Corso di laurea in Fisica

Antonio Condorelli

A new trigger system for the radio detection

elaborato finale

Relatori:
Prof.ssa Rossella Caruso
Prof. Charles Timmermans

---

academic year 2015/2016

Indice

1 Introduction

In contrast to many other fields of physics, cosmic radiation caused a great excitement from the very beginning. This new phenomenon very soon involved new effects, new particles were discovered because the colliders weren't enough energetic, and astrophysicists were attracted to understand its origin. Without the discovery of cosmic radiation,the rapid scientific development in the experimental field of nuclear and particle physics and to a lesser extent in space and solar physics would not have occurred.

The discovery started at the beginning of the 20th century with the study of the electric conductivity of a gas: although a dielectric-like behavior was expected,there existed a residual conductivity which was not understood. Among the explanations for this conductivity which included radioactivity from the ground and spontaneous ionization of the gas, there was the possibility that it could be due to a new phenomenon: radioactivity from above, from the Universe. The altitude dependence of the residual conductivity was studied using a *balloon-borne ionization chamber* with the scientist accompanying the instrument since telemetry was not yet feasible. After some preliminary misinterpretation of the results **Victor Hess** (1883-1964) finally showed in 1912, in Austria, that beyond an altitude of about 1000 m the conductivity increased and he made the revolutionary statement that a new penetrating radiation from outside the Earth must be responsible. Since the lack of differences in intensity between day and night balloon flights was found he concluded that the Sun was not the source of this new radiation.

Nowadays the term *cosmic rays* may be defined as all radiation consisting of relativistic particles striking on the Earth's atmosphere from outer space or, more narrowly, including only charged particles. The qualification *relativistic* (or high energy) means that we do not consider cosmic rays below a few GeV in energy, that may be produced or are influenced by the Sun and its wind. The cosmic ray spectrum, a nearly featureless power-law extending over eleven decades in energy up to a few 10^{20} eV eV, is shown in Fig. 1.1.

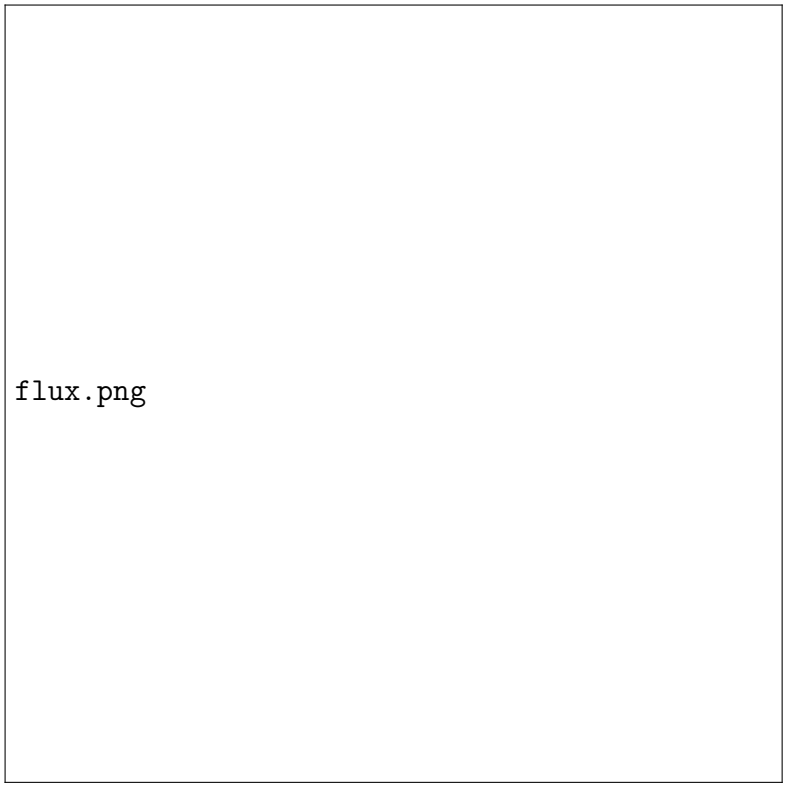

Figura 1: Cosmic rays spectrum

1.1 Ultra high energy cosmic rays

Even though their existence has been known from more than a century, the nature and origin of cosmic rays of the highest energies remains elusive. The open questions of high-energy cosmic rays can be resumed in three fundamental questions:

- *What are the sources of high energy cosmic rays?*
- *By which processes are they accelerated?*
- *What are the laws governing their interactions in Earth's atmosphere?*

The most important obstacle in conclusively answering these questions is the presence of (inter)galactic magnetic fields, which influence the trajectory of the particles traveling towards Earth. Because of their deflection in these fields, it

is difficult, if not impossible, to point back measured cosmic rays towards their sources.

Ultra-high-energy cosmic rays present an enigma to the science community. There are only few types of astronomical sources that would be able to accelerate particles to these energies, but at the same time there are limitations on the distance charged particles of these energies can travel through the universe. Yet, up to now it has been impossible to identify actual sources, and various theories exist about the mechanism to accelerate these particles. By studying the cosmic rays properties accessible to us on Earth, their energy, composition and arrival direction, we can attempt to figure out the fundamental and interconnected questions about the sources, acceleration and propagation of ultra-high-energy cosmic rays.

Now we would focus on what happens when a cosmic ray arrives and interacts in the highest part of the atmosphere: an **Extensive Air Shower** .

1.2 Extensive Air Shower

Only few of the many proposed solutions to the three fundamental questions of the previous chapter have been verified experimentally. These primary particles could direct us towards the sources of cosmic rays, which in turn will teach us about their acceleration and propagation. The debris from the collision of the cosmic rays with an atmospheric molecules that arrives at the earth's surface is the tail of a cascade of interactions and decays: an extensive air shower.

(geisser) An air shower is a cascade of particles generated by the interaction of a single high energy primary cosmic ray nucleus or photon near the top of the atmosphere. The number of particles at first multiplies, then reaches a maximum and attenuates as more and more particles fall below the threshold for further particle production.

A cosmic ray induced air shower has three components, *electromagnetic, muonic and hadronic.*

The initial collision creates a host of hadrons, mostly pions but also kaons and baryons, and depending on the composition of the primary cosmic ray, secondary nuclear fragments. This part of the shower is called *hadronic component.* Neutral pions have a life time of just $10^{-16}s$, and will therefore almost in an extensive air

shower decay into two gamma rays:

$$\pi^0 \longrightarrow \gamma\gamma \tag{1}$$

These photons initiate sub-cascades of electrons, positrons and photons trough pair production, bremsstrahlung, ionization and Compton scattering. This *electromagnetic component* of the air shower will continue to grow until the energy of the electrons falls below the critical value necessary for radiative processes, at which point they will mostly ionize the surrounding air and the component starts to die out. The charged pions have a longer decay time than the neutral pions, and will either interact within the atmosphere to create more secondary particles, or decay into muons:

$$\pi^{\pm} \longrightarrow \mu^{\pm} + \nu \tag{2}$$

The muons have long life times and small cross-section, and will therefore travel to the ground almost unimpeded. They constitute the *muonic component* of the air shower.

The shower consists of a core of high energy hadrons that continually feeds the electromagnetic part of the shower, primarily by photons from decay of neutral pions and eta particles, as it is shown in the picture 2. Each high energy photon generates an electromagnetic sub-shower of alternate pair production and Bremsstrahlung starting at its point of injection. Nucleons and other high energy hadrons contribute further to the hadronic cascade. Lower energy charged pions and kaons decay to feed muonic component. So at each hadronic interaction, slightly more than a third of the energy goes into the electromagnetic component. Since most hadrons re-interact, most of the primary energy eventually finds its way into electromagnetic component. In addition, because of the rapid multiplication of electromagnetic cascades, electrons and positrons are the most numerous particles in cosmic rays air showers.

At high energy the flux of cosmic rays is so small that the showers are the only way to observe them. Already Pierre Auger and collaborators discovered that the high density area of the showers is of the order of $10^4 m^2$. The method of observation is to put many counters at some distance and look for coincidental hits; if only 1 % of the total area is instrumented, the chance of detection increase

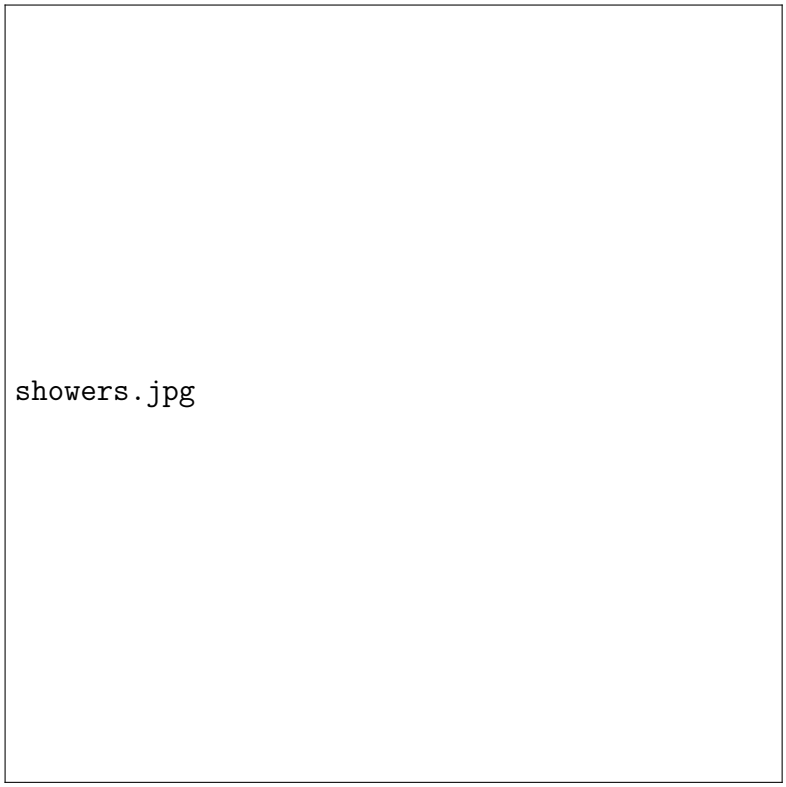

Figura 2: Extensive air shower

by many orders of magnitude because of $10^4 m^2$ effective area of the high energy cosmic ray particle.

The interpretation of these events is, however, difficult. The shower array detects particles that cross almost simultaneously the observation level. There is no information on the energy and type of primary cosmic ray; these have to be derived from the shower properties.

1.3 Pierre Auger results

I would like to present some results from the last years of operation of the Pierre Auger Observatory, the most important experiment about cosmic rays. The two techniques thanks to which it was able to answer to some of the questions explained in the last chapter were two: **SD** and **FD**.

The ground array, named Surface Detector (SD), consists of 1660 stations -

water-Cherenkov tanks and their associated electronics arranged in triangular cells, where each station is 1.5 km apart from another. The Cherenkov light generated in water is collected by 3 photomultipliers and the signal is read out by 40 MHz FADC's, which give a 25 ns time scale for further shower reconstruction. The signal unit is called Vertical Muon Equivalent (VEM), the signal generated by a muon crossing the water tank vertically, at its center. The Surface Detector has a duty cycle near to 100%.

Figura 3: Surface detector (tank) in P.A. Observatory

Fluorescence Detector (FD) is an ensemble of 27 optical telescopes, grouped in 5 buildings, that can observe the longitudinal development of the shower in the atmosphere in a stereoscopic view. The shower profile is derived by detecting the fluorescence light emitted by excited nitrogen molecules - this is an established technique, used in previous experiments as Fly's Eye and HiRes and now in Telescope Array (TA). In the Auger Observatory, the telescopes record light profiles from showers that may be some kilometers away, over the ground array. This light

suffers attenuation along the path from the production point to the telescopes, thus the detected profile must be corrected to reproduce the fluorescence production at the origin. To estimate the attenuation, which is due to molecules, aerosols and clouds, regular measurements of the atmospheric conditions are performed: UV laser shooting, radiosonde launching, optical observations and cloud measurements. They are incorporated into the event reconstruction software, so that the reconstructed fluorescence profile gives an almost calorimetric measurement of the energy of the primary particle. The duty cycle of the Fluorescence Detector is 10 - 15 % . The fluorescence and surface detector techniques are combined, in the Observatory, to give detailed information about air showers.

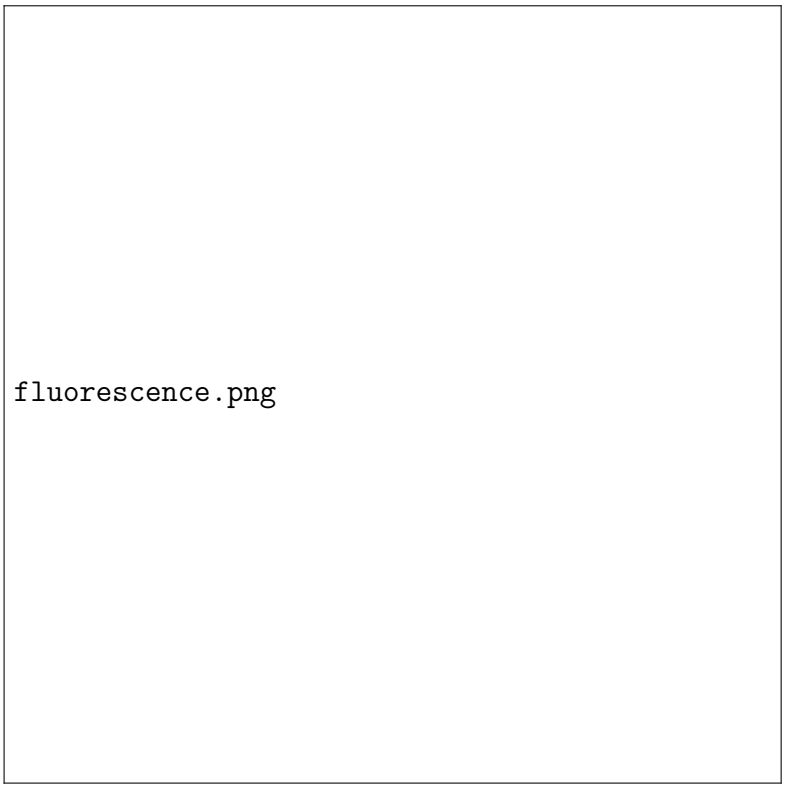

Figura 4: Fluorescence detector in P.A. observatory.

An important piece in the event reconstruction is the lateral distribution of the signal, as recorded in the Cherenkov stations. A lateral distribution function, based on shower simulations and previous data from the Auger Observatory, is fitted to the experimental curve, in order to obtain the value of signal at 1000

m from shower core. A universal calibration curve using this information and those from fluorescence detector, is used to obtain the primary energy, within a systematic error of 22%, whereas the statistical error is 15 %. The calibration curve is then used for all showers, also for non-hybrid detection. The figure below shows a high-precision energy spectrum, at energies above $10^{18} eV$.

Figura 5: A high precision spectrum obtained by P.A.

The atmospheric profile of an extensive air shower, the number of particles as a function of depth, carries important information about the mechanisms of particle production and absorption. The depth at which the shower reaches its maximum, called X_{max} in the literature gives important clue about the mass of the primary particle. The change of $< X_{max} >$ per decade of energy, the so-called Elongation Rate and its shower-to-shower fluctuations, RMS (X_{max}) are sensitive to changes in mass composition with energy. These variables have different behaviors, if the primary is a proton or an iron nucleus, as predicted by different hadronic interaction models. They found a tendency in the composition: if the models

describe correctly the interactions, when the energy gets higher, the composition departs from that expected for protons and tends to that expected for iron primary.

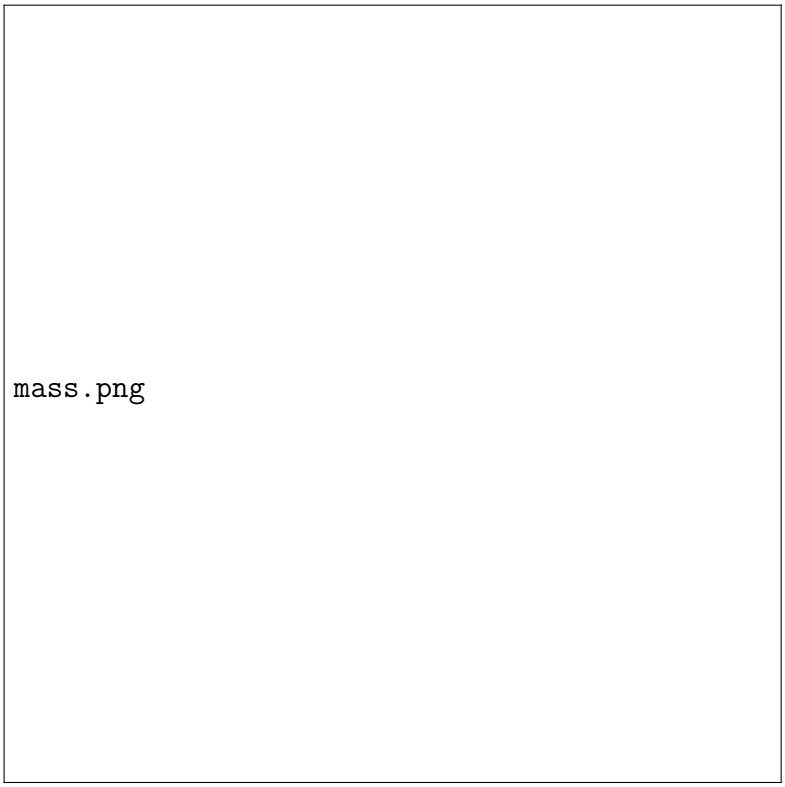

Figura 6: Results about mass composition of UHECR obtained by Pierre Auger Observatory.

2 Radio emission and detection

Together with the Surface and Fluorescence detectors, which are inherited by the Pierre Auger Observatory from the other important experiments in the past, in the last years a new way to get information about properties of cosmic rays has been developed by German and Dutch collaborations.

Charge creation and acceleration in an extensive air shower induces radio emission, a phenomenon which follows directly from their description using classical electrodynamics. In the 1960s, it was measured for the first time and at around the same time the principal processes by which the radio emission is produced were already identified. In this chapter were described in a first moment about the two components of the radio emission caused by an extensive air shower, and after the detectors of AERA thanks to which we could able to read the radio signals.

2.1 Geo-magnetic Mechanism

(radio emission of air shower libro blu di fabrizia)

The primary process by which radio emission is produced is the *geomagnetic* mechanism, or current excess, described by Kahn an Lerche. The local geomagnetic field accelerates electrons and positrons in the shower front in opposite direction. This leads to a net-drift of the electrons and positrons, which can be interpreted as a current. This current varies with the amount of charge present in the air shower. This current varies with the amount of charge present in the air shower; this leads to the effect that the current increases until the shower reaches the shower maximum and starts to decrease from there. This effect is either being to as the *geomagnetic effect* or *time varying transverse currents*.

While traveling through the geomagnetic field in the atmosphere, trajectories of particles, with charge q, are bent under the influence of Lorentz force

$$\mathbf{F}_1 = q(\mathbf{v}_s \times \mathbf{B}) \tag{3}$$

This force is perpendicular to the geomagnetic field $\mathbf{B}$, and the direction of the

Figura 7: Geomagnetic mechanism

propagation of the particles $\mathbf{v}_s$. Therefore, is strength is equal to:

$$F_1 = qv_s sin(\alpha) \tag{4}$$

with α the angle between $\mathbf{B}$ and $\mathbf{v}_s$. The electrons and positrons gain a velocity, v_t, transverse to v_s. The angular deflection ψ, due to $\mathbf{F}_1$, of the trajectory of an electron or positron with mass m_e is approximated by

$$\psi = \frac{L_f}{R_B} = \frac{qBL_f sin\alpha}{m_e \beta_s c} \tag{5}$$

in which R_B is the radius of curvature of the trajectory of an electron in magnetic field B with an energy $\epsilon = \gamma mc^2$. The length L_f gives the mean free path of an electron passing through air, which should depend on air density ρ_{air} and the γ the factor of the electrons. The average drift velocity of the electrons is approximated to be

$$\langle v_d \rangle = \frac{c\psi}{2} = \frac{qBL_f(\gamma, \rho_{air})sin\alpha}{2m_e\beta_s c} \tag{6}$$

in which the factor 2 results from averaging over the whole trajectory. The direction of the drift velocity depends on the sign of q. For this mechanism of emission the electric field orientation does not depend on the location of the observer, but only on the direction in which the Extensive Air Shower propagates.

In general, the amplitude of the electric field scales with the number of electrons and positrons $N_{e^\pm}$ as the emission is coherent. The radiated energy then scales with $N_{e\pm}^2$ which is directly correlated to the energy of the primary particle by equation

$$N_{max} = \frac{E_0}{E_c} = 2^{n_c} \tag{7}$$

In addition the strength scales with the sine of the angle between the shower axis and the geomagnetic field (geomagnetic angle) and the relation of the amplitude and radiated energy needs to be corrected accordingly.

Furthermore the geomagnetic-emission mechanism induces an electric field E_G which is pointing along the direction of $-\vec{v}\times\vec{B}$) where v is a vector in the direction of the shower. Thus, for this emission mechanism, the relative contribution of the registered signals in each of the two arms does not change as a function of the position of the RDS.

2.2 Charge Excess (Askarian) Mechanism

(Cosmic radiation, libro nero)

A couple of years after the development of Heitler model, which describes in a simple but effective way the development of the electromagnetic and hadronic showers, Askarian proposed the emission of radiation by electromagnetic cascades developing in a medium. The origin of the radiation with frequencies in the MHz-regime is explained with an excess of negative charges in the air shower front and called *charge excess*. As the shower develops in the medium, electrons are dragged from molecules in the medium along with all the shower particles leaving positively charged ions behind. This results in a charge separation along the shower axis.

Figura 8: Askarian mechanism

During the shower development, the number of particles first increases and after decreases and both effects together can be seen as a changing dipole moment. Furthermore, positron in the shower front can annihilate with electrons in the medium amplifying the charge separation. The radiation emitted by the dipole is coherent as long as the wavelength is larger than the thickness of the shower front. The polarization of the radiation is determined by the dipole moment and parallel to the shower axis.

The electric field measured at ground level is a superposition of the two electric field emitted by the different emission mechanism and the direction of the electric field vector depends on the strength of the individual components. The magnitude of the induced electric current as well as the induced charge excess is roughly proportional to the number of particles in the shower and thus changing in time. The latter results in the emission of coherent radio waves at sufficiently large wavelengths. The shower front, where both the induced transverse current and

the charge excess reside, moves through the air with nearly the velocity of light.

2.3 Background for radio signal

(finger on the pulse of cosmic rays)

We have to account for the fact that the refractive index n of the atmosphere is larger than one to understand the measured radio signal. The charged particles of the E.A.S. travel with a speed greater than the speed of light in the medium and polarize it. When the molecules turn back to their ground state, they emit Cherenkov radiation. The generated radiation of all the emission mechanisms travel with the speed $v = c/n$, whereas the shower front travel with a higher speed, close to c. The refractive index changes with the altitude and depends on temperature, pressure and humidity. Neglecting the latter effects, the height dependence follows roughly an exponential function with at sea-level $n(0 \text{ km}) = 1.000325$ and n (10 km) = 1.000096. Nevertheless The polarization of the emitted radiation differs for current-induced and charge-induced radiation, but its direction for each of these individual components does not depend on the Cherenkov emission, this is why the Cherenkov emission don't disturb our acquisition.

All the background for this physics comes from two components: the noise and the RFI. The first contribute has no distinct characteristics besides a frequency dependent amplitude; Both can be natural or man-made and there is a smooth transition from one to the other. The anthropogenic emission which increases the background further varies over time, depending on its origin. This contribution is really small in AERA due to the location in the Argentinean Pampas and it is obtained by the superpositiom of radio emission from all kinds of technical equipment and machines that cannot be resolved as individual sources.

It's very important for the radio measurements and especially when we decide to set the trigger level, the radio emission the overall noise, set by radio emission of the Galactic plane; this contribute varies in time and in particular it has a maximum when the galactic center is in the middle of the field of view of AERA. It could be used for calibration purpose.

The other important component of the radio background is called **RFI** (Radio Frequency Interference), which consists of a superimposition of the contributions

analyzed before, and we can subdivide it into narrowband and broadband RFI.

Narrowband RFI is emitted continously by communication devices that operate in AERA range. Their intensity varies in time and the sources turn on and off. A lot of sources of narrowband noise negatively influence the data recorded by the radio detector array. Narrowband RFI can be identified as high-powered sources in a small spectral region.

Man-made broadband RFI is the most challenging background for a radio detection, in particular for a radio araay operating in self-triggering mode. In fact these pulses mimic cosmic ray induced signal and in radio self-trigger mode is around 200 Hz, while the coincidence rate with SD is only about one per day.

3 The AERA detector

3.1 Pierre Auger Observatory

(Salvato nella sitografia) Such a we say in the previous chapter, The Pierre Auger Observatory has been conceived to measure the flux, arrival direction distribution and mass composition of cosmic rays from $10^{18}eV$ to the very highest energies with high statistical significance over the whole sky.

To resume, the surface array will have the following properties:

- 100% duty cycle.
- A well-defined aperture that is independent of energy above $10^{19}eV$.
- Uniform coverage in right ascension on a daily basis.
- A response that is largely independent of weather conditions.
- The quality of the data for each event improves with energy.
- Sensitivity to showers arriving at large zenith angles.
- In situ calibration of the detectors by cosmic ray muons.
- Measurement of the time structure of the arriving signals, which is sensitive to the mass of the primary particles.

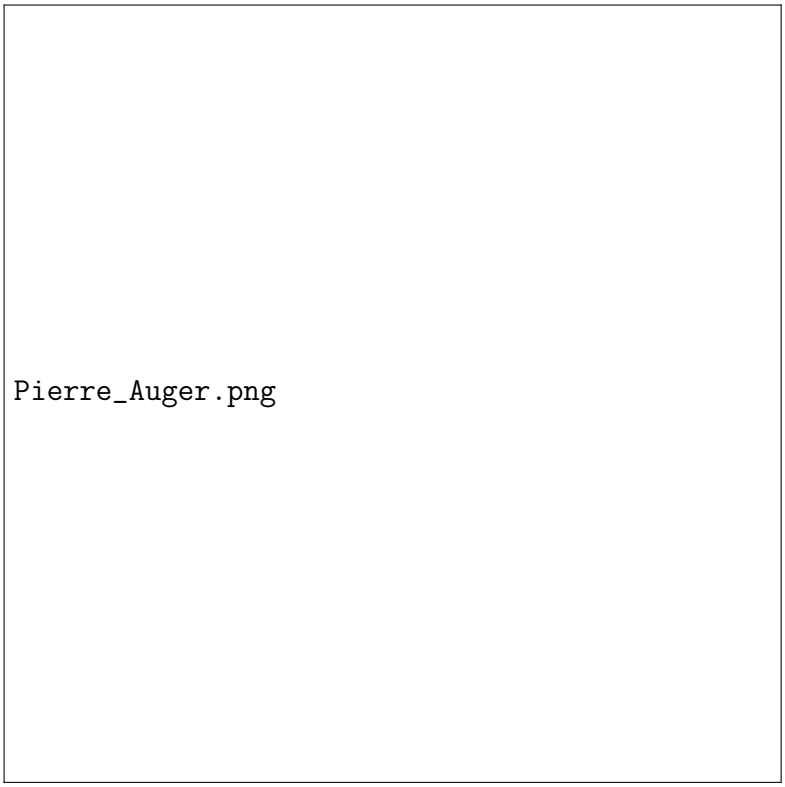

Figura 9: Pierre Auger Observatory

The fluorescence detectors can be operated during clear nights with little moonlight and have the following characteristics:

- Every event above $10^{19} eV$ is registered by at least one fluorescence detector: 6% of these events will be recorded by two or more fluorescence detectors. Essentially, every trans-GZK event will be a stereo event. Multiple station coverage improves the energy resolution.
- A coincidence of a single detector of the surface array with a single fluorescence telescope constrains the shower geometry as precisely as a stereo fluorescence detector.
- The longitudinal development profile is measured directly.
- The fluorescence detectors provide a more direct measure of the shower energy. The small, unseen, fraction of the total energy carried by neutrinos and

muons depends somewhat on the mass of the primary particle as well as on the hadronic interaction model.

3.2 AERA

The most important goal of Pierre Auger Observatory is to understand properties of Cosmic Rays, such as its energy, is incoming direction and its composition. During the last years, together with surface detection and fluorescence detection, techniques about radio detection are developed, thanks using radio auger antennas sensitive in MHz domain. All these antennas form an array, called AERA (Auger Engineering Radio Array), and these data from radio detector, in combination with the data of surface detectors and fluorescence detectors, can provide additional information on the development of extensive air shower. It has a site which overlaps with the SD infill array and is within the field of view of both the Coihueco and HEAT fluorescence detectors.

AERA is a new antenna system to measure short radio pulses emitted by cosmic ray air showers of the highest energies. It consists of an array of dozens of antennas sensitive in the frequency range of 30 to 80 MHz with signal processing and electronics developed specifically for this purpose.

AERA antennas are active 24 hours a day, like the surface detectors of the Pierre Auger Observatory. Radio detection of cosmic rays has been applied first over 50 years ago, but only with the digital signal processing available today it could be implemented on large scales yielding detailed and high-quality measurements.

AERA Antennas are 124 AERA radio stations covering an area of 6 km^2 are in operation. The 124 stations have been deployed in two steps. AERA-24 was deployed in 2010 and consisted of 24 radio stations equipped with **LPDA** (logarithmic-periodic dipole antennas) on a grid with 150 m antenna separation. Then in May 2013, 100 stations of a different design using so-called **butterfly antennas** were added. The 100 new stations in AERA-124 are spaced 250 m or 375 m apart from each other. Additional prototype stations with the ability to measure the electric field vector in 3D and at lower frequencies were also installed. In spring 2015, 25 more antenna stations will be deployed on a grid with up to 750

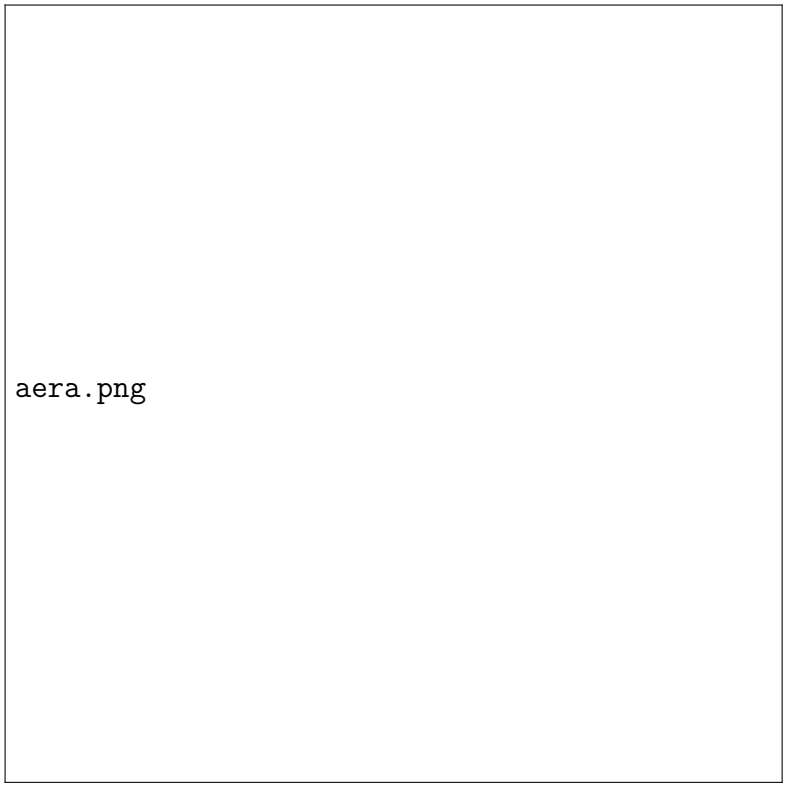

Figura 10: General Setup of AERA antennas

m antenna separation, increasing the instrumented area significantly and allowing improved studies of horizontal air showers.

3.3 General Setup

Before AERA there were experiments which intended to measure radio emission in the MHZ regime at Pierre Auger Observatory, like *RAUGER*. Since April 2013 AERA has been taking data; it consist of 95 radio detector station equipped with a type of bow tie antenna called the *Butterfly*. This kind of antenna was designed to be sensitive between 30 and 80 MHz, and measures in two polarization measure. One antenna arm is aligned parallel to the geomagnetic field (approximately 3°east of true north), with the other arm perpendicular to it.

These radio stations are equipped with either one of two different types of electronics. One type was developed jointly by the Karlsruhe Insitute of Tech-

Figura 11: Example of AERA antenna

nology (KIT) and the Bergische Universitat Wuppertal (BUW), and focused on the external trigger provided by the baseline Auger detectors. The other type was developed by Radboud University (RU) in Nijmegen and Nikhef, and focused on self-triggering on radio pulses. The station are disposed in a semi-regular grid, with 55 stations equipped with KIT/BUW electronics (mostly with a 250 m spacing) and 40 stations with RU/Nikhef electronics (mostly with a 375 m spacing). The major difference between the two types of electronics is still the focus of the different trigger mechanism.

The stations are designed to operate autonomously, generating power with a solar panel attached to the electronic box, which houses the digitizer and the battery. Into the stations equipped with RU/Nikhef electronics, this box also houses the scintillators used for triggering. Furthermore a commercial wireless comunication system antenna is installed on the top of the physics antenna, which communicates with one of the central data acquisition system (DAQS). In addition,

a GPS receiver is attached to the station to provide timing.

3.4 The Dutch electronic chain

Thanks to these antennas we're able to detect radio signals. The measured signal is amplified by a low-noise amplifier (**LNA**), located close to the center of Butterfly antenna. The characteristic of the electronics are determined by the interplay between the LNA and the antenna. The frequency responses and the group delays are there combined in a single response pattern, which will be called the *antenna response*. It should be noted that because of the directional sensitivity of the antenna this is not just dependent on the frequency and the phase of the signal, but is also a function of the incoming radiation. After the LNA, the signal arrives via coaxial cables to the electronics box and the filter boards inside the digitizer, which filter the signal between 30.0 and 80.0 MHz. After filtering, the signal is processed by analog-to-digital converters (ADCS), four of which are present in each digitizer. In RU/Nikhef digitizers there are 14 bit ADCS, which digitize the signal at 200 Msps, while the KIT/BUT digitizers use 12 bit ADCS, which sample with 180 Msps. The digitized signals are further handle by the field-programmable gate array (**FPGA**) and the central processing unit (**CPU**) on the digitizer board, which are able to acquire and format the data and make a trigger decisions. Two channels in the dutch digitizers are used to read out the signals from the antenna, while for the other two channels the filters are bypassed and are used to process signals from the scintillators. All the dutch stations are equipped with two small plastic scintillators, expect for five experimental stations with vertically polarized antennas, which only have one. There's one positioned on the top of the battery and one below it; the scintillator measure the charged particles in the air shower and are used for triggering. (REMEMBER: Download the images of the box!!) The stations equipped with dutch digitizers are able to self-trigger on any of the four connected channels. In its simplest form, a self-trigger is able to see just a signal crossing a threshold. The scintillator channels are shielded from most outside influences, so the signal is very clean and there is no need for a refined trigger scheme, and a single threshold is used. For the signal coming from the radio antenna this is different; the signals are weak and inherently noise, and can

be contaminated by pulsed *radio frequency interference* (**RFI**) that is unrelated to the radio signal from the air showers. that is unrelated to the radio signal from the air showers. For AERA an elaborate trigger algorithm was developed to reject noise; we're analyzing it in the following chapter.

If the FPGA and CPU decide there is a trigger on a station level, the data is temporarily stored and a time stamped message is sent to the DAQ, if three or more time stamps received by the DAQ fall within a coincidence windows of 3 μs, the DAQ requests the data from all triggered stations and builds an events. After the DAQ has created an event, it is written to disk and saved for off-line analysis. Initially, the length of all recorded time traces was 1024 samples (ns), but this was increased to 2048 samples (10240 ns) after 2 October 2014. In addition to the self-triggered events, the DAQ requests read-outs of all active stations once every 10s. These events can for instance be used to study detector performance or the noise situation and they are called *periodically trigger data*.

4 Trigger system for radio detection

(Wiki) In particle physics, a trigger is a system that uses criteria to rapidly decide which events in a particle detector to keep when only a small fraction of the total can be recorded. Trigger systems are necessary due to real-world limitations in computing power, data storage capacity and rates. Since experiments are typically searching for interesting events (such as decays of rare particles) that occur at a relatively low rate, trigger systems are used to identify the events that should be recorded for later analysis.

Because of the fundamental differences between the two electronic chains, the two types of radio stations use different trigger strategies; in particular we note that they are focused on two different trigger mechanism: the KIT/BUW stations on the external trigger provided by the baseline Auger detectors, while the RU/Nikhef stations is focused on self-triggering on radio pulses. We decide to analyze in the follow section the trigger system of the second type of station, for understanding the reasons why it was necessary to change it.

4.1 The previous trigger algorithm

The station trigger consists of a level 1 and a level 2 trigger. The level 1 trigger is formed in the FPGA and the level 2 in the Central processing Unit (CPU). The simplest trigger one can imagine a level 1 is a single threshold above baseline. This is the basis of AERA trigger, as indicated with threshold P_1 at time T_1 on the rising side of the pulse in the figure below. Several other parameters are implemented to select only isolated , transient pulses, as the rate of single threshold crossings cannot be handled by DAQ.

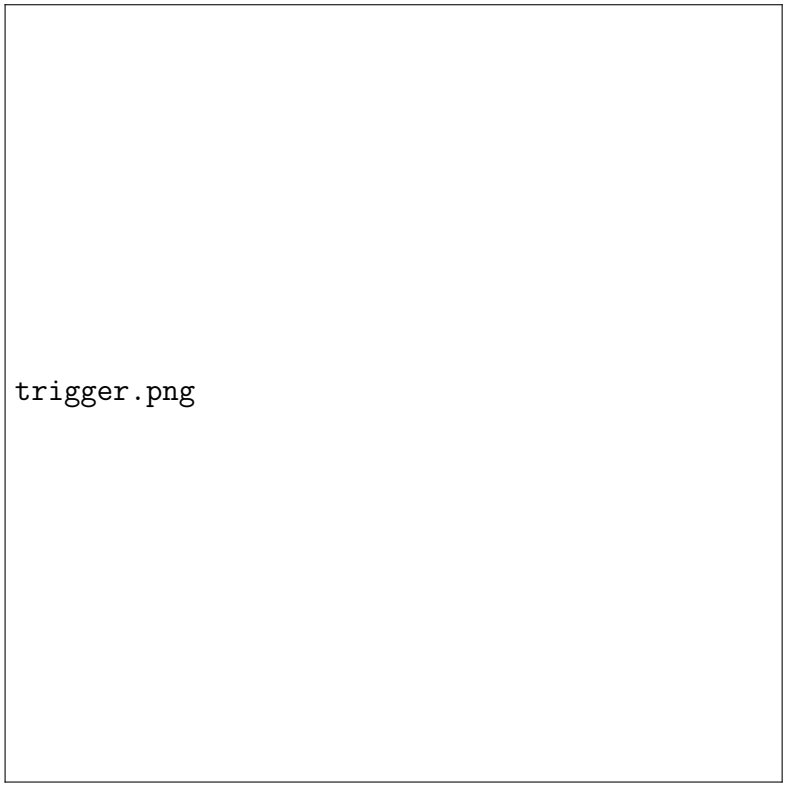

Figura 12: Schematic view of the station-level trigger.

The orange dots show where the signal crosses the threshold levels. The number of positive going threshold crossings and the time in which they occur, have to meet several conditions as specified by the following parameters:

- **Tprev** (previous) is the time during which there are no T1-threshold (signal) crossings allowed. This time may be set to values ranging from 0 to 255 time

5 ns, so the maximum is 1275 ns.

- **Tper** (period) is the time during which the rising T2-threshold (noise) crossings are counted (in NC) starting from a valid T1-crossing. This time may be set to values ranging from 0 to 255 time 25 ns, so the maximum is 6375 ns.

- **TCmax** is the maximum time allowed between the T2-threshold crossings. This time may be set to values ranging from 0 to 255 time 5 ns, so the maximum is 1275 ns.

- **NCmin** is the minimum number of T2-threshold crossings. This number may be set to values ranging from 0 to 255.

- **NCmax** is the maximal number of T2-threshold crossings. This number may be set to values ranging from 0 to 255.

Usually the trigger thresholds (P_1 and P_2) are dynamic and differ from station to station to keep the station-trigger rate constant at a manageable rate. Typical threshold values for a noise level with an RMS of 40 ADC counts are in the order of 200 ADC counts for P_1 threshold and 160 P_2 threshold. The other trigger parameters are not varied during a data-taking run.

4.2 Reasons to change our trigger system

4.3 Triggering on a raw traces

It was decided to analyze these data, collected in the last three months of 2014. The files that I use for my analysis comes from two different type of data detection: the first one consists in signals in coincidence with the SD, in which we are sure that those events were recorded due to an extensive air shower: the second one contains only traces detected by the radio detectors, really useful to understand the noise that affected these measures. However, for these traces it's really hard to have a good efficiency, because the only trigger system is given by the scintillators inside the electronic boxes, and it could start not only caused by the E.A.S. (for example cause by a muon not related with it).

In the picture (13) it's shown on the top some information, such as the number of the event, of the station and of the channel: channel 1 means that the event is recorded by the arms of the station in the direction East-West, channel 2 in the direction North-South, while channel 3 and 4 means events from the two scintillators stored in the electronic box. In these traces we report in the x-axis the time (in nanoseconds) versus ADC arbitrary unit(around 1 mV), a relative unit of measurement to show the ratio of amount of intensity to a predetermined reference measurement. Our goal is find in these traces to single out the signal comes from cosmic rays from the noise.

For this reason we write a code that permits us to recognize the signals of each stations. In a first time we try to analyze our traces detected by radio antennas.

For all the traces we need to obtain two important parameters from the histograms for our analysis : the bin of the histogram in which I have the maximum value and the width of my histogram after I cut it with a specific threshold. We decide to use two indexes whose start a the beginning an at the end of the histogram. Sliding this two indexes towards the center of the histogram we stop them only when we see a bin below the threshold and the follower is above, and the opposite in the following bins. After we will fill a 2D histogram with the bin which contains the maximum (**Xmax**) and the difference between the two indexes(**Width**).

Thanks to this two-dimensional plot we were able to define the region in the spectrum in which the signal is present. We choose to take 20 bins, between 455

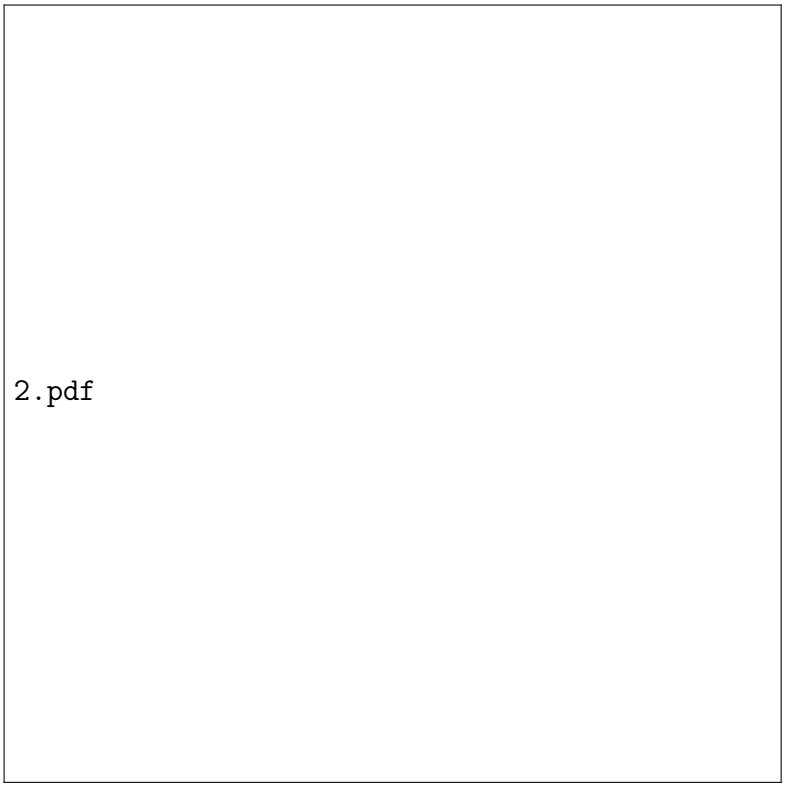

Figura 13: Example of raw trace.

and 475, defined as signal region, while the other parts of the spectrum are defined as Noise region. In detail:

- Whenever you read **signal** means: pulses between the bins 455 and 475 minus the pulses between 500 and 1500, divided by the appropriate factor of normalization (1000/20); In C++ language we could write:

  $Signal = Histogram- > Integral(455, 475)-20*hp- > Integral(500, 1500)/1000.$

- Whenever you read **noise** means ulses between 500 and 1500, divided by the appropriate factor of normalization (1000/20). In C++ language we could write:

  $Noise = 20 * Histogram- > Integral(500, 1500)/1000.$

So we decide to use a similar approach:

- Analyze only the histograms of the channels 1 or 2;

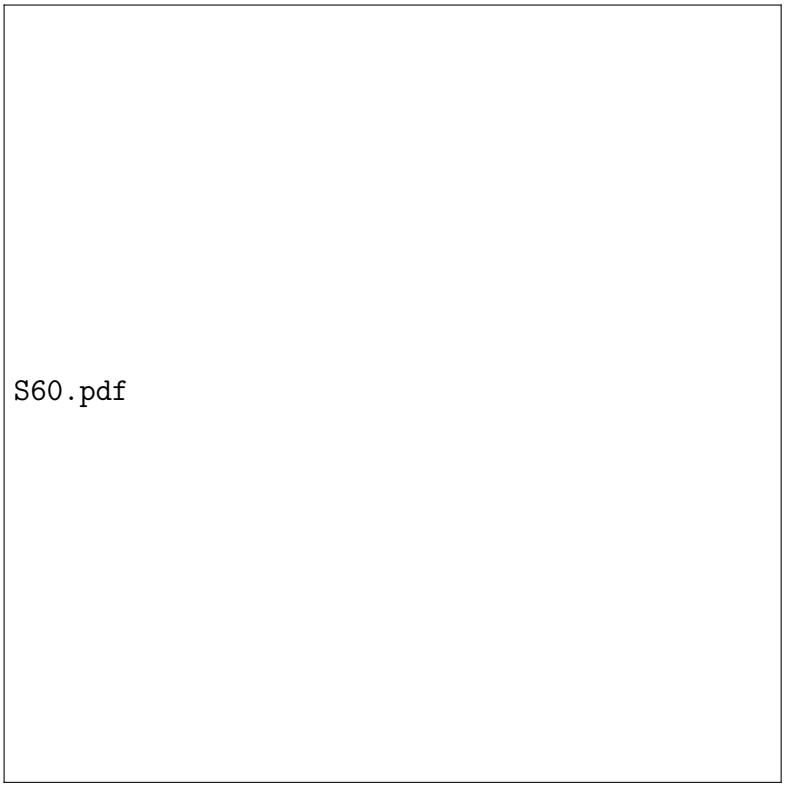

Figura 14: Example of two dimensional plot.

- For every histograms we use an index **imax** to find the maximum (if the content of one bin is greater of the following and the previous bins);
- Check if this maximum is greater than the set threshold. If yes, take two indexes **bmin** and **bmax**, defined as the difference or the sum between imax and a set width (**bsize**) the position of from which the following cycles start.
- Use another two indexes, at the beginning and at the end of the histogram an do the same of before. If we find maximum in these loops we call the position of the maximum **ifirst** and **ilast**;
- if the difference between ifirst and ilast is not greater that a set width, we could fill the histogram with the position of the maximum.
- Print at the end of the program the signal, defined as the integral between 455 and 475 minus the noise, defined as the integral between the bins 500

and 1500.

What we obtain in the end is shown in the picture below.

Figura 15: Final result of the first algorithm.

4.4 Triggering on a smoothed traces

Another algorithm that we apply to minimize the noise in these traces is called smoothed algorithm.

Smoothing is a process that It could be applied to obtain a better resolution; in many experiments in Science, the true signal amplitudes (y-axis values) changes rather smoothly as a function of the x-axis values, whereas many kinds of noise are seen as rapid, random changes in amplitude from point to point within the signal. In the latter situation it may be useful in some cases to attempt to reduce the noise by a process called smoothing. In smoothing, the data points of a signal

are modified so that individual points that are higher than the immediately adjacent points (presumably because of noise) are reduced, and points that are lower than the adjacent points are increased. This naturally leads to a smoother signal (and a slower step response to signal changes) . As long as the true underlying signal is actually smooth, then the true signal will not be much distorted by smoothing, but the noise will be reduced. In terms of the frequency components of a signal, a smoothing operation acts as a low-pass filter, reducing the high-frequency components and passing the low-frequency components with little change. It is often useful to apply a smoothing operation more than once, that is, to smooth an already smoothed signal, in order to build longer and more complicated smooths.

Most smoothing algorithms, called **linear smoothing**, are based on the shift and multiply technique, in which a group of adjacent points in the original data are multiplied point-by-point by a set of numbers (coefficients) that defines the smooth shape, the products are added up and divided by the sum of the coefficients, which becomes one point of smoothed data, then the set of coefficients is shifted one point down the original data and the process is repeated. Smooth operations can be constructed for any desired smooth width, m, usually an odd number. If the noise in the data is white noise (that is, evenly distributed over all frequencies) and its standard deviation is s, then the standard deviation of the noise remaining in the signal after the first pass of an unweighted sliding-average smooth will be approximately is over the square root of m $s/\sqrt{m}$, where m is the smooth width. Despite its simplicity, this smooth is actually optimum for the common problem of reducing white noise while keeping the sharpest step response. For shifting the points in our algorithm we wrote:

$$s_j = Y_{j-2} >> 4 + Y_{j-1} >> 2 + 3Y_j >> 3 + Y_{j+1} >> 2 + Y_{j+2} >> 4$$

Where we use the symbol s_j to indicate the j element of the new array, Y_j the j element of the old array and $<<$ the left-shift operator. It is the equivalent of moving all the bits of a number a specified number of places to the left.

For instance, consider the number **8** written in binary 00001000. If we wanted to shift it to the left 2 places, we'd end up with 00100000; everything is moved to the left two places, and zeros are added in the end. This is the number 32. In

fact, left shifting is the equivalent of multiplying by a power of two.

- Before all these operations we should work only in a side of the graphic: for this reason before that the algorithm is used it's necessary to overturn all the point below zero changing their sign.
- Assign the first two values of the old array, contained the elements of the traces, to a new array.
- Create a loop from the second bin until the end of the histogram and use smoothing algorithm.
- Repeat the last two points using another new array.
- Fill a new histogram with the last array;
- For this new array do the same analysis for the row traces (read the previous chapter);
- Print the ratio Signal/Noise and the smoothed trace (as in the figure below).

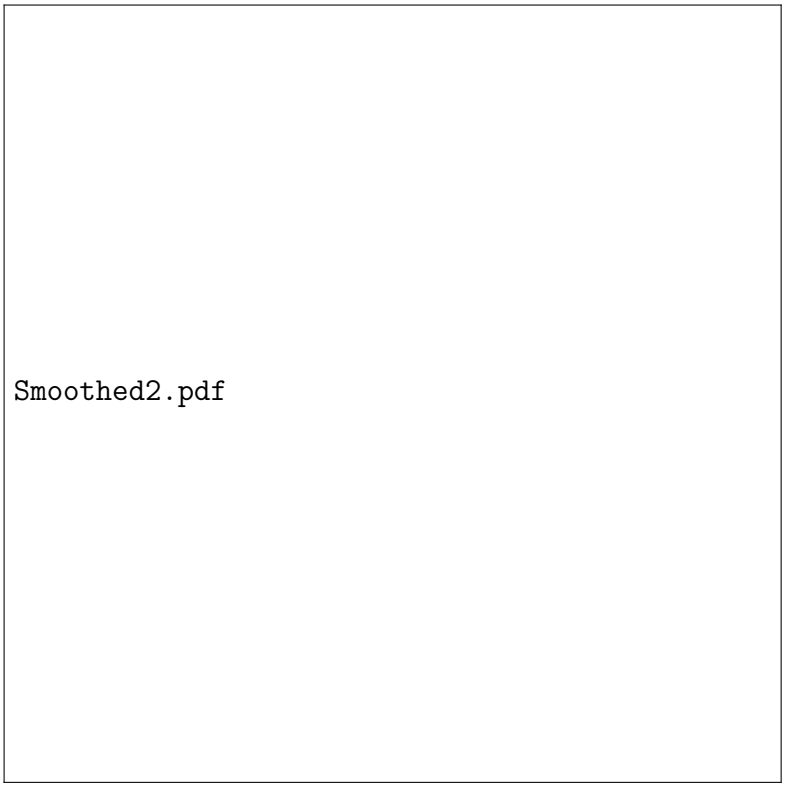

Figura 16: Example of smoothed trace.

4.5 Using the width of the signal

Trying to maximize the ratio $Signal/Noise$ we found another method for analyze our traces, reflecting about the features of the noise traces. In fact noise pulses look really wide or sometimes repetitive (for example in the case of **RFI**). So the general idea is, sliding the two indices as the previous algorithms, to check the distance between them for each histogram: if it results bigger than a set value, this is a proof that the pulse wasn't be produced by a cosmic ray.

In details:

- At the beginning we overturn our traces below the zero (take in absolute value);
- In addition assign the first two values of the old array, contained the elements of the traces, to a new array.

- Create a loop from the beginning until $end-bsize$, where $bsize$ is a set width.
- In this loop we create another loop, from 0 to bsize, in which we assign to every element of the first array the value got by the sum of the previous value plus the value of the correspondent element of the new array.
- The other $n - bsize$ elements of the new array, in which n is the size of the two arrays, are copied from the old array;
- We transfer the element of the last array in a new histogram, and after we check every bins: if the value is higher of the previous and the following bins, and furthermore is greater than a set threshold, we decide to acquire the value of the bins and its position in the histogram.
- if the position of this bin is between 455 and 475 it's defined as signal, otherwise noise (like the previous algorithm).

4.6 Optimize Parameters to maximize the ratio S/N

Looking in details these three algorithms it results clear that our analysis will be dependent on some parameters that we set before. In fact, instead of the logic it's different in the three cases, the acquisition part is quite similar in each algorithm. These parameters that we need to set before are:

- **Threshold**: it appears in all the algorithms analyzed. We decide to save the value of a particular bin only when it is above a set threshold.
- **Bsize**: it is present in all the algorithms analyzed as well. It represents the number of bins that we exclude from our main loop and in the first two algorithms it is helpful for define the two indexes *bmin* and *bmax*, that we use to check the rising and the falling edges of the pulses.

  Width : This is typical of the first two strategies and it is present in the last part: only in the position of the two indexes of the rising and falling edges is lesser than this quantity, then we could acquire the value and the positions of these bins. This number could be a non-integer number.

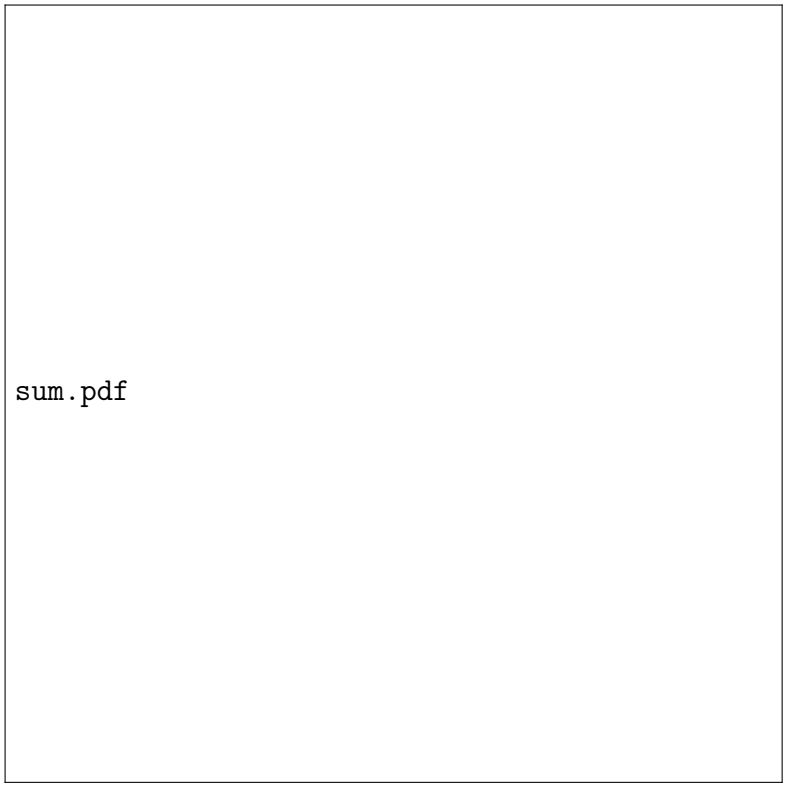

Figura 17: Final histogram of this kind of algorithm.

Changing these parameters in a small range it was possible to write tables (in the appendix) to study how the ratio Signal/Noise varies. The best ratios is shown in the tables below.

	Raw traces	Smoothed	Sum
Threshold	120	35	125
Bsize	80	50	3
Width	8	30	
Best ratio	8.51	0.54	0.35

Tabella 1: Values, for every algorithm used, which maximize the ratio Signal/Noise defined in the previous chapter for the East-West traces.

	Raw traces	Smoothed	Sum
Threshold	200	150	175
Bsize	50	120	3
Width	20	30	
Best ratio	1.71	0.35	0.19

Tabella 2: Values, for every algorithm used, which maximize the ratio Signal/Noise defined in the previous chapter for the North-South traces.

4.7 Estimate of background

The results obtained at this point are not enough: the main reason is that we would like to use these algorithms in a particular part of our electronic chain, and so it's not important only the ratio Signal/Noise but we need to check the value of noise, mease bye measure, in order to avoid pile-up problems.

Well, we know that the maximum rate that the Dutch electronic chain is able to support is around 500; so it's necessary to analyze our results with all this methods in order to esitmate the background that exist in them.

With this I mean that if we have N traces in which we study the rate in a length of B bins; and we know that each bin corresponds to a time T, we study in total a period of $NxBxT$ ns. If we know the number of noise pulses in that period, we know the rate in Hz.

In our case T is always 5 ns, we set N as 20 bins, while B depends generally on the direction of the pulses (North- South or East-West) and on the number of the traces present in the file. For example for the East-West pulses:

$20bins * 5ns * 11846histograms = 0.00186s$

So the noise pulses are divided by this value in order to get the noise background, and this is an important feedback to estabilish which algorithm is the best.

4.8 Estimate of events read by three or more stations

The next step in our analysis is to check how many events are read by three or more stations. This following check is necessary because, as it's written in the previous chapter, the Dutch stations are focused on self-triggering on radio pulses.

If a signal comes from an extensive air shower is read by three or more station, we're able to fix the three parameter and and then the geometry of the shower, as shown in the picture below.

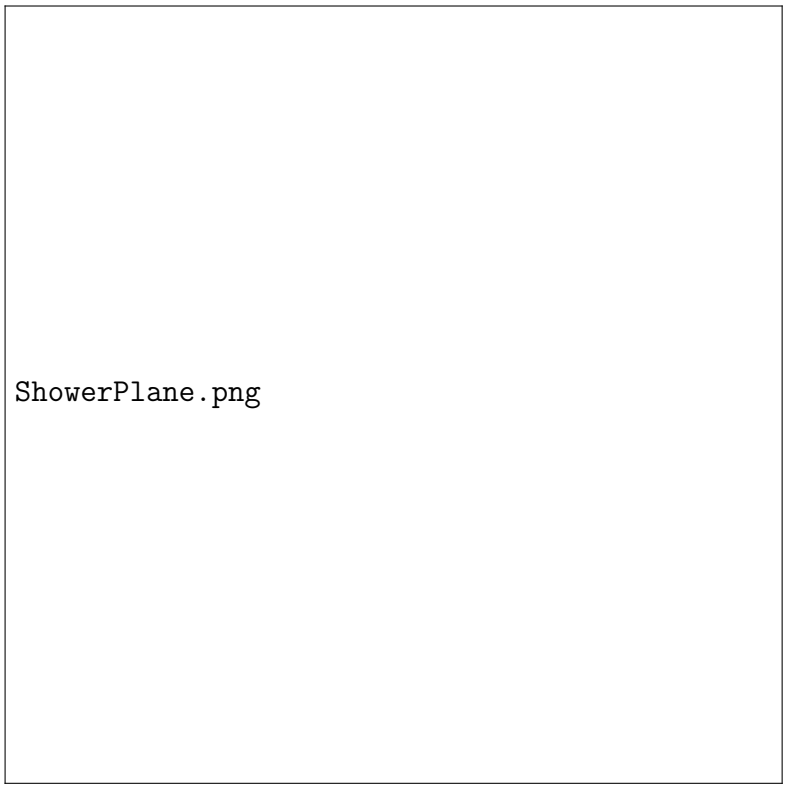

Figura 18: Geometry and parameters of an estensive air shower.

So we had to modify the code in order to do it.

- Define two new histograms: *hs* and *hsel*;
- Define new variables: *iev,ls,nls,nsel and isel* , inizialize to 0, and *event* , inizialized to -1;
- Use the function *sscanf* (scanf for the strings), in order to acquire and save, in the variables *iev* and *ls* , the number of the event and the number of the channel.

- Save the value iev in the variable event; if they are different we fill the histogram hs with the variable nls and the variable nsel and after we reset both.
- Increment the variable *nls*;
- Before starting the loop on the histograms reset the *isel* to zero.
- At the end of the analysis if we found the position of the bin in the signal range, we set the variable isel to 1.
- At the end of the second loop if *isel* is still one, we increment *nsel*;
- At the end of the main loop, when we print the ratio S/N, together we print the two histograms: the histogram *hs* in the end rapresents the number of stations per event, while the histogram *hsel* rapresents the selected stations per event.

4.9 Using events without SD informations

The tables show few events which pass the last two checks: if we set the value of threshold and widths in order to minimize the background (lesser than 500 Hz), we minimize also the events with three or more stations and we don't have enough statistic for build the geometry of the shower. (How can I link?)

This is the main reason why we decide to use other measures, obtained thans to a different tecnique: this second file consists of pulses not in coincidence with Surface Detectors (SD) and without any external trigger, except for the scintillator present in the box below the antenna. This is the reason why it is more difficult to have more efficiency, but at the same we have a lot of traces (low background), and it is easier to figure out the Background traces); so they are really useful later, when we know in which region we could find signal and in which noise.

Furthermore there are also problems linked to the trigger system, because not only muons come from the Extensive Air Shower could give signal in the scintillators, and surely it makes noise in our traces.

(Insert Image of the second data).

5 FPGA

5.1 What is an FPGA?

A **field-programmable gate array** (FPGA) is an integrated circuit (IC) that can be programmed in the specific field after manufacture. FPGAs are similar in principle to IC, but have vastly wider potential application than **programmable read-only memory** (PROM) chips.

In the last years FPGAs might allow computer users to make microprocessors in order to fit their own individual needs.

FPGA contains an array of programmable logic blocks, and a hierarchy of reconfigurable interconnects that allow the blocks to be wired together, like many logic gates that can be inter-wired in different configurations. Logic blocks can be configured to perform complex combinational functions, or merely simple logic gates like AND and XOR. In most FPGAs, logic blocks also include memory elements, which may be simple *flip-flops* or more complete blocks of memory.

The functionality of a digital Integrated Circuit (IC) consists of many integrating interconnected logical gates in a single chip. Most integrated circuits are application-specific, which means that they have a fixed function for a specific purpose. This kind of circuits are generally referred to as Application Specific Integrated Circuits, or **ASIC**s. ASICs are expensive and not easy to design, but single-unit costs are low, making ASICs the ideal solution for mass-productions. However, in applications such as instrumentation development it is preferable to create ICs with new functionalities in limited quantities and relatively short time (ASICs are really time spending for design). For these occasions a **Field Programmable Gate Array** (FPGA) is often the best solution: FPGAs are mass-produced ICs containing large numbers of configurable logic gates and programmable interconnections, allowing rapid and economical development of new digital designs. They have great flexibility compared to full-custom ASICs, they are re-programmable to any specific application and most errors can be simply corrected by loading a new firmware design into the device. This flexibility comes at some cost: the general-purpose architecture of FPGAs compromises digital performance to some degree relative to an ASIC. And the price per unit of an

FPGA is larger than that of a comparable ASIC, making them not useful for large production volumes.

This is the main reason why FPGAs are normally used in products with relatively small quantities. However, FPGAs can also be used in products where future updates to the functionality might be expected and at the same time they can even be used during the prototyping process for a new ASIC, since the same FPGA can be reconfigured many times to test new functions and to correct mistakes, compared to reiterations of ASIC could be very expensive.

5.2 How to modify our code for an FPGA

It should be really careful to not forget where we would like to insert our trigger code: in the electronic chain we remember that if the FPGA and CPU decide there is a trigger on station level, the data is temporarily stored and a time stamped message is sent to the DAQ.

Figura 19: Block Diagram of the NIK/RU.

So we need to translate our code (in C++) in order to replace the old firmware in the FPGA. This is not so easy because it's necessary to write the same ideas in two different languages. In this chapter we try to introduce the new language for programming and the fundamental differences between the two languages.

We develop a design for an FPGA involves writing a description of the design's functionality in a so-called Hardware Description Language (HDL). The most common HDLs are VHDL (VHSIC Hardware Description Language) and Verilog. The functional level of the code can be decided by the user. If someone prefers to write high-level coding style by describing the functionality of the design, then the compiler will do its best to translate it into working hardware. The user can also choose low-level coding style to describe the design in terms of the FPGA primitive station level, in order to have better control over performance. Most designs usually end up as a mixture of both high and low-level coding. Besides simple text editors there exist a number of graphical design tools, which in turn generate HDL code for implementation.

So let me compile some of the basic differences between C programming and VHDL programming.

- C is a middle level language; It is a mix of a high level language and an assembly language. VHDL is a hardware description language(HDL); It is used for implementing the hardware circuit.

- C can only handle sequential instructions while VHDL allows both sequential and concurrent executions.

- C is a sequential language, which means no two lines of code can not generate results at same time. This is because C runs on micro-controller/processor and it can only execute one instruction at a time. Verilog/VHDL are not sequential rather parallel, which means different parts of code will generate hardware which will work in parallel at the same instance in time.

- A C program can be successfully written with pure logical or algorithmic thinking. A successful VHDL programmer needs thorough working knowledge of the hardware circuits and he should be able to predict how a given code will be implemented in hardware.

- Normally we don't care about resource usage in C because a C program is usually ran on a computer which uses a powerful processor with high speed.Furthermore we also don't care about the memory usage. But when it comes to VHDL a slightly complicated code could be different; the memory and other logic elements are limited in a FPGA(where normally put the VHDL code in). This is why it is very difficult to implement image processing algorithms in VHDL than in C.