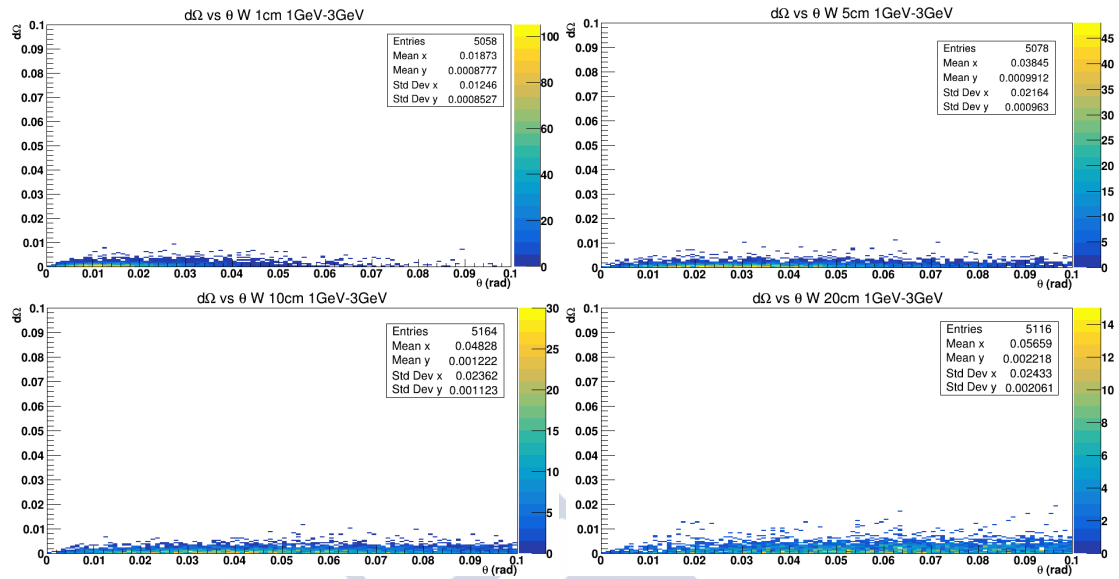
$d\Omega$ vs θ high energies (from 3 GeV to 6 GeV)

Fe
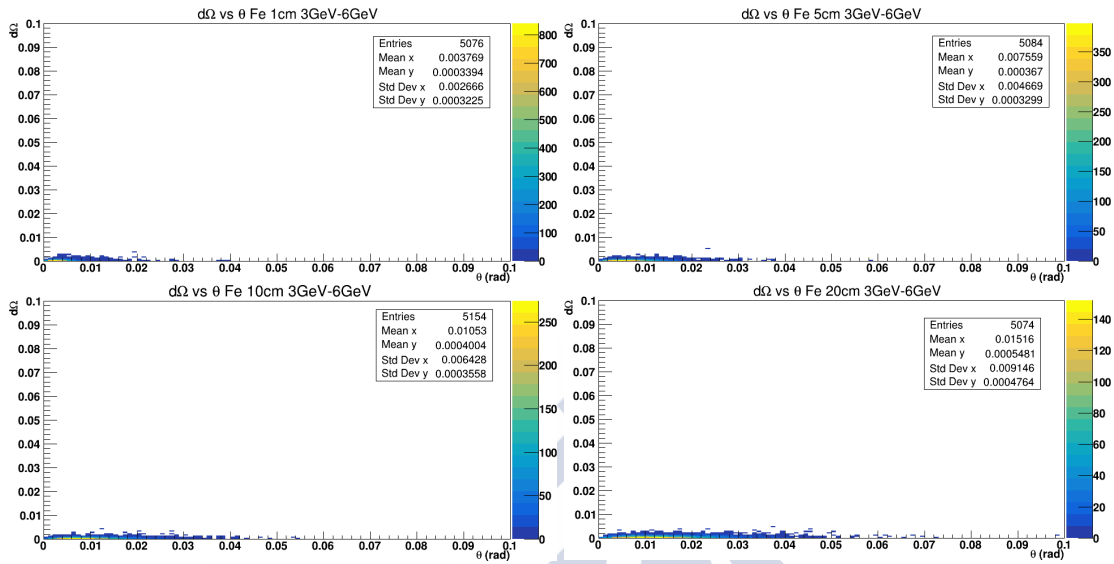
Pb
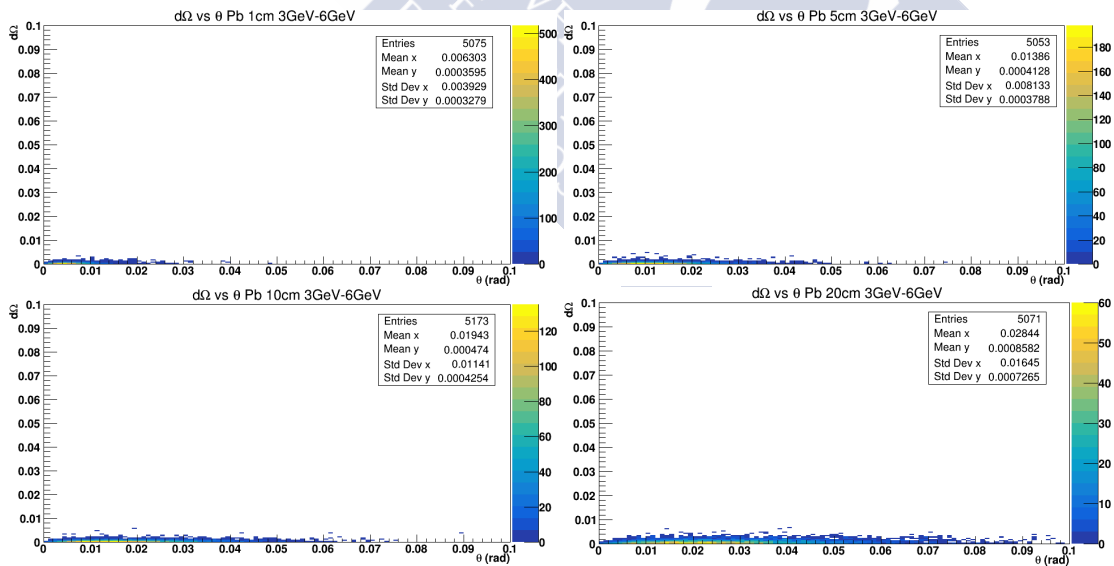
W
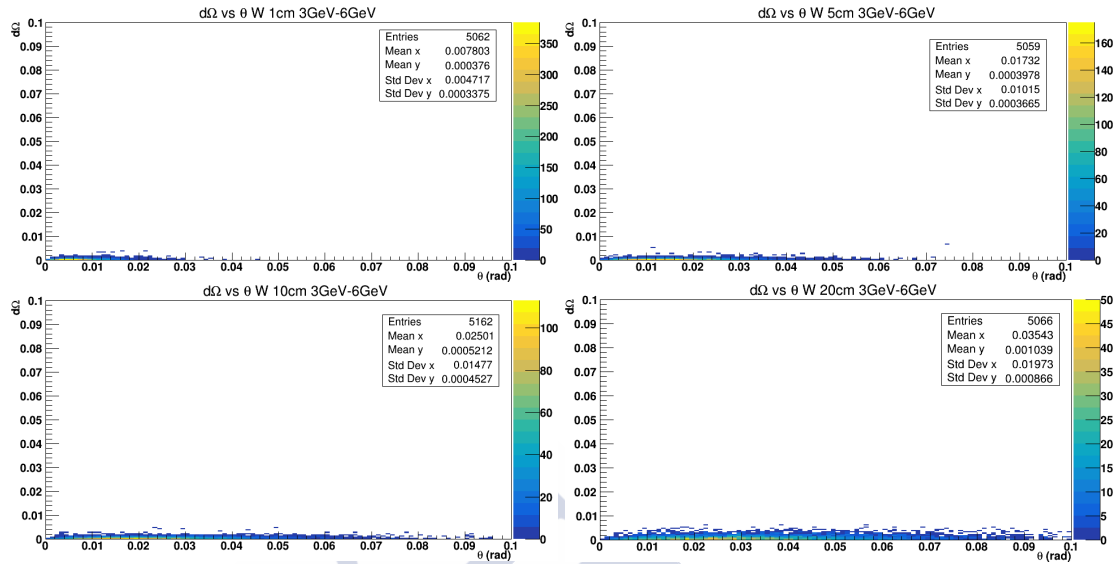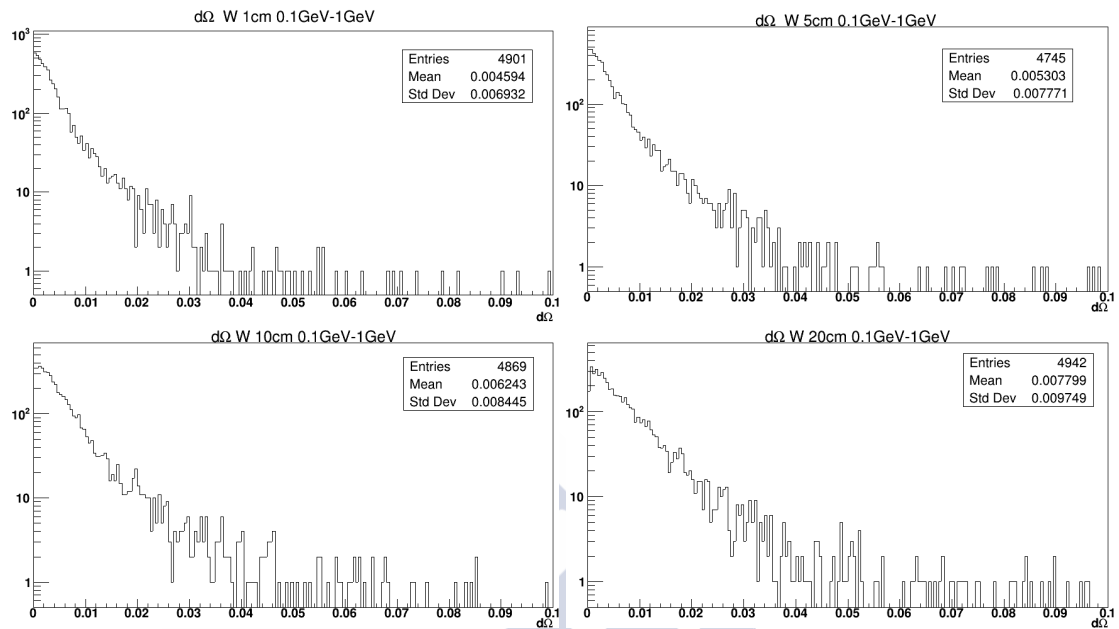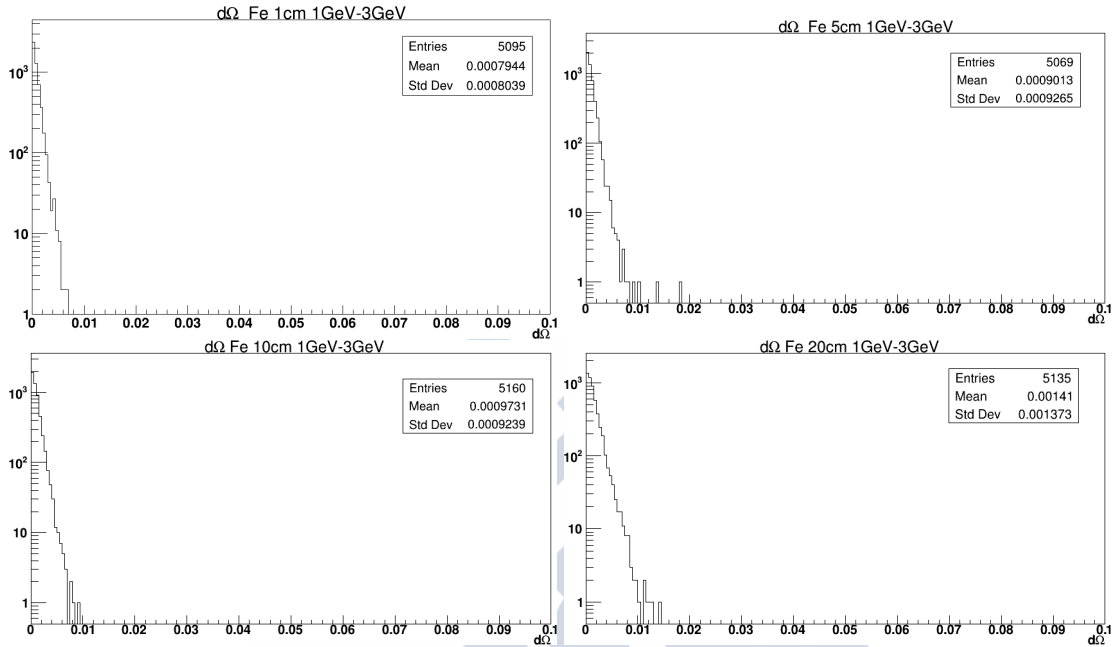
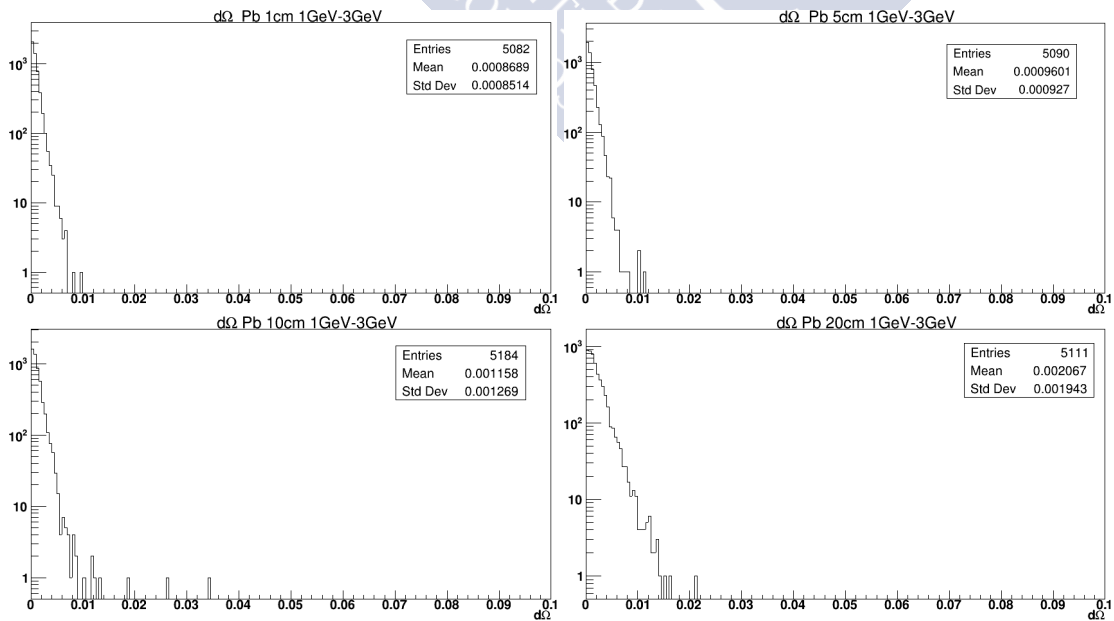# $d\Omega$ low energies (from 0.1 GeV to 1 GeV)

Fe
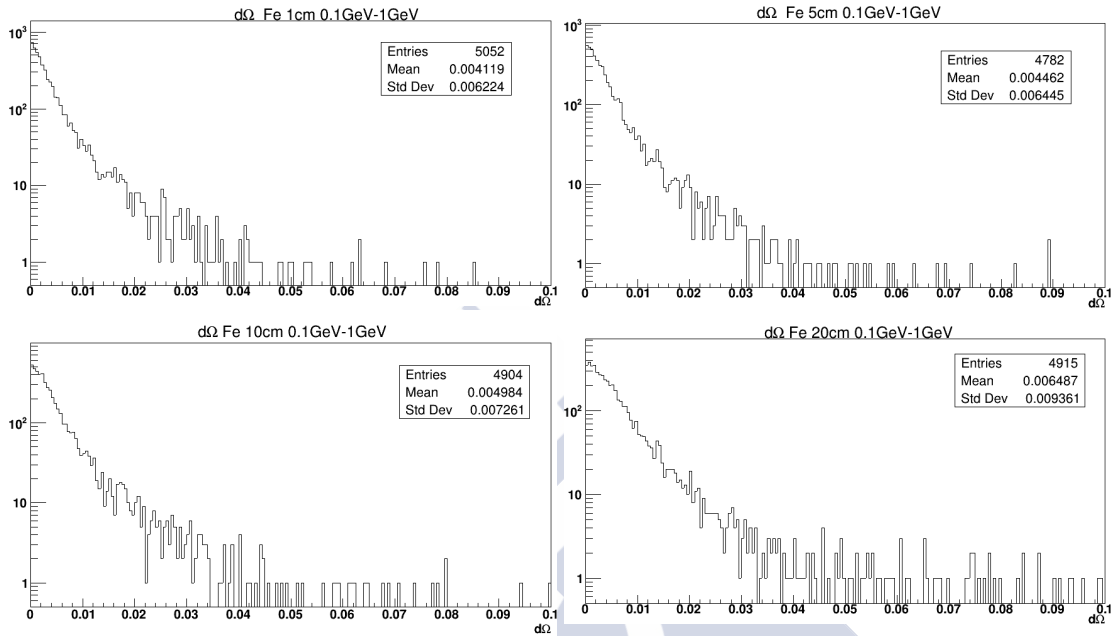
Pb
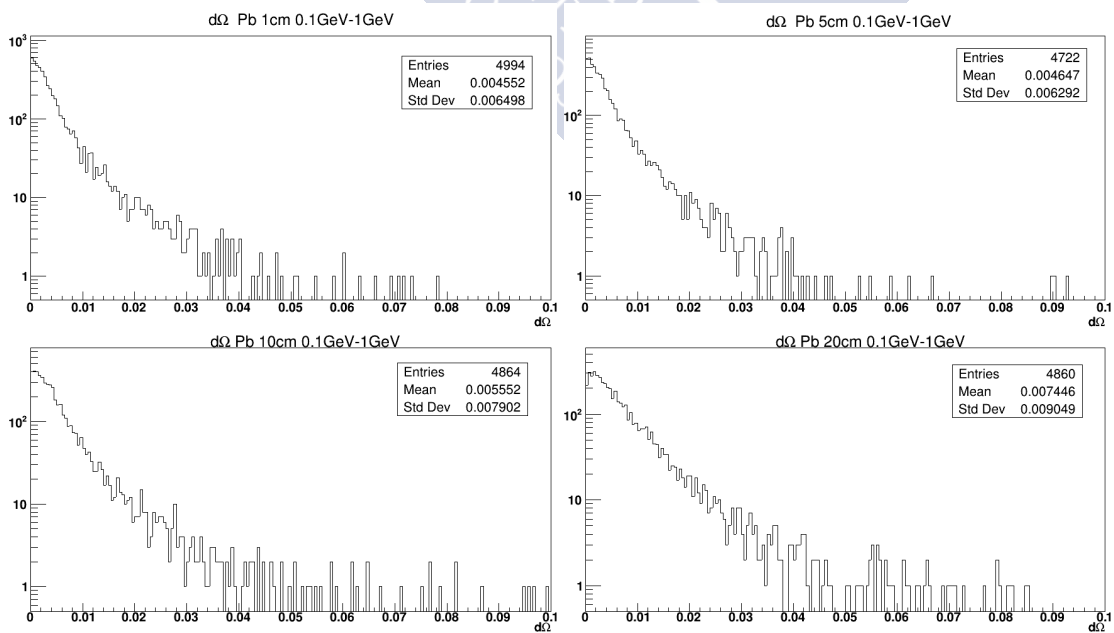
W

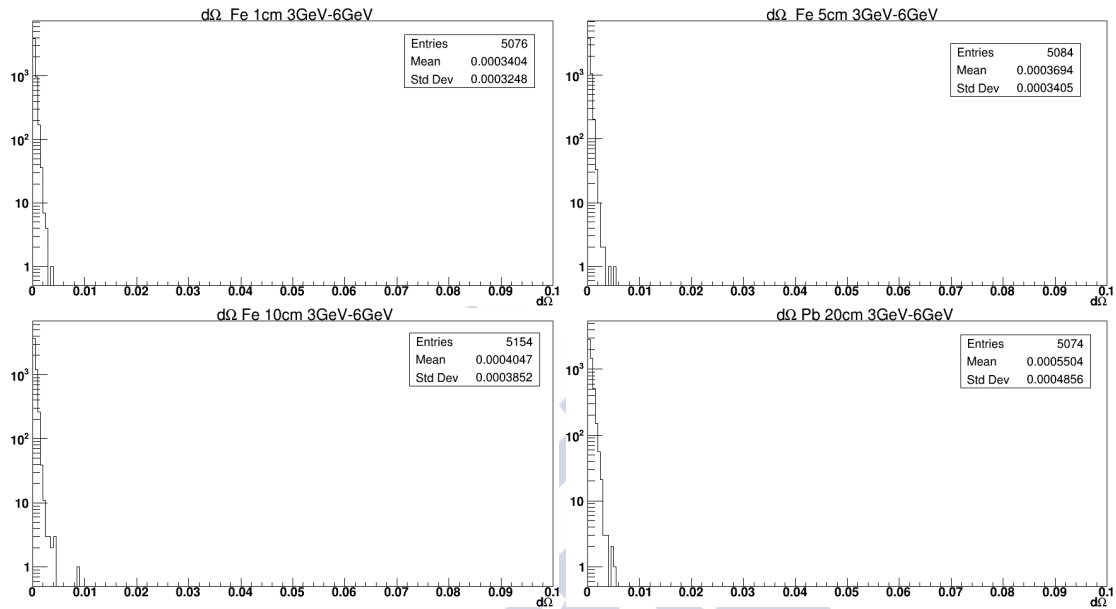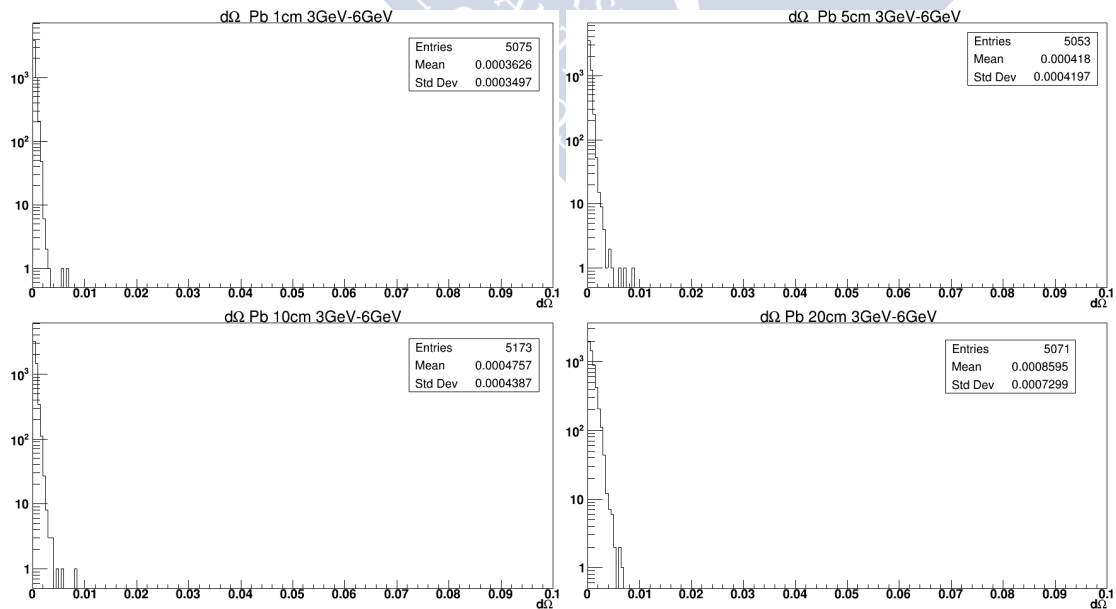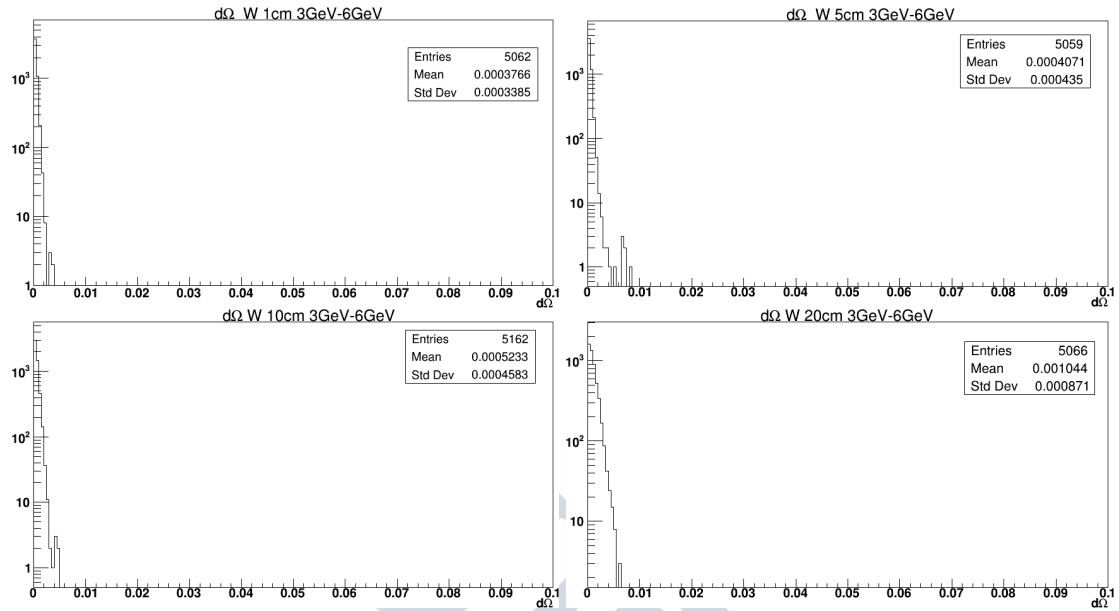# $d\Omega$ intermediate energies (from 1 GeV to 3 GeV)

Fe

Pb

W

$d\Omega$ high energies (from 3 GeV to 6 GeV)

Fe

Pb

W

dΩ W 1cm 3GeV-6GeV

Entries	5062
Mean	0.0003766
Std Dev	0.0003385

dΩ W 5cm 3GeV-6GeV

Entries	5059
Mean	0.0004071
Std Dev	0.000435

dΩ W 10cm 3GeV-6GeV

Entries	5162
Mean	0.0005233
Std Dev	0.0004583

dΩ W 20cm 3GeV-6GeV

Entries	5066
Mean	0.001044
Std Dev	0.000871
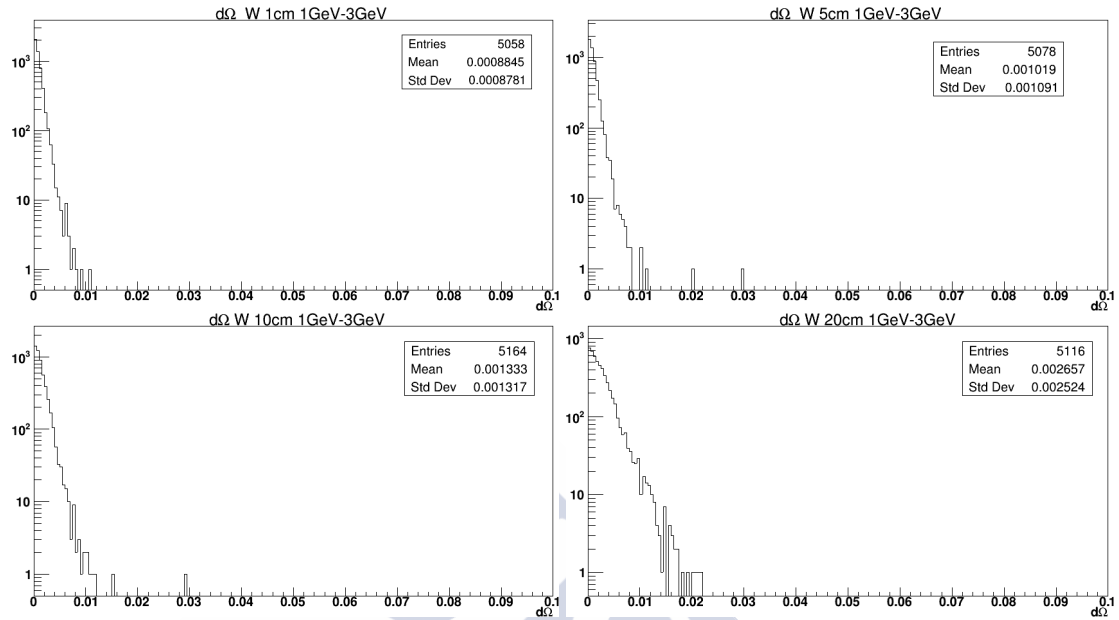
B.3 Study of the background of $d\theta/dz$ in Fe, Pb, and W for vertical muons with low, intermediate, and high energies

In this section we perform a study of the background of the relative polar deviation observable. Each histogram shows the value of $d\theta/dz$ as a function of x_{ss}.

In the simulations we used four bricks of the material indicated in the title of the histogram. The bricks are, from left to right, of 1 cm , 5 cm, 10 cm, and 20 cm thickness. Muons of low, intermediate, and high energy were used.