Capitolo 3

The simulation program

Simulation is important during in the project of an experiment to verify the efficiency of the building choices and to develope the analysis instruments while waiting the raw data.

For this mean AliRoot has been developed; it is a framework for simulation, reconstruction and analysis od the Alice data. AliRoot is based on the Object Oriented programming, it is written in C++ and it is developed from the pre-existing framework Root, which includes other libraries such as Pythia and Hijing for the physics simulation and Geant for transporting particles through the detectors. [1]

3.1 The simulation in theAlice experiment

AliRoot simulates the work of all the Alice subdetectors and it can be used to analyze the results. The simulation starts producing the physical event that caracterize the experiment and goes on transporting the particles through the present materials; if a particle goes through a sensible detector, the program simulates the response, such as the the charge on the SSD strips as ADC channels. In this way the simulation includes a lot of phenomena from the physic event we want to study to the strumental readout. All these informations must be analyzed by special software instruments.

AliRoot manages all the informations about the Alice geometry. Particulary, it knows the shape, the position and the composition of all the subdetectors and their parts. The geometry is converted from the technical draws used to build up the Alice detector; detector but also support structures, such as ladder in carbon fiber to support the silicon modules of ITS or the tubes of the cooling system. It is important to take care on all the

[1] See appendix A for more technical details

part of the ALICE detector, because also non-sensible parts can affect the measures, for example by multiple diffusion or back-scattering phenomena.

Simulation begin with the phisical events production. To obtain this, ALIROOT uses specific generators, such as PYTHIA and HIJING, which simulate the starting crash between lead iones.

ALIROOT gives to GEANT the geometrical informations about position and composition of the materials constituting the experimental apparatus. The GEANT libraries allow to simulate the interaction between a particle and the matter it goes through; when a particle goes from a material to an other, the program stores its position and the energy loss; these four coordinates (three of position and one of energy) are called *hit*.

ALIROOT simulates the response of each detector using the hits produced by GEANT. The set of numeric informations which a detector can extract from a hit is called *digit*, and the process to transform hits in digits is called *digitisation*; a digit is, for example, the strip o a SSD module and the energy it has collected. To transform hits in digits, ALIROOT recalls geometrical informations again, to to connect each hit to the corresponding detector, and so it can apply the right digitisation algorithms.

The set of informations of the digits is used, at the end, to reconstruct points, tracks and so events. Geometry is again very important to connect the informations of all the detectors. The points are reconstructed for each detector, but only considering together the informations of more than one detectors allows track reconstruction and to recreate an event all the detectors are necessary. The reconstruction algorithms can be applied to simulated data as well as to real data.

The power of ALIROOT is its completeness; in fact the framework, starting from the phisical parameters set out run time, simulates the physical events, transpots the particles, simulates the detectors response and reconstructs the events.

3.2 Beam test simulation

As just seen, set out a particular geometry of the detector, AliRoot provides all the instruments for the simulation and the reconstruction. From that we decided to introduct in AliRoot the beam test geometry to reuse completely the existing codes, in a tipical object-oriented way. So we we needn't to rewrite the algorithms; moreover, as written later, we can test the existing algorithms.

3.2.1 Beam test geometry

In the simulation of the Alice experiment, all the detector are present, particularly ITS whose modules are disposed as in section 2.2. For the beam test we used only some SSD modules, so the geometry has benn completely rewritten to describe this setup.

A SSD module is simulates, in Alice as well as in beam test, as a silicon parallelepipedon with a surface of 7.3×4.0 cm^2 and a thickness of 300 μm. In Alice each module has a specific position inside ITS, but here the testing detector is the only structure. The geometry was completely rewritten rather than maodify the existing one, beacuse of the possibility of introducing othe modules forming a telescope such as the one used during the beam test. The alternative was setting as no-sensible all the SSD modules of ITS but one, and studying its behavior witha specific beam. In this way we couldn't introduce a telescope with an arbitrary setup.

Beause of its simplicity, the beam test geometry was insered manually in AliRoot. The choice of the geometry to use, the Alice's one or the beam test's, takes place run-time and it is completely transparent after that, during simulation and riconstruction, beacuse of the framework modularity.

3.2.2 Beam simulation

In Alice the production of the physical event is performed by algorithms coded in the AliRoot libraries (Hijing and others). In the beam test the physical event is a beam of minimum ionizing particles (see paragraph 4.1), in particular we used π^+ of 7 GeV/c momentum. The particles are created with a momentum constant in modulus and direction. To simualte the real dimensions of the beam, the position of π^+ production changes randomly in a Gaussian distribution (σ=2 cm) around the axis of the beam.

The beam characteristics, and in general all the Geant parameters, are set run-time, making the simulation more efficient beause the source code doesn't need to be recompiled any more.

3.3 Response simulation for SSD modules

The response simulation for a SSD module has been yet implemented in the past[9]. To make the simulation as realistic as possible, the parameter must be modified to reproduce properly the experimental results. The comparison must take place using a simulation reproducing at the best the experimental conditions.

In the next paragraphs I'll present the simulation steps; next I'll present the results of the simulation. The simulation reflects the beam test setup. The description and the analysis of the beam test are explained in the chapter 4, while the comparison between experimental data and simulation is explained at the paragraph 5.1.

3.3.1 Hit production

For GEANT a SSD module is simply a parallelepipedon made of silicon. Each particles crossing the module produce al least two hits, the first one when it enters the module (*entering hit*) and the second one when it goes out the module (*exiting hit*); sometimes, if the crossed thickness or the energy loss are greater than a threshold, also *inner hits* are produced.

Usually in the beam test simulation there are only entering and exiting hits, because the particles are minimum ionizing so the energy loss is very small.

3.3.2 Digitisation

For each event, ALIROOT considers all the hits connected to the module. For any track the entering hit is stored and the program sums its energy to the one of the exiting and inner hits; also the exiting position is stored. In this way for each particle the entering and exiting posistion and the energy loss in the module are known. A particle can not exiting the module, for example because of decays or loosing of all the cinetic energy. In this case the particle *dies* inside the module and the stopping position is considered as the exiting position.

The module response is simulated in this way[9]:

1. The charged particle crossing the module lose energy continuously, but GEANT considers only the global loss. It is not possible to simulate a continuous, so the travel of the particle in the module is divided in 25 μm steps and the same for the energy; this choice is a compromise between the execution time and the need of simulating a energy loss

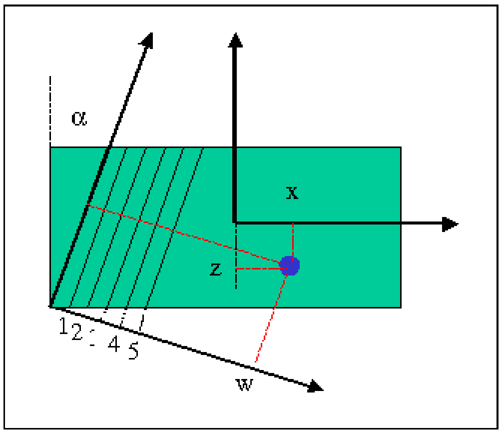

Figura 3.1: Transformation of the coordinates (x,z) in the plane. The strips are sloping respect to the module side.

as realistic as possible. Each point plays as a source of electron/hole couples.

2. The drif time taken by the electron/hole couples to reach the strips is

$$t_p = \frac{y + s/2}{v_p} \qquad t_n = \frac{s/2 - y}{v_n}$$
$$v_p = 0.86 \times 10^6 cm/s \qquad v_n = 2.28 \times 10^6 cm/s$$

   where y is the direction of the electric field, $s/2$ is the tickness of the SSD module and v_p and v_n are the drif speeds of electrons and hole respectively[14].

3. The coordinates couple (x,y) is converted into the local coordinate w, connected to the position respect to the strips (see figure 3.1).

4. The charges arrive to the strips with a Gaussian dispersion respect to the origin point, and the variance is

$$\sigma^2 = (2 \cdot D \cdot t)$$

$$D_n = 30\, cm^2/s \qquad D_p = 11\, cm^2/s$$

   where t is the drift time; about D_n and D_p values see [14]. The charge is now divided around the strips according to the part of gaussian area covering them (see fig. 3.2).

5. To each strip signal, obtained as seen, a gaussian noise (σ_p=420, σ_n=625) is added; it simulates the total effect given by all the real noise. These

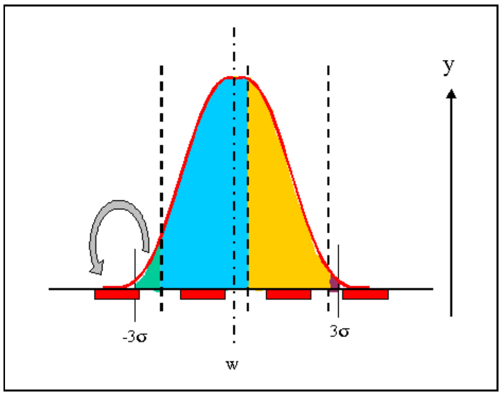

Figura 3.2: Charge integration: for each strip the charge is obtained integrating the dispersion Gaussian in the strip space. The Gaussina is truncated at 3σ

values were calculated using the misures on previous version of SSD modules: so they must be adjusted according to the results of this work.

6. A ratio of the previous strip and of the following one is added to each strip; this is the simulation of the activation due to the capacitive coupling. The value of this ratio was setted according to the following idea.

   The figure 3.3 shows the sketch of the inter-strip coupling circuit.

   If Q is the charge collected by the central strip, Q_1 is the signal read by the readout chip for the strip affected by the particle crossing and Q_2 is the signal for the near strips, there are the relationships

$$Q_1 = Q\frac{C_1 + C_2}{C_2 + 3C_1} \qquad Q_2 = Q\frac{C_1}{C_2 + 3C_1} \tag{3.1}$$

   If we change C_1 and C_2 into the experimental values[2], we obtain the coupling of the table 3.1, also used in the simulation.

Side	C_1	C_2	Q_1/Q	Q_2/Q	Q_1/Q_2
P	3.34	306	0.98	0.01	0.01
N	11.10	297	0.93	0.03	0.03

Tabella 3.1: Capacitive coupling. The C_1 and C_2 values were measured[2]. The other ratios are obtained inserting C_1 and C_2 in the relationships 3.1.

7. The so obteined signal is converted into an ADC value; to decide the conversion factor we consider the ADC channel corresponding to the

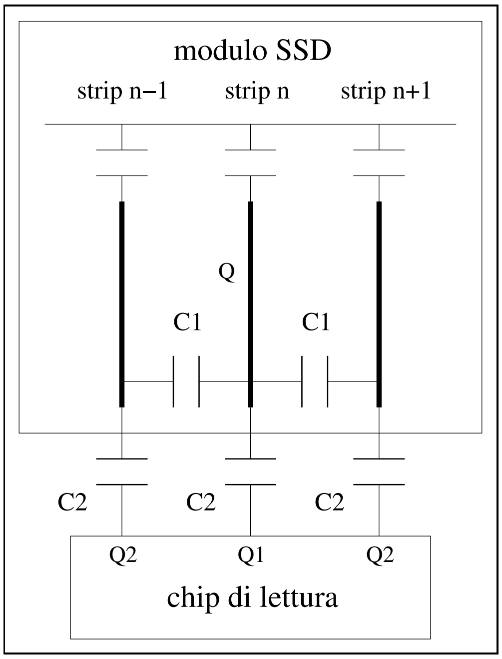

Figura 3.3: Inter-strip capacitive coupling circuit. Q is the charge collected by the strip,Q_1 and Q_2 are the signals read by the readout chip

peak of the energy loss for a minimum ionizing particle. Then we decided that channel 40 corrisponds to the signal from by 25 000 electron/hole couples produced by a minimum ionizing particle going through 300μm thick silicon[6]. This choice follows the results for the old module version and it will be verifies according to the results of the next chapter. The total number of ADC channel is 1024; in the simulation, if the signal overflows channel 1024, it is piled up in this channel; also this choice must be verified looking at the tecnical charactestic of the modules used in the beam test.

8. If the signal doesn't go over the noise threshold, it becomes zero and it is not stored, to reduce the amount of recording data.

The digits stored on disk record the informations about the activated strip and the produced signal.

3.4 Simulation results

For the simulation we consider a statistic of 10 000 events; for each event a 7 GeV/c π^+ was created, with momentum perpendicular to the SSD module. The transport (GEANT) and the digitation took place as seen in the previous paragaph.

The figure 3.4 shows the charge distributions on the strips for the P-side and for th N-side. They are the histogram of the signal (ina ADC channels) for each strip. The 34 000 entries give an idea of the mean number of digits for each event.

Because of the conversion factor from electron/hole couples and the signal in ADC channels, for minimum ionizing particles a peak around channel 40 is expected. We can also see a second peak around channl 20 and a third one just over the noise threshold. These secondary peaks exist because, also if the more of the charge is collected by the strip which is the nearer to crossing point of the particle, a little part is collected by the other near strip , because the particle entered the module almost in the middle of two strips or because of the capacitive coupling. Sometimes the signal can exceed the noise threshold, so it increase the secondary peaks.

Also if the histogram was drawn after the cut of the noise, there is a peak at small ADC values. With a 4σ threshold we can cut off 99.9% of the noise; to know how many noise events we can find over 4σ, we must remind that for each event all the 768 strips of each side are read, so the total counts for each side is of the order of 10^7; in this way the number of noise events over a 4σ threshold is of the order of 10^5. This value is comparable to the integrals of the first peak in each histogram.

The primary peak followes a Landau progress[14]: the results of the fit with a Landau function are in the table 3.2.

3.4.1 Cluster reconstruction

Usually a particle losing energy in a SSD module actives more than one strip; the set of the near strip activated by the same particle il called *cluster*. The charge of a cluster is the sum of its strips signals; this value is also the energy loose by the particle which created the cluster.

The position of th cluster is obtained averanging the position of the acivated strips, using their signal as weight; this value is the *centre-of-gravity* of the cluster.

The *size* of the cluster is the number of its strips. This parameter is very important to understand if the cluster was rightly reconstructed an if the charge was correctly divided around the strips. Usually a particle crossing

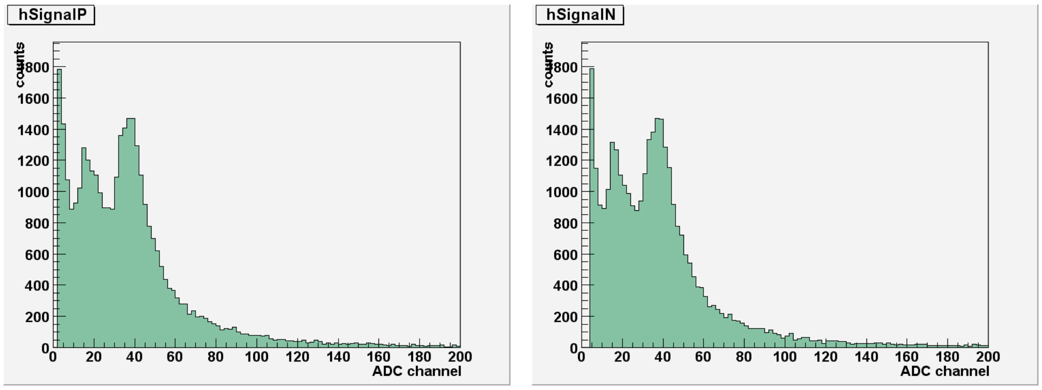

Figura 3.4: Histogram of the charge distribution on single strip.

perpendicularly the module, actives one or two strips[3], hardly even tree ones, according to the position respect to the strips.

The figure 3.5 shows the charege distribution for the clusters. Now the secondary peaks of figure 3.4 are added to the primary peak, because the strip activated by the same particle are collected in the same cluster. The distribution has the characteristic Landau shape; the table 3.2 shows the fit paramenters.

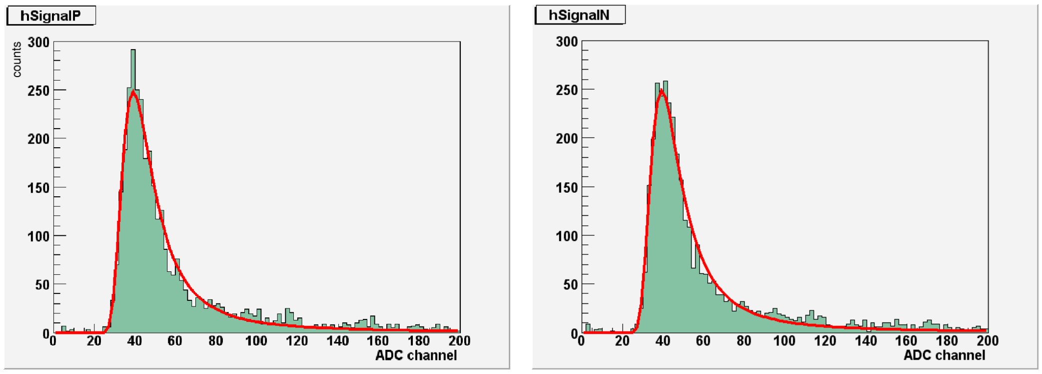

Figura 3.5: Cluster charge distribution histogram.

The histogram integral is smaller than the previous one because we don't consider the single strip but the cluster. The peak moved to higher value because now we consider all the charge which was spreaded on more tha one strip in the previous analysis.

Most of the cluster are made of one or two strips as we expected, while

	P-side		N-side	
	mean (ADC ch.)	σ (ADC ch.)	mean (ADC ch.)	σ (ADC ch.)
Single strip				
First peak	18	6	18	6
Second peak	38	4	38	4
Clusters	40	5	40	5
SNR	36	4	25	3

Tabella 3.2: Fit parameter using Landau curves for the charge distribution.

three or more stris clusters are less frequent (see figure 3.6). The mean cluster size is 1.8 strip on P-side and 1.9 on N-side.

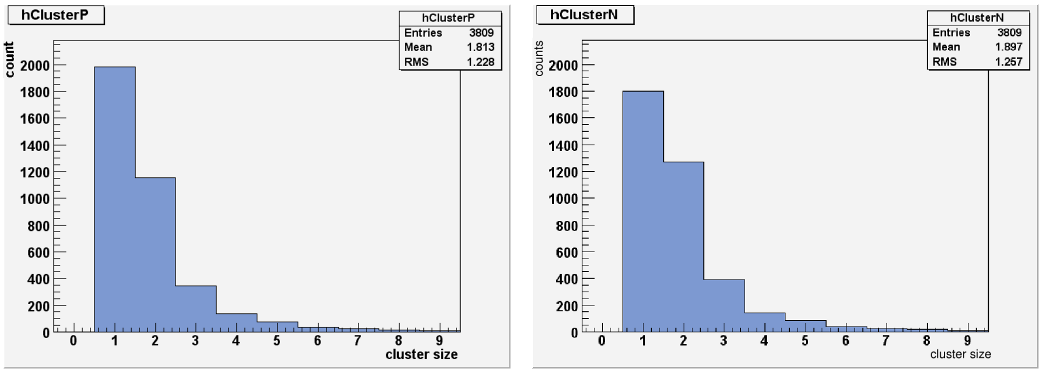

Figura 3.6: Cluster size histogram

3.4.2 Geant parameters analysis

Before to compare the beam results with to the simulatin ones and to check all teh important parameters of the simulation, it is necessary to study the Geant parameters, apart from the digitisation ones seen in the paragraph 3.3.2. One of the most important parameter for tansporting particles is the behavior of $\delta - rays$. Geant allows to treat $\delta - rays$ in three ways:

1. $\delta - rays$ are not reproduced. Energy loss changes according to a Landau distribution (see figure 3.7 and 3.8, blue histogram)

2. energy loss follows the Bethe-Block function, with $\delta - rays$ energy only over 1 MeV; under this threshold the simulation goes on as in the previous item (see figure 3.7 and 3.8, red histogram)

3. there are no fluctuation in energy loss; $\delta - rays$ are produced and transported also if the energies are lower than 1 MeV (see figure 3.7 and figure 3.8, green histogram)

Figures 3.7 and 3.8 show the comparison among the three simulations.

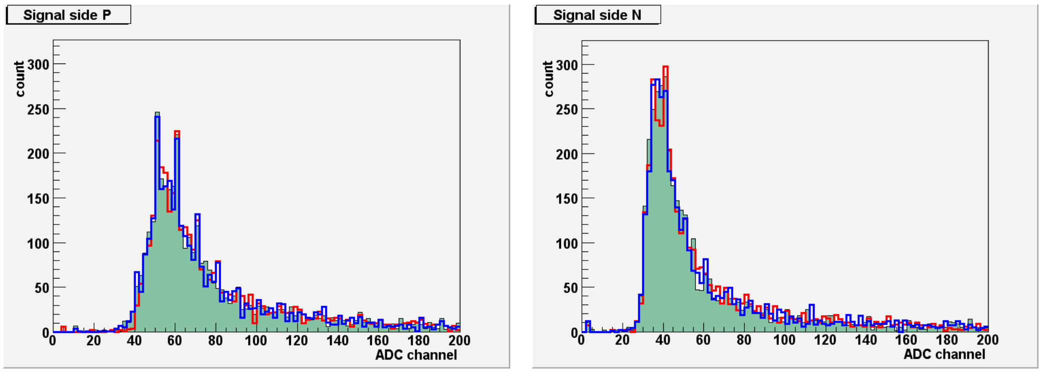

Figura 3.7: Cluster charge distribution. In the blue histograms there aren't any $\delta - rays$, in the red ones there are only the $\delta - rays$ with an energy over 1 MeV, in the green ones all the $\delta - rays$ are produced and transported.

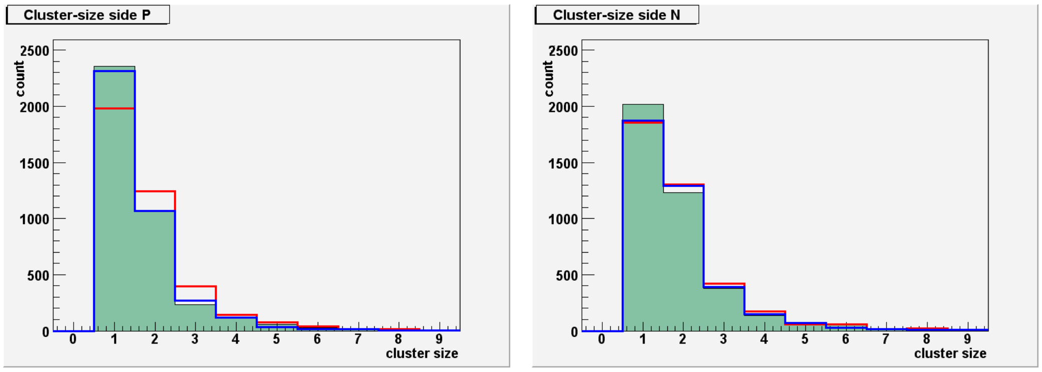

Figura 3.8: Clusters size. In the blue histograms there aren't any $\delta - rays$, in the red ones there are only the $\delta - rays$ with an energy over 1 MeV, in the green ones all the $\delta - rays$ are produced and transported.

The charge distribution doesn't depend too much on the simulation paramenters, so the treatment of the energy a particle loses is suitable independently on $\delta - rays$. Clusters size is more sensible to $\delta - rays$, as expected. The simulation is stable respect to changing in these GEANT parameters.