Università di Pisa

Dipartimento di Fisica

Curriculum in
Fisica delle Interazioni Fondamentali

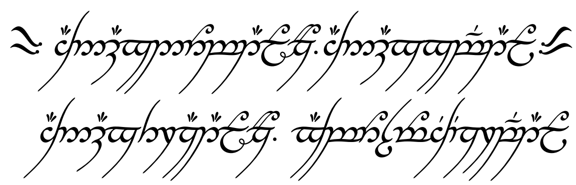
Real-time triggering with GPUs in the CERN NA62 K decay experiment

Candidato:
Stefano Di Lorenzo

Relatore:
Pr. Marco Sozzi
Co-Relatore:
Dott. Gianluca Lamanna

Maggio 2016

To Rosanna



Contents

Abstract

This thesis reports a study for a new real-time trigger for the NA62 experiment based on Graphics Processing Units (GPUs).

The aim of the NA62 experiment is to measure the branching ratio of the ultra-rare decay $K^+ \to \pi^+\nu\bar{\nu}$, a process mediated by Flavour-Changing Neutral Currents (FCNC), with a precision of 10%. Since the value predicted by the Standard Model is very precise, the measurement of this quantity represents an excellent way to investigate the existence of New Physics, or in case of agreement with the Standard Model(SM) to improve the current knowledge of the $|V_{td}|$ parameter of the CKM matrix.

The use of a high-rate kaon beam will result in an event rate of about 15 MHz, so high that it is impossible to store data on disk without a very selective reduction. The experiment will use devised three trigger levels, allowing to reduce the data rate fed to the readout PC farm down to $\sim$10 kHz.

In this thesis I report a study for a fast multi-ring fitting algorithm, fed with the data of the RICH(Ring Imaging Cherenkov) detector to be used in L0 level trigger of the experiment.

The necessity of running the algorithm in real-time, with a maximum latency of 1 ms per event, drove the choice of exploiting the parallel computing power of GPUs. I developed an online seedless ring fit algorithm running on GPUs, satisfying the L0 trigger time requirement, which achieves resolutions comparable to those obtained by the offline reconstruction.

I studied how the use of the algorithm at L0, increases the quality of the collected data with respect to the standard trigger of the experiments for a specific case, the Lepton Violation decay $K^+ \to \pi^-\ell^+\ell^+$. The measurement of this decays would be judicate a Majorana nature of neutrinos and the existence of New Physics.

This work proves that alternative trigger designs are possible for the NA62 experiment, and represents a starting point for the introduction of flexible GPU-based real-time triggers in High Energy Physics.

Part I

The NA62 Experiment

Chapter 1

The NA62 Experiment

Contents

The main goal of the NA62 experiment is the measurement of the ultra-rare decay $K^+ \to \pi^+ \nu\bar{\nu}$ at the CERN SPS (Super Proton Syncrhotron).

This decay was observed for first time at BNL in the two dedicated experiments E787 and E949 and the measured branching ratio was $\left(1.73^{+1.15}_{-1.05}\right) \cdot 10^{-10}$[11]. This decay, with its the neutral companion $K^0_L \to \pi^0 \nu\bar{\nu}$ is a unique probe to test the Standard Model and search for the existence of new Physics. NA62 aims to collect a hundred $K^+ \to \pi^+ \nu\bar{\nu}$ decays with a $10:1$ signal to background ratio.

1.1 Theoretical framework

The decays $K^+ \to \pi^+ \nu\bar{\nu}$ and $K^0_L \to \pi^0 \nu\bar{\nu}$, the second one studied by the KOTO experiment in Japan[39], due to their theoretical precision (in SM) are among two of the strongest test of the Standard Model . Both decays are *Flavour-Changing Neutral-Current* (FCNC) processes: this

kind of transition is strongly suppressed in SM and therefore is very sensitive to new Physics scenarios. The processes described above are due to $\bar{s} \to \bar{d}\nu\bar{\nu}$ transition at quark level and can be described by "penguin" Feynman diagrams (Figure 1.2).

The predicted branching ratio $K^+ \to \pi^+\nu\bar{\nu}$ in the Standard Model is $\left(7.81^{+0.80}_{-0.71_{CKM}} \pm 0.29\right) \cdot 10^{-11}$ [17], where the first error takes in account the uncertainty of CKM matrix elements and the latter one is pure theoretical.

Any discrepancy between the Standard Model prediction and an experimental result would be an evidence of new Physics and the hint for the existence new undiscovered particles intervening in the decay.

In case of agreement with the Standard Model, a precise measurement would improve the accuracy of $|V_{td}|$, which is one of the least precisely known parameters of the CKM matrix. In particular this measurement would be independent from those obtained from the measurement of $\Delta m_d = m(B^0) - m(\bar{B}^0)$ in neutral B-meson mixing[40, 30, 56].

1.2 CKM matrix

The CKM matrix provides an extension of the Cabibbo 2×2 matrix, that encodes how flavour-changing charged currents mediated by $W^\pm$ couple u, c and d, s quark states [22]. The coupling is described by means of the intermediate weak eigenstates d' and s' obtained from mass eigenstates d and s through a rotation by angle θ_C.

$$\begin{pmatrix} d' \\ s' \end{pmatrix} = \begin{pmatrix} \cos\theta_C & \sin\theta_C \\ -\sin\theta_C & \cos\theta_C \end{pmatrix} \begin{pmatrix} d \\ s \end{pmatrix} \tag{1.1}$$

The (d', s') are the weak eigenstates appearing in the weak charged current

$$J_\mu \propto (\bar{u}, \bar{c})\gamma_\mu(1-\gamma^5)\begin{pmatrix} d' \\ s' \end{pmatrix} \tag{1.2}$$

The Cabibbo-Kobayashi-Maskawa (CKM) generalizes the Cabibbo matrix including the quarks b, t of the third generation[43]:

$$\begin{pmatrix} d' \\ s' \\ b' \end{pmatrix} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \begin{pmatrix} d \\ s \\ b \end{pmatrix} \tag{1.3}$$

The CKM matrix is an unitary matrix so $V_{CKM}^{\dagger} V_{CKM} = \mathbb{I}_3$. The main consequence of the CKM matrix unitarity is the absence of flavour changing neutral current (FCNC) processes at tree level in the SM.
Experiment yield the following results for the CKM matrix elements[54]:

$$\begin{pmatrix} |V_{ud}| = 0.97425 \pm 0.00022 & |V_{us}| = 0.2253 \pm 0.0008 & |V_{ub}| = (4.13 \pm 0.49) \times 10^{-3} \\ |V_{cd}| = 0.225 \pm 0.008 & |V_{cs}| = 0.986 \pm 0.016 & |V_{cb}| = (41.1 \pm 1.3) \times 10^{-3} \\ |V_{td}| = (8.4 \pm 0.6) \times 10^{-3} & |V_{ts}| = (40.0 \pm 2.7) \times 10^{-3} & |V_{tb}| = 1.021 \pm 0.032 \end{pmatrix} \tag{1.4}$$

From the above, obtained from a large number of experiments, the diagonal elements are clearly dominant. Therefore the transition between quark belonging to same family like $u \leftrightarrow d, c \leftrightarrow s, t \leftrightarrow b$ are favoured, while transition. between different families are suppressed.

The CKM matrix doesn't represent a pure rotation like the Cabibbo matrix but includes a complex parameter. The standard parametrization (Eqs 1.5 and 1.6) uses three Euler angles $(\theta_{12}, \theta_{13}, \theta_{23},)$ and one CP-violating phase (φ). Cosines and sines of the angles are denoted c_{ij} and s_{ij}, respectively. θ_{12} is the Cabibbo angle.

$$V_{CKM} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & c_{23} & s_{23} \\ 0 & -s_{23} & c_{23} \end{pmatrix} \begin{pmatrix} c_{13} & 0 & s_{13}e^{-i\varphi} \\ 0 & 1 & 0 \\ -s_{13}e^{i\varphi} & 0 & c_{13} \end{pmatrix} \begin{pmatrix} c_{12} & s_{12} & 0 \\ -s_{12} & c_{12} & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{1.5}$$

$$\Rightarrow \begin{pmatrix} c_{12}c_{13} & s_{12}c_{13} & s_{13}e^{-i\varphi} \\ -s_{12}c_{23} - c_{12}s_{23}s_{13}e^{-i\varphi} & c_{12}c_{23} - s_{12}s_{23}s_{13}e^{i\varphi} & s_{23}c_{13} \\ s_{12}s_{23} - c_{12}c_{23}s_{13}e^{i\varphi} & -c_{12}s_{23} - s_{12}c_{23}s_{13}e^{i\varphi} & c_{23}c_{13} \end{pmatrix} \tag{1.6}$$

But because the CKM matrix is close to $\mathbb{I}_3$, is it reasonable to expand it in powers of $\lambda = V_{us}$, so according to Wolfenstein parameterisation we can rewrite the matrix at order $\mathcal{O}(\lambda^4)$ in the form described by Eq. 1.7[59]:

$$V_{CKM} = \begin{pmatrix} 1 - \lambda^2/2 & \lambda & A\lambda^3(\rho - i\eta) \\ -\lambda & 1 - \lambda^2/2 & A\lambda^2 \\ A\lambda^3(1 - \rho - i\eta) & -A\lambda^2 & 1 \end{pmatrix} + \mathcal{O}\left(\lambda^4\right) \tag{1.7}$$

where:

$$A, \lambda > 0 \tag{1.8}$$
$$\lambda = \sin\theta_{12} = \sin\theta_C \tag{1.9}$$
$$A\lambda^2 = \sin\theta_{23} \tag{1.10}$$
$$A\lambda^3(\rho - i\eta) = \sin\theta_{13}e^{-i\varphi} \tag{1.11}$$

In this parametrization θ_{ij} are three Cabibbo-like angle, while $e^{i\varphi}$ is a complex phase which encodes the CP violation.

The unitarity of the CKM matrix imposes six conditions

$$\sum_{i=u,c,t} |V_{ij}|^2 = 1 \qquad j = d, s, b \tag{1.12}$$
$$V_{ud}^* V_{ub} + V_{cd}^* V_{cb} + V_{td}^* V_{tb} = 0 \tag{1.13}$$
$$V_{ud}^* V_{us} + V_{cd}^* V_{cs} + V_{td}^* V_{ts} = 0 \tag{1.14}$$
$$V_{us}^* V_{ub} + V_{cs}^* V_{cb} + V_{ts}^* V_{tb} = 0 \tag{1.15}$$

The three vanishing combinations can be represented as triangles in the complex plane (Fig:1.1).

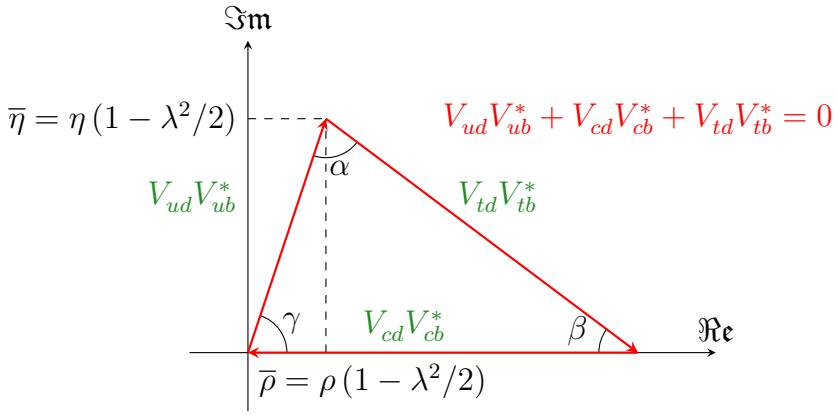


Figure 1.1: Unitary triangle defined by relation 1.13.

1.3 The $K^+ \to \pi^+ \nu\bar{\nu}$ decay

According to the Standard Model, the process $\bar{s} \to \bar{d}\nu\bar{\nu}$ is due to one-loop contributions (two "penguin" and one box diagram) as shown in Fig.1.2.

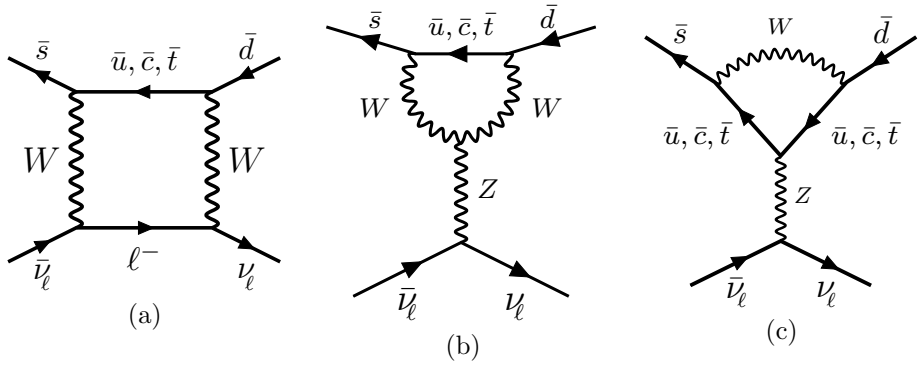


Figure 1.2: A W-box (a) and two Z-penguin (b, c) diagrams. These are the one-loop Feynman diagrams contributing to the $\bar{s} \to \bar{d}\nu\bar{\nu}$ process in SM.

The effective Hamiltonian for $K^+ \to \pi^+ \nu\bar{\nu}$ can be written as [4]:

$$H_{eff} = \frac{\alpha G_F}{2\sqrt{2}\sin^2 \pi\theta_W} \sum_{\ell=e,\mu,\tau} (V_{cs}^* V_{cd} X_c^\ell + V_{ts}^* V_{td} X_t)(\bar{s}\bar{d})(\bar{\nu}_\ell \nu_\ell) \tag{1.16}$$

where G_F, α, and θ_W are Fermi and fine-structure constants and Weinberg angle respectively. X_t is a function describing the top quark dominant contribution [4]. The X_c^ℓ functions (with $\ell = e, \mu, \tau$) encode instead the contribution of the charm quark and can be computed to next-to-next-to-leading order with an error below than 4% [38]. The last factor $(\bar{s}\bar{d})\,(\bar{\nu}_\ell \nu_\ell)$ is the $V - A$ neutral weak current.

Generally, the computation of the hadronic matrix element is the greatest source of uncertainty in weak meson decays, because low energy QCD isn't perturbative. In the case of $K^+ \to \pi^+ \nu\bar{\nu}$ the matrix element $(\bar{s}\bar{d})\,(\bar{\nu}_\ell \nu_\ell)$ can be obtained with the help of isospin symmetry from the measurement of the $BR(K^+ \to \pi^0 e^+ \nu_e)$, which has an error of 0.79%[54]. So using the effective Hamiltonian for $K^+ \to \pi^0 e^+ \nu_e$ (Eq. 1.17)[4]:

$$\mathcal{H}_{eff}(K^+ \to \pi^0 e^+ \nu_e) = \frac{G_F}{\sqrt{2}} V_{us}^* (\bar{s}\bar{u})(e^+ \nu_e) \tag{1.17}$$

and isospin symmetry (Eq.1.18)

$$\langle \pi^+ | (\bar{s}\bar{d})_{V-A} | K^+ \rangle = \sqrt{2} \langle \pi^0 | (\bar{s}\bar{u})_{V-A} | K^+ \rangle \tag{1.18}$$

We obtain, neglecting the effects due to $m_{\pi^+} \neq m_{\pi^0}$ and $m_e \neq 0$, a relation between the BRs of $K^+ \to \pi^+\nu\bar{\nu}$ and $K^+ \to \pi^0 e^+ \nu_e$[4].

$$\frac{BR(K^+ \to \pi^+\nu\bar{\nu})}{BR(K^+ \to \pi^0 e^+ \nu_e)} = \frac{\alpha^2}{2\pi^2 |V_{us}|^2 \sin^4\theta_W} \sum_{\ell=e,\mu,\tau} \left| V_{cs}^* V_{cd} X_c^\ell + V_{ts}^* V_{td} X_t \right|^2 \tag{1.19}$$

Using Eq. 1.19 and including isospin breaking corrections one obtains [4]:

$$\frac{BR(K^+ \to \pi^+\nu\bar{\nu})}{BR(K^+ \to \pi^0 e^+ \nu_e)} = \frac{3\alpha^2 r_{K^+}}{2\pi^2 \lambda^2 \sin^4\theta_W} \left\{ \left[X_t \mathfrak{Im}\left(V_{ts}^* V_{td}\right) \right]^2 + \right. \\ \left. + \left[\lambda^4 P_0 \mathfrak{Re}\left(V_{cs}^* V_{cd}\right) + X_t \mathfrak{Re}\left(V_{ts}^* V_{td}\right) \right]^2 \right\} \tag{1.20}$$

In this equation P_0 describe the total charm quark contribution

$$P_0 = \frac{1}{\lambda^4} \left(\frac{2}{3} X_c^e + \frac{1}{3} X_c^\tau \right) = 0.42 \pm 0.06 \tag{1.21}$$

under the assumption that X_c^μ and X_c^e are equal[18], $r_{K^+} = 0.901$ takes into account the isospin breaking correction to be applied in order to relate the two branching ratios.

The theoretical expectation is then:

$$BR(K \to \pi^+\nu\bar{\nu}) = \left(7.81^{+0.80}_{-0.71} \pm 0.29\right) \cdot 10^{-11} \tag{1.22}$$

Where the first uncertain is due to the error on CKM matrix elements and the second one is pure theoretical [17].
As discussed above, the measurement of branching ratio of $K^+ \to \pi^+\nu\bar{\nu}$ decay is very sensitive to new Physics: predictions for this branching ratio are available for various extensions of the SM including models with 4th generation of quarks and leptons[21], Littlest Higgs[16], supersymmetric flavour models[5] and Z' models with FCNC quark couplings[20]. If no hint of new Physics is found, this measure can be used to improve the current experimental precision of $|V_{td}|$ parameter.

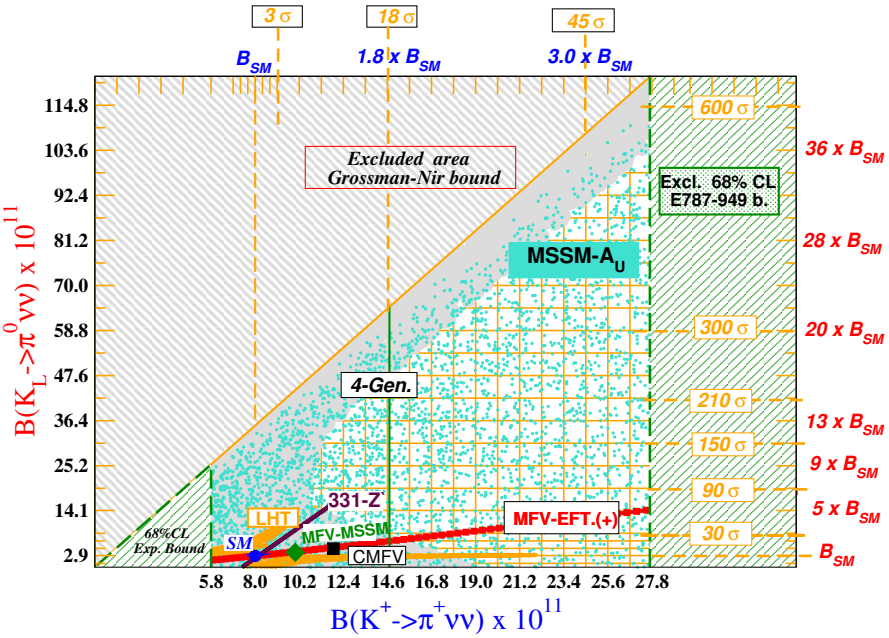


Figure 1.3: Plot illustrating $K^+ \to \pi^+\nu\bar{\nu}$ and $K_L^0 \to \pi^0\nu\bar{\nu}$ Physics sensitivities. The left and right edge of the plot are excluded by the measurement of the $B.R.$ obtained at E949 experiment[11]. The top part of the plot is excluded using the measured $B.R.$ and the Grossman-Nir relation, $B.R.(K_L^0 \to \pi^0\nu\bar{\nu}) < 4.4 \times B.R.(K^+ \to \pi^+\nu\bar{\nu})$, which is model independent[34]. The BSM theories predicting the $K \to \pi\nu\bar{\nu}$ decays include models with 4th generation quarks and leptons[21], Littlest Higgs[16], supersymmetric flavour models[5] and Z' models with FCNC couplings [20], the colored points are the results computed for the two branching ratios varying the main parameters of each model according to the physical constraint.

1.4 Previous searches for $K^+ \to \pi^+\nu\bar{\nu}$

The first search for the $K^+ \to \pi^+\nu\bar{\nu}$ decay was attempted in 1969, in a bubble chamber experiment at Argonne National Laboratory of Michigan, defining an upper limit to its branching ratio [24]:

$$BR(K^+ \to \pi^+\nu\bar{\nu}) < 10^{-4} \qquad (1969) \tag{1.23}$$

After fours years the result was improved to $BR(K^+ \to \pi^+\nu\bar{\nu}) < 5.6 \cdot 10^{-7}$ by a spark chamber experiment at the Berkley Bevatron [23], followed by an experiment at the KEK Proton Synchrotron that yielded [13]:

$$BR(K^+ \to \pi^+\nu\bar{\nu}) < 1.4 \cdot 10^{-7} \qquad (1981) \tag{1.24}$$

In 2004 at Brookhaven National Laboratory the E787 collaboration obtained the first observation based on 3 events [8]:

$$BR(K^+ \to \pi^+ \nu\bar{\nu}) = 1.47^{+1.30}_{-0.89} \cdot 10^{-10} \qquad (2004) \qquad (1.25)$$

After 5 years the follow-up experiment E949 was able to collect 4 more candidate events leading a combined result of [12, 11]

$$BR(K^+ \to \pi^+ \nu\bar{\nu}) = 1.73^{+1.15}_{-1.05} \cdot 10^{-10} \qquad (2009) \qquad (1.26)$$

which is consistent with the Standard Model expectations, within the large statistical errors.

Each of these experiments used low energy kaons stopped. NA62 will instead employ an high energy beam, studying in-flight kaon decays.

1.5 Experimental Strategy

The presence of two neutrinos and a single charged track in the final state makes NA62's goal a challenging precision measurement, requiring hermetic background rejection as well as an excellent detector system for particle identification. The signature of a $K^+ \to \pi^+ \nu\bar{\nu}$ event consists in only one charged track, and all other events with one charged track contribute to the background.

Protons from the SPS at 400 GeV/c impinge on a beryllium target and produce a secondary charged beam. Consideration about signal acceptance drive the choice of a secondary beam of 75 GeV/c. About 6% of particles in the secondary beam are K^+, the rest are π^+ and protons.

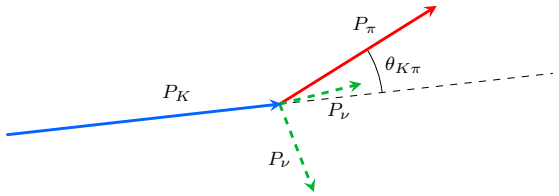


Figure 1.4: $K^+ \to \pi^+ \nu\bar{\nu}$ decay kinematics.

The only measurable physical quantity of $K^+ \to \pi^+ \nu\bar{\nu}$ are the momenta of kaon and π^+ and the decay angle between them in the laboratory frame. So it is convenient to use as discriminating variable the squared missing mass of the decay.

$$m^2_{miss} = \left(P^\mu_K - P^\mu_\pi\right)^2 = (E_K - E_\pi)^2 - \left(P^2_K + P^2_\pi - 2|P^2_K||P^2_\pi|\cos\theta_{K\pi}\right)^2 \quad (1.27)$$

where P_K and P_π are the momenta of kaon and pion, E_K and E_π their energies and $\theta_{K\pi}$ is the decay angle between them in laboratory frame.

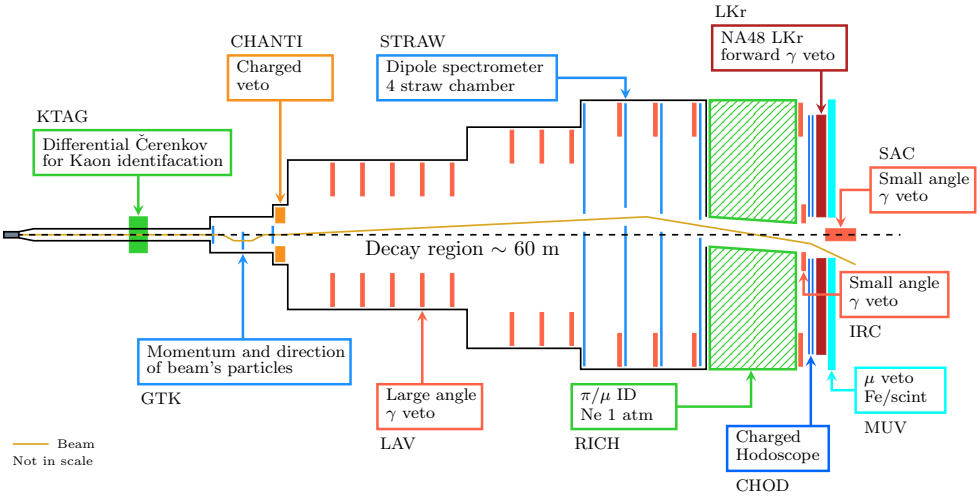


Figure 1.5: Sketch of the NA62 detector. For a brief description see Chapter 2.

m^2_{miss} is computed under the assumption that the charged track is a pion, and the it allows to separate the signal from the most important background processes as shown in Fig:1.6. The top part of Fig:1.6 shows the contribute of the largest background processes in the m^2_{miss}, the bottom part shows the one from background modes which are not kinematically constrained: these modes are the radiative version of decay channels from the previous plot and 3- and 4-body semi-leptonic decays. The fiducial signal region is split in two to exclude the $K^+ \to \pi^+\pi^0$ region, while upper and lower limits exclude $K^+ \to 3\pi$ and $K^+ \to \mu^+\nu_\mu$.

In Table 1.1 the main backgrounds are listed with their respective rejection criteria.

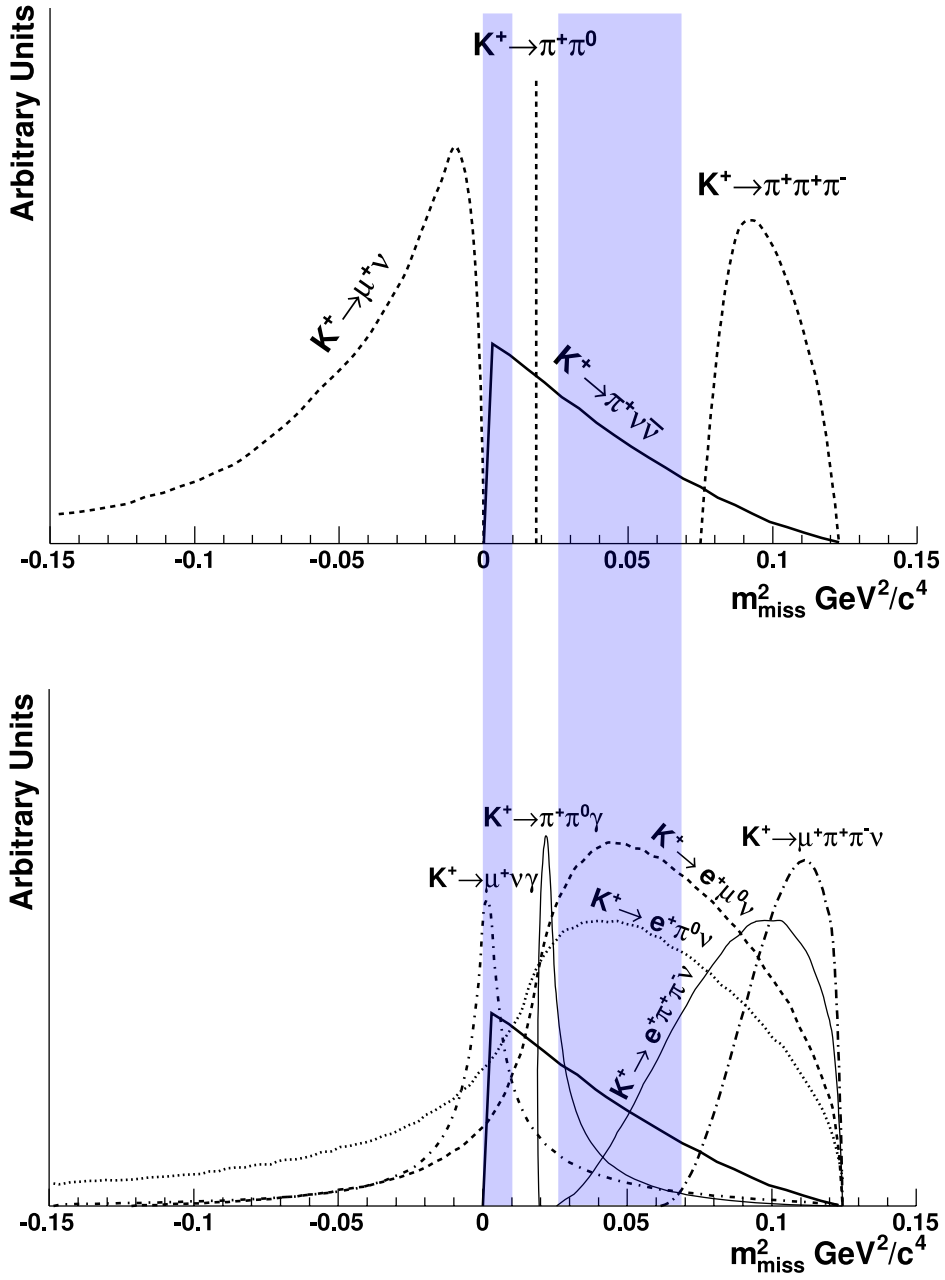


Figure 1.6: Distribution of the reconstructed squared missing mass resulting from koan decay under the hypothesis the charged track is a pion. The solid line indicates the $K^+ \to \pi^+\nu\bar{\nu}$ signal in both plots. The top figure shows the kinematically constrained backgrounds, which are also the channels with larger branching ratios; the bottom one shows the others main backgrounds for which the reconstructed missing mass overlaps the signal. The blue areas display the fiducial signal regions [37].

Decay mode	BR	Rejection
$\mu^+\nu_\mu$	63.56%	Kinematics + μ PID
$\pi^+\pi^0$	20.67%	Kinematics + γ veto
$\pi^+\pi^+\pi^-$	5.58%	Kinematics + $\pi^\pm$veto
$\pi^+\pi^0\pi^0$	1.76%	Kinematics + γ veto
$\pi^0\mu^+\nu_\mu$	3.35%	μ PID + γ veto
$\pi^0 e^+\nu_e$	5.07%	e PID + γ veto

Table 1.1: The main backgrounds for the NA62 experiment with their rejections strategies.

Chapter 2

The experimental setup

Contents

The NA62 experiment is located at the CERN-SPS North-Area, on the beam line originally used by NA48 experiment. The NA62 collaboration devised a new experimental apparatus on the basis of the NA48 experience [50].

An unseparated 750MHz beam composed of protons,pions and kaons is produced in collisions of a 400 GeV/c proton beam on beryllium target. The decay region starts around 100 m from the target and ends 65m downstream, where the main detectors are located.

There are 3 sub-detectors located upstream of the decay region. The beam kaons are identified by the CEDAR, a Čerenkov differential counter. The

GigaTracker (GTK) is a beam spectrometer composed by 3 stations of silicon micro-pixels which provide a measurement of momentum, direction and time of beam kaons. Finally the CHANTI (CHarged ANTI counter), made by plastic scintillators vetoes large-angle charged particles due to inelastic beam interactions with collimator or upstream materials.

The LAV (Large Angle photon Vetoes) are 12 annular stations made of lead glass crystals situated between 120 m and 240 m along the decay line axis, they provide photon rejection in a $8.5 \div 50$ mrad cone around z-axis. The straw chamber spectrometer (STRAW) and dipole magnet placed between the second and third STRAW chamber, provides a measurement of decay vertex, direction of flight and momentum of decay products. The RICH detector provides π/μ separation in the $15 \div 35$ GeV/c moemntum region.

A segmented plastic scintillator hodoscope (CHOD) provides a fast trigger signal for charged particles just after they emerge from the RICH vessel. Downstream the CHOD detector there are three calorimeters for small-angle photon rejection. The LKr (Liquid Krypton) calorimeter inherited from the NA48 experiment. An intermediate ring- shaped calorimeter (IRC) and the small-angle electromagnetic calorimeter (SAC) add photon suppression in the region not covered by the geometric acceptance of the LKr.

Suppression of $K^+ \to \mu^+\nu_\mu$ is done with MUV3 (muon-veto) a scintillator system which detects muon after a 80 cm iron block. Two additional muon-veto stations (MUV1 and MUV2) are used to distinguish muons from hadron by the energy released by hadronic showers [37].

A longitudinal view of the NA62 experiment is shown in Fig. 2.1.

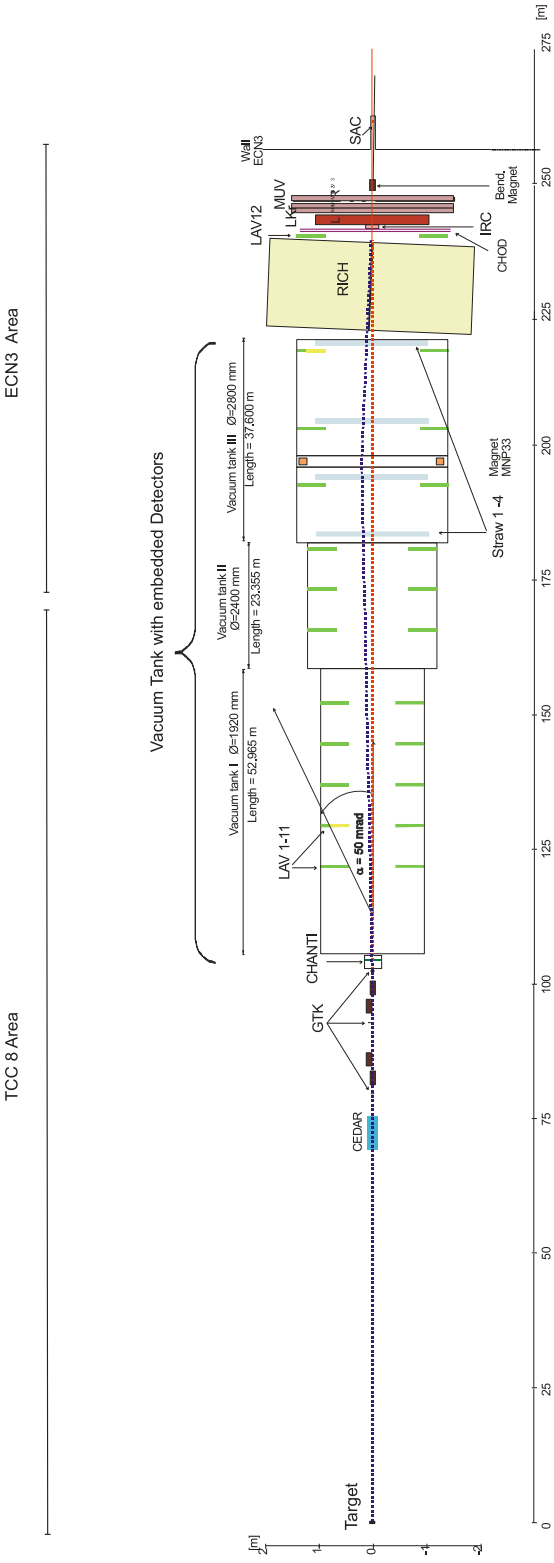


Figure 2.1: Schematic view of NA62 experiment

2.1 Beam

It is convenient to use a high energy proton beam in order to maximize the production of positive kaons by beam interactions on a beryllium target[14].

The highest kaon production is achieved at $P_K/P_p \simeq 0.35$, where P_p is the central proton beam momentum and P_K is the momentum of the produced kaons. Furthermore, the use of high energy kaons increases the detection efficiency of most sub-detectors. Due to these considerations, a central beam momentum 75 GeV/c.

The choice of positive kaons is due to the ratio particles abundances in a hadron beam produced by 400 GeV/c protons [37]:

$$\frac{K^+}{K^-} \simeq 2.1 \tag{2.1}$$

$$\frac{K^+/\pi^+}{K^-/\pi^-} \simeq 1.2 \tag{2.2}$$

Tab. 2.1 shows the different components of the beam.

Momentum	75 ± 0.9 GeV/c
Rate	750 MHz
Composition	70% π^+ 23% p^+ 6% K^+ 1% other

Table 2.1: NA62 beam composition

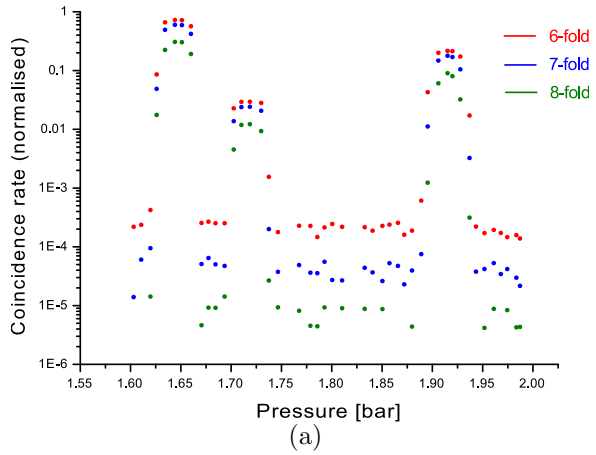
2.2 Detectors upstream of the decay region

In this section the detectors used to track the beam are described, they are the CEDAR, a differential Čerenkov used to tag the kaon in the beam, the GigaTracker (GTK) used to measure the angle and momenta of beam particles and the CHANTI for the rejection of charged particles with angle between 28.5 mrad and 1.38 rad.

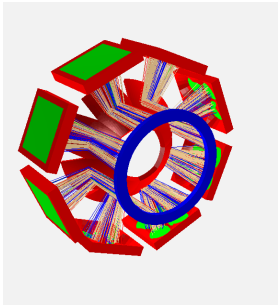
2.2.1 CEDAR.

One disadvantage of high-energy beams is that kaons cannot be efficiently separated from other beam particles. So the detection of kaons before decay is a crucial aspect for the expeirment. Kaon tagging is achieved by letting the beam traverse a differential Čerenkov counter (CEDAR). The detector at this time is filled with nitrogen at pressure of 1.7 bar, and has a total thickness of $4X_0$.

A particle crossing a radiator with refractive index n at a velocity βc emits a cone of Čerenkov light at an angle $\theta_c(\beta, n)$. Since the momentum of the beam is known, the Čerenkov angle, at a fixed gas pressure and therefore fixed n, is a function of the mass of the particle. The gas pressure is therefore adjusted so that only the desired particle type can emit Čerenkov radiation at the chosen light detection angle. A layout of the CEDAR is shown in Fig 2.2(b).



(a)



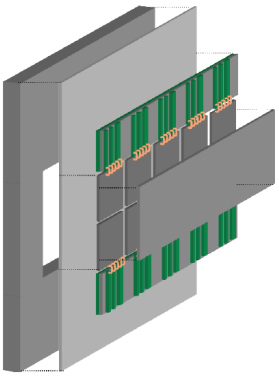
(b)

Figure 2.2: Left figure: a pressure scan on a 75 GeV/c beam shows three peaks corresponding respectively to pions, kaons and protons, the detector was filled N_2, the three different color are for different coincidences request in the eight light spots. Right figure: a view of CEDAR.

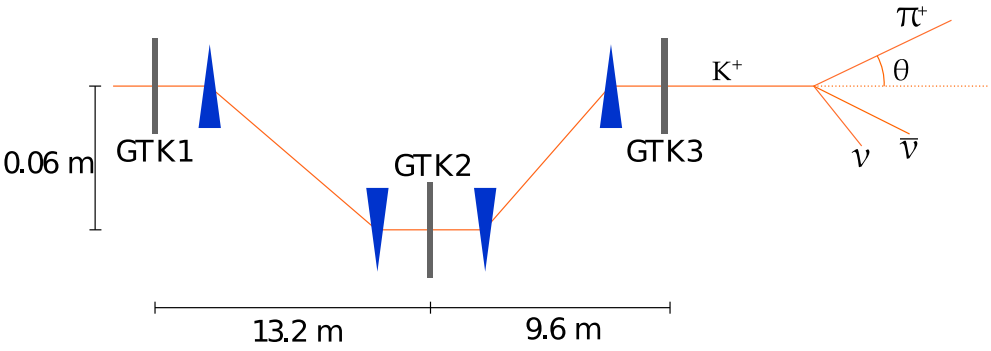
2.2.2 GTK

The GigaTracker detector provides precise measurement of angle, momentum and time of the crossing particle.

In order to limit hadronic interactions and to preserve the beam divergence, the GigaTracker is composed of three station (3 is the minimum number of station to have a spectrometer) for a total thickness less than $0.5X_0$ [37]. Each station contains 18000 300×300 μm^2 silicon micro-pixels 200 μm thick, bump-bonded to 10 readout ASIC chips 100 μm thick. The three stations of the GTK are mounted inside the vacuum tank preceding the decay region, and they are interlaced with 4 achromat magnet pairs as shown in Fig 2.3.



(a)



(b)

Figure 2.3: Top, sketch of a GigaTracker station[25]. Bottom, layout of the GigaTracker stations and magnets used to bend the beam[57].

2.2.3 CHANTI

The reduction of accidental backgrounds to a level of 10^{-11} is a crucial point for the experiment. The purpose of the CHANTI is to detect charged particles due to inelastic interactions between beam and collimator or upstream material at an angle larger than that allowed for the beam as they emerge from the last GigaTracker station. The CHANTI is made of six double-layer stations [37]. Each station is a 30×30 cm^2 square with a 90×50 mm^2 hole to allow the passage of the beam and each layer is composed of 24 (22) scintillator bars aligned to the x axis (y axis). A sketch of the CHANTI is shown in Fig 2.4.

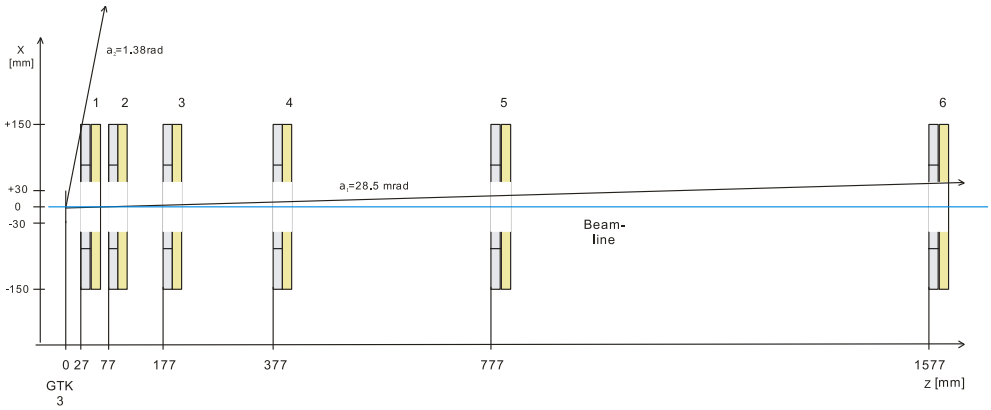


Figure 2.4: Sketch of CHANTI stations on the beam line [37].

2.3 Detectors downstream of the decay region

The downstream detectors have been designed to detect K^+ decay products.

2.3.1 Photon veto system

In order to efficiently reject the photons originating from $K^+ \to \pi^0\pi^+$ a photon veto system was developed, that ensures a rejection inefficiency lower than 10^{-7}. The photon veto detector cover a 50 mrad angular range around the beam.

The photon veto system is composed by four sub-detectors that cover different angular region between $0 \div 50$ mrad

- Twelve LAV stations cover the angular region between 8.5 and 50 mrad.

- The LKr calorimeter covers angles between 1 and 8.5 mrad.
- SAC and IRC cover the inner region, from about 0 to 1 mrad.

The Large Angle Veto detector reuses 2496 $10 \times 10 \times 37$ cm^3 leadglass blocks from the OPAL electromagnetic calorimeter, arranged in 12 annular stations.

Of the 12 LAV counters, 11 are placed inside the $3 \cdot 10^{-7}$ mbar vacuum tank hosting the decay region, and one is placed between the RICH and the CHOD sub-detectors.

Photons incident into LAV blocks produce electromagnetic showers, detected through the collection of Čerenkov light emitted by e^+e^- pairs. Due to its low threshold, the LAV system can also detect muons and pions in the beam halo. A schematic view of one of the LAV stations is shown in Fig. 2.5.

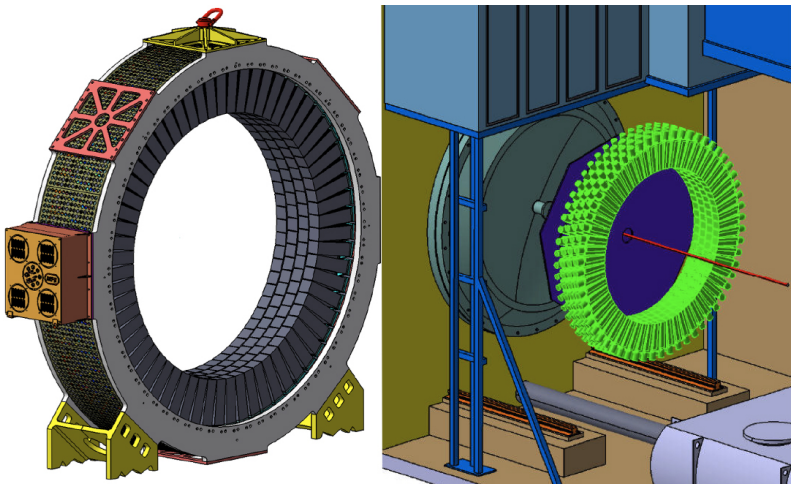


Figure 2.5: A layout of LAV 12 station.

The two small-angle veto calorimeters, IRC and SAC, are *"shashlik"* type calorimeters, i.e. detectors made of lead absorber layers with plastic scintillator plates used as active material.

The IRC is placed around the beam line in front of the LKr, and covers the angular region between LKr and the SAC. A dipole magnet bends the beam so that charged particles cannot hit the SAC, the most forward detector in the NA62 setup.

The Liquid Krypton Calorimeter, placed between RICH and the MUV detectors, is the same as in the NA48 experiment. Its main purpose is to reject

photons between 1 and 8.5 mrad. The LKr also provides accurate measurement of the energy of electrons and positrons, useful to reject $K^+ \to \pi^0 e^+ \nu_e$ background.

2.3.2 STRAW

The purpose of the STRAW magnetic spectrometer is to measure the directions and momenta of kaon decay products. The kinematical constraint needed to reject most of the background requires an accurate reconstruction of the secondary particles tracks.

The full spectrometer consists of four straw chambers. A dipole magnet, placed between the second and the third chamber, generates a vertical field of 0.36 T, corresponding to a kick of 270 MeV/c along the x-axis. Each chamber is composed of four "views" (x, y, u and v). Each view is made of 256 straw tubes. Fig. 2.6 shows the four views of a STRAW chamber.

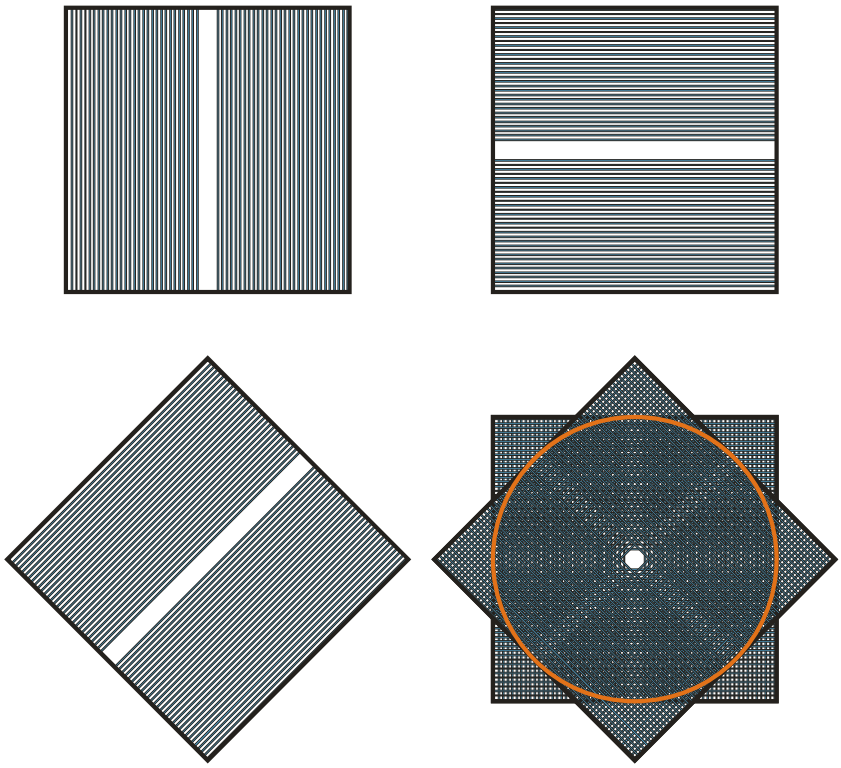


Figure 2.6: The four view of a STRAW chamber [37]. In the bottom right corner the four view are super-imposed.

2.3.3 RICH

The purpose of the RICH detector of NA62 is to separate pions from muons in the momentum range between 15 and 35 GeV/c, in order to suppress the main background $K^+ \to \mu^+\nu_\mu$ by a factor 10^{-2}. Moreover this detector provides Level 0 trigger primitives for charged tracks. Due to the primary importance of RICH in this thesis work, I'm going to describe it in more detail. First I describe in general terms how a RICH detector works, then I illustrate the setup of the NA62 RICH.

Principle of RICH detector

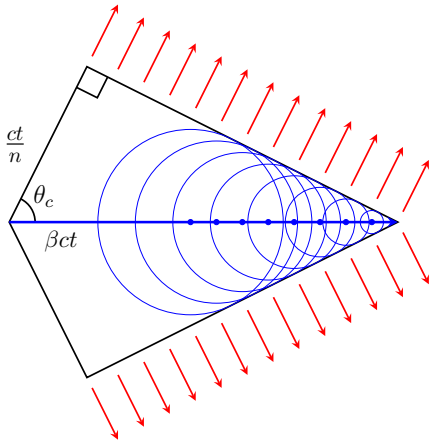


Figure 2.7: Geometry of Čerenkov radiation.

The principle of a RICH detector are shown in Fig. 2.7.When a particle goes through a medium at velocity $\beta = v/c > 1/n$, where n is the refractive index of the medium, it emits Čerenkov light at an angle θ_c relative to the particle trajectory, such that :

$$\cos\theta_c = \frac{1}{n\beta} \tag{2.3}$$

thus forming a *Čerenkov cone*. From Eq. 2.3 follows that a velocity threshold β_{th} exists below which no radiation is emitted:

$$\beta_{th} = \frac{1}{n} \qquad (\theta_c = 0) \tag{2.4}$$

while the maximum angle of emission is achieved for $\beta \to 1$

$$\cos\theta_{max} \to \frac{1}{n} \qquad (\beta \to 1) \tag{2.5}$$

From Eq. 2.4 we obtain the threshold momentum P_{th} for a particle of mass m to emit Čerenkov radiation:

$$P_{th}(m) = \frac{m}{\sqrt{n^2 - 1}} \tag{2.6}$$

The light is projected on the focal plane, perpendicular to the beam direction, of a spherical mirror of focal length f. For particles travelling parallel to the beam line, the resulting image on the focal plane is a ring of radius

$$r_c = f \tan \theta_c \tag{2.7}$$

while, for particles traveling at an angle θ_c to the beam line, the same image appears shifted by a distance:

$$d = f \tan \theta \tag{2.8}$$

from the focus. Fig. 2.8 shows a sketch of a basic RICH detector.

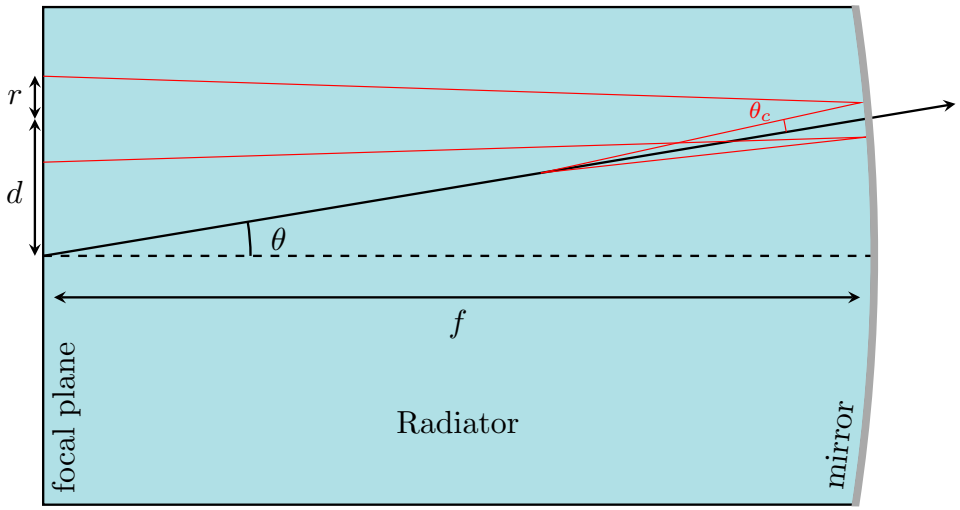


Figure 2.8: Draft of a simple RICH detector. The radius (r) of the circle in the focal plane is determined by the velocity of the particle, while the position (d) of its center depends on the particle direction.

Larger rings on the focal plane correspond to particles crossing the RICH radiator volume at a larger velocity (keeping the type of particle fixed). So if the beam momentum is known, the radius of the Čerenkov ring can be used to compute the mass of the particle based on the mass found according to Eq.2.10

$$P(r_c) = m\frac{\sqrt{f^2 + r_c^2}}{\sqrt{r_{max}^2 - r_c^2}} \simeq \frac{mf}{\sqrt{r_{max}^2 - r_c^2}} \tag{2.9}$$

$$\Rightarrow m(P, r_c) = \frac{P}{f}\sqrt{r_{max}^2 - r_c^2} \tag{2.10}$$

where $r_{max} = f\sqrt{n^2 - 1}$ and $P(r_c)$ is the momentum for particle of mass m and Čerenkov radius r_c.

NA62 RICH

The RICH detector is positioned between the last STRAW chamber and the CHOD. The vessel, a 18 m long a 2.8 m wide cylinder, is filled with neon at atmospheric pressure. The refractive index n is such that $(n-1) \simeq 62 \cdot 10^{-6}$ [37] corresponding a threshold momentum for pions of $P_{th} = 12.1$ GeV/c. A layout of the RICH is shown in Fig. 2.9.

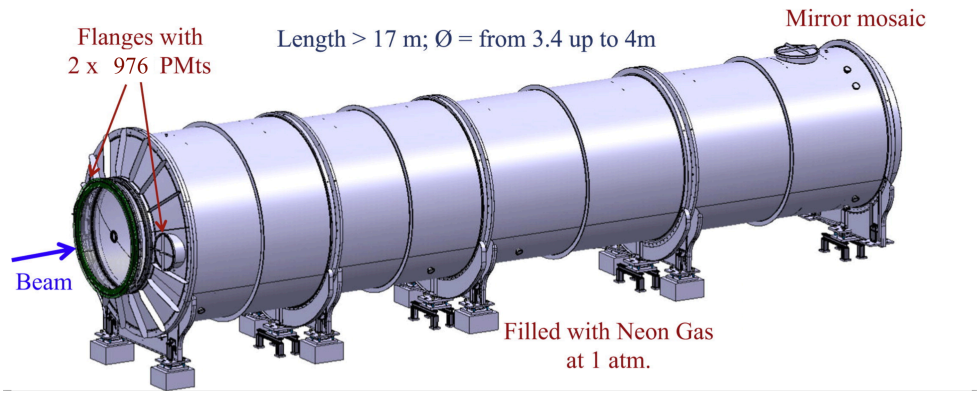
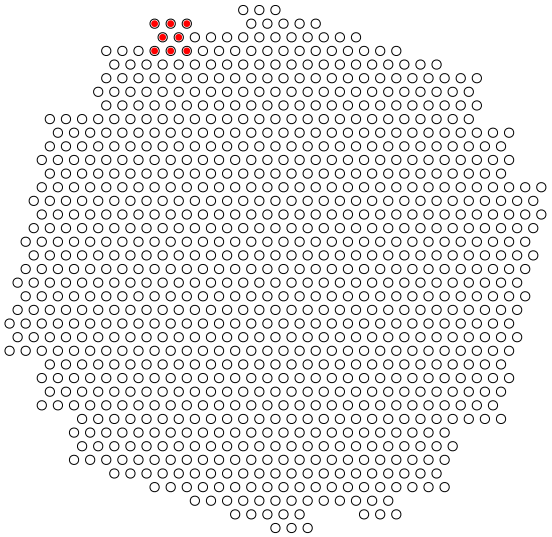


Figure 2.9: Layout of RICH detector[37].

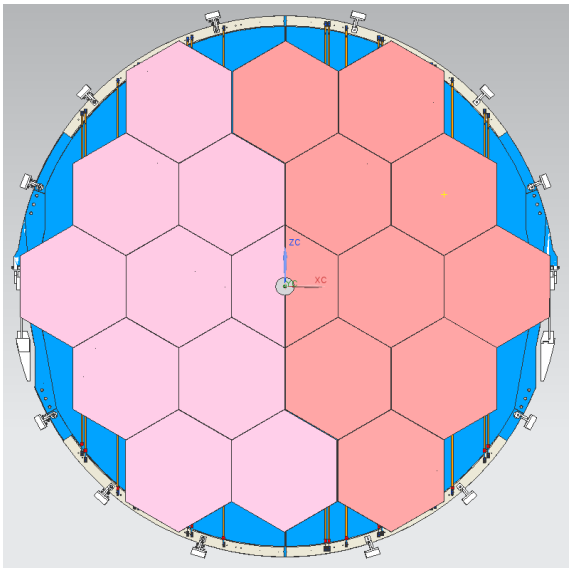
A mosaic of 18 hexagonal and 2 semi hexagonal mirrors made of aluminum-coated 25 mm thick glass covered with a thin dielectric film, with sides 35 cm long arranged to form a spherical mirror, reflects the Čerenkov cone onto the RICH focal plane. In order to avoid absorption of light by the beam pipe, the mirrors actually form two independent spherical surfaces (as shown in Fig. 2.10(b)), with the foci corresponding respectively to the two PMT flanges[1], each one with 976 PMT, Fig. 2.10(a) shows a flange with a SuperCell (digital

[1]In the following, the left and right sides will be often referred to as Jura and Salève flanges respectively, after the two mountains overlooking the Geneva area.

OR of 8 PMTs) in eveidence. The mirror curvature radius is 34 m which result in a nominal focal length $f = 17$ m.



(a)



(b)

Figure 2.10: Fig.2.10(a) shows a flange with 976 PMTs; Fig.2.10(b) shows the 18 hexagonal and 2 half-hexagonal mirrors of RICH detector.

RICH geometric corrections

As said the photons emitted by the charged particles are reflected onto two different flanges. In order to achieve the largest efficiency as possible for the π^+ from the $K^+ \to \pi^+\nu\bar{\nu}$ decay, the mirrors are rotate with respect to the vessel axis of different angles.

If no rotation is applied the center $C(0,0)$ of the two flanges coincide with the vessel axis $O(0,0)$. But if a rotation it's applied to one or both the mirrors, this is no longer true. The two rotations move the flanges in two frames different from the original, so to reconstruct correctly the events these tilts need to be considered. A shift of the two flange can correct the effect of the rotation, and after the shift the two flange are moved back to the original frame. To explain how the shift was applied, let φ and φ' be the Jura and Salève mirrors angles to the vessel along x axis respectively, $f = R/2$ is the focal length of the RICH, R is the curvature radius of the mirrors. I shift the x coordinate of the flange centers of the hit by a quantity φf or $\varphi' f$ depending on which spot is illuminated (at this time no corrections are needed for the y coordinate):

$$x \equiv \begin{cases} x - f\varphi & 0 \leq \text{ChannelID} < 976, \text{ for Jura side} \\ x - f\varphi' & 977 \leq \text{ChannelID} < 1952, \text{ for Salève side} \end{cases}$$

where $f\varphi$ and $f\varphi'$ are 127 mm and 177 mm respectively. Figure 2.12 shows an event seen by the two flanges before and after the correction are applied.

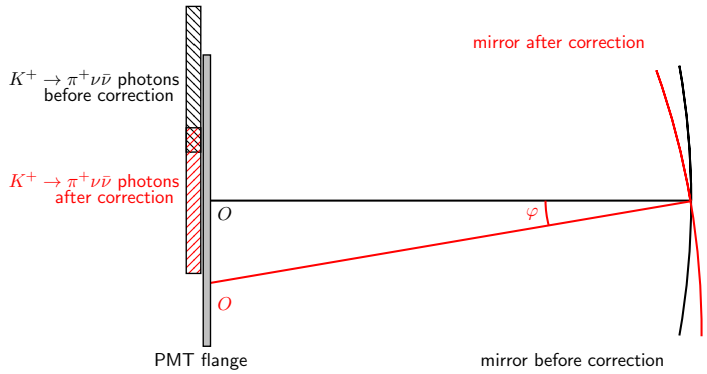
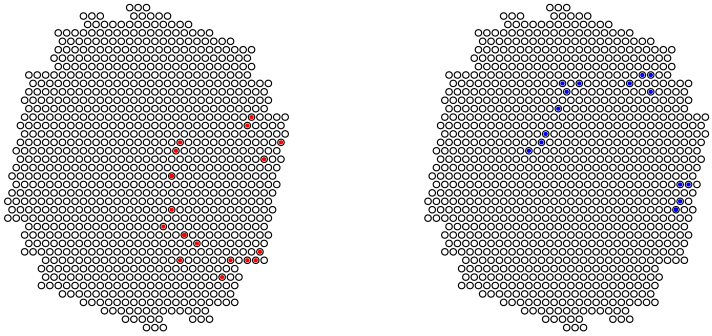
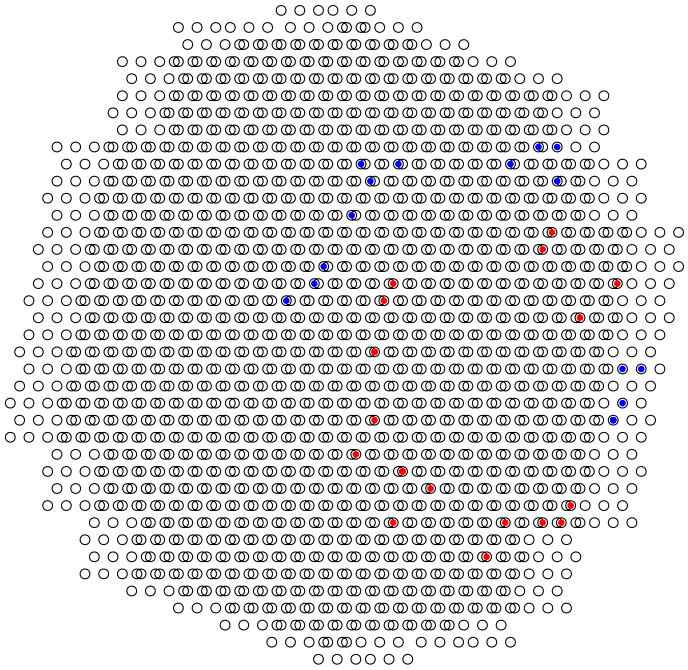


Figure 2.11: Before the correction the vessel axis is aligned with the center of the flange, but in this way the Čerenkov photons from $K^+ \to \pi^+\nu\bar{\nu}$ will be mostly reflected on the mirror edge. After the correction light from $K^+ \to \pi^+\nu\bar{\nu}$ is collect around the central zone of the flange. After this rotation, the position of the PMTs need to be changed to make coincide the $(0,0)$ of the PMTs map with the vessel axis.



(a)



(b)

Figure 2.12: An event seen by the RICH, before and after the geometric correction. The red dots hits came from the Jura side and the blue ones came from the Salève side.

The RICH is also used in to the Level 0 trigger, producing a primitive each time a charged particle crossing its volume, with the time resolution of 100 ps. The RICH role in the L0 trigger is explained in more detail in section 2.3.6 .

2.3.4 CHOD

A plastic scintillator hodoscope provides a fast signal to trigger data acquisition on the passage of a charged particle. The CHOD inherited by the NA48 experiment, is composed of two planes of 64+64 plastic scintillator bars aligned respectively to the x and y directions.

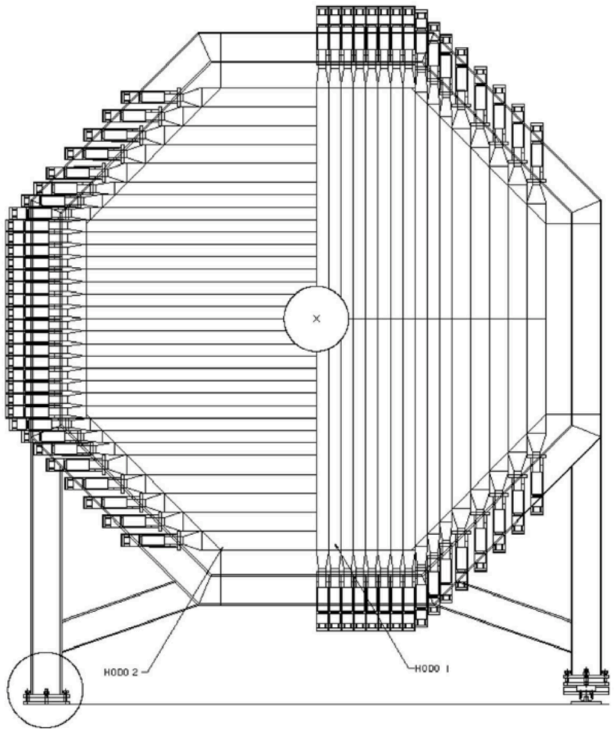


Figure 2.13: Schematic view of CHOD detector.

Because the age and the geometry of the CHOD, for the 2016 run a New CHOD detector was built: the New CHOD will take data together with the CHOD during the run. The new detector is made of 148 plastic scintillator tiles of variable dimension according to the distance from the beam, so that the particle rate is below 500 kHz in each tile[42].

2.3.5 MUV

For a further muon suppression in addition to that achieved by the RICH, a two stage muon veto system was built[37]:

1. A fast muon veto detector (MUV3),with a time resolution below 1 ns, rejects events in coincidence with GTK and CEDAR detectors. This module is placed downstream of an 80 cm thick iron wall and is used in the fast Level 0 trigger.

2. Two segmented hadronic calorimeters (MUV1 and MUV2) identify crossing particles depositing a significant amount of energy. These two modules are composed of alternate layers of scintillator (10 mm thick) and iron (25 mm thick). The total thickness of each module is 62.5 cm.

Fig. 2.14 shows a sketch of the three muon veto stations.

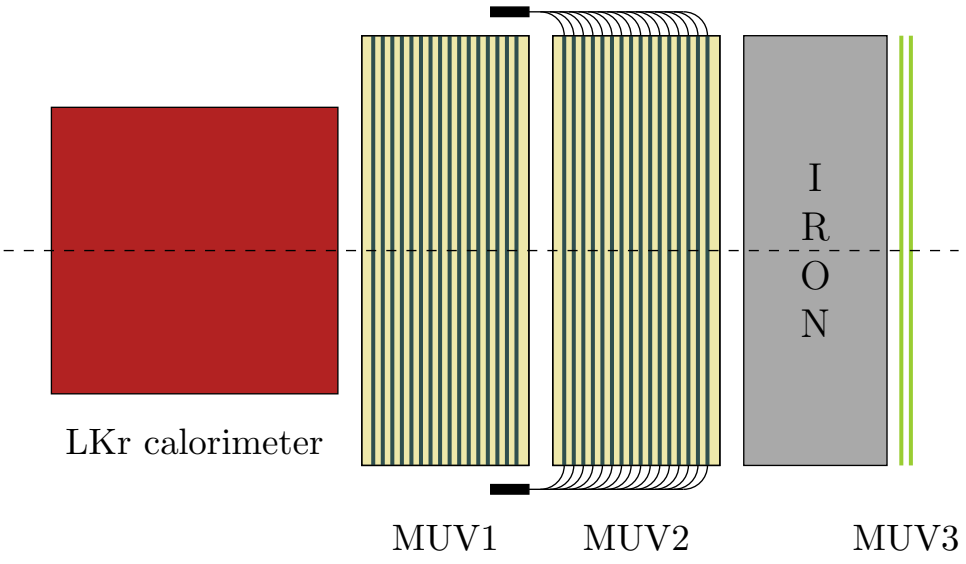


Figure 2.14: Layout of the three MUV stations.

2.3.6 TDAQ

The high rate of events and the presence of 12 sub-detectors results in a high output data rate that it is impossible to store on disk without filtering. A multi levels Trigger and Data Acquisition (TDAQ) system is therefore needed, which should identify the events to be saved and reject the rest. The building block of the TDAQ system for many detector is a common general-purpose integrated trigger and data acquisition board developed in Pisa, nicknamed TEL62 (Trigger ELectronics for NA62) [7].

The NA62 trigger is made of three logical levels:

L0: a hardware trigger, based on the input from few sub-detectors. Rate reduction from 10 MHz to 1 MHz, with a maximum latency of 1 ms.

L1: a software trigger, based on information computed independently by each sub-detector system. Rate reduction from 1 MHz to 100 kHz.

L2: a software trigger, based on assembled and partially reconstructed events, in which informations from different from sub-detectors are used. Rate reduction from 100 kHz to about 15 kHz.

L0 Hardware Trigger

The hardware L0 trigger will be mainly based on input from the CHOD, the MUV3, the LKr, the RICH and the LAV. These detectors will continuously evaluate their incoming data for the fulfillment of certain condition (called *primitives* in TDAQ) and associated time. Trigger primitive and data will be packed in Multi Trigger Packet Format (MTP)[36] and sent through standard ethernet links to L0 Trigger Processor (L0TP).

The L0TP time-matches different sub-detectors primitive checking if L0 trigger conditions have been satisfied, in case of positive response, L0TP will issue a L0 trigger.

L0 Trigger for the RICH

Because my work is focused on the RICH L0 trigger, I'm going to describe in more detail how the L0 RICH trigger works. The standard RICH L0 trigger is based on hits multiplicity or SuperCell (digital OR of 8 PMTs) multiplicity.

In case hits multiplicity is used four TEL62 are needed to fully cover the 1952 the RICH channel. Each TEL62 receives data only from half flange, Figure 2.15 shows the different area assigned at each board.

Instead if SuperCells multiplicity is used, a fifth TEL62 is needed.The fifth board receives the data from RICH SuperCells (digital OR of 8 channels), so only 244 readout channel are needed.

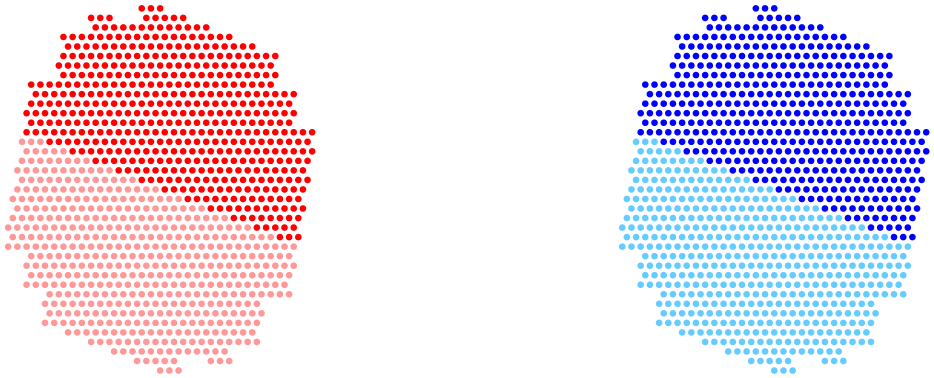


Figure 2.15: The Figure shows how RICH channels are divided between four TEL62s, two dedicated to the Jura side flange (blue ones) and two to the Salève side flange (red ones).

The trigger primitive of the RICH is based on SuperCells multiplicity and encoded in the way described below:

R1 number of SC hits between $2 \div 9$;

R2 number of SC hits between $9 \div 33$;

R3 number of SC hits between $33 \div 59$.

RS is the single ring trigger and is R1 OR R2;

RM is the multi ring trigger R2 OR R3;

MB for minimum bias trigger R1 OR R2 OR R3.

This thesis work describes study for a possible alternative L0 trigger for the RICH which use GPUs. In this case hits from the 2 flanges are needed, so the hits from four TEL62 need to be merged in order to be analyzed from GPUs, I will explain in more detail this aspect in section 4.1.2.

L1/L2 Software Trigger

After a positive L0 trigger, all sub-detectors data are moved to PCs for processing. If the L1 trigger condition are fulfilled, each sub-detector sends a L1 trigger primitive to the L1 Trigger Processor PC.

The L1 Trigger Processor will match these primitive and issue a L1 decision, at which time the data will be further processed(in case of a positive L1) or discarded (in the case of a negative L1 verdict).

By correlating information between different sub-detectors, the events will be partially reconstructed and made available for the L2 trigger decision. All data satisfying the L2 trigger condition will be saved to disk. While In case L2 conditions are not satisfied, the data will be deleted.

Part II

GPU

Chapter 3

Use of GPUs in trigger

Contents

3.1 Reasons for a GPUs trigger

In the last few years there was increased interest in the use of commercial graphic board (*Graphics Processor Unit*) for scientific computation. While these are mainly developed for video games, due to their parallel data computing and high performance GPUs are now used in various scientific fields such as medicine, chemistry, theoretical and experimental physics. This new branch of research where GPUs are used for non-graphics applications is referred as GPGPU (General Purpose computing on GPU).

A High Energy Physic (HEP) experiment is naturally parallelizable because the independence of each event and the same set of instructions to perform in order to analysing it. The possibility of use of GPUs in High Level Trigger to improve the performance achieved in the selections of interesting events and rejection of background in the online analysis is fascinating.

The main advantages are:

- higher performance achieved by GPUs with respect to CPUs in computing (last generation of video cards provide a computing power exceeding a Teraflop 3.1(a))
- larger memory bandwidth of GPUs with respect to CPUs Fig(3.1(b))
- scalability: multi-GPU systems are possible, in which the computation of data are shared between 2 or more GPUs, to have better time performance
- versatility: GPUs can be easily programmed for different purpose, using the most popular programming language, thanks to devised toolkit
- thanks to the use of commercial components, designed for sectors with a very large market, this solution appears to be very cheap with respect to other solutions based on specialized hardware, in terms of cost and human work
- a system based on GPUs benefits directly by the continuous technological progress required from the video games and image processing industry, so upgrade can be easily achieved changing old and outperformed GPUs with new ones

Some HEP experiments such as ATLAS[41] and LHCb[15] are exploring the advantages of using GPU in their *High-Level Triggers* (HLT), in order to perform an online faster analysis.

The NA62 represents a feasibility test to integrate the GPUs in the lowest level trigger, which is more challenging, due to the high rate to sustain and small latency requirement with respect to High Level Trigger.

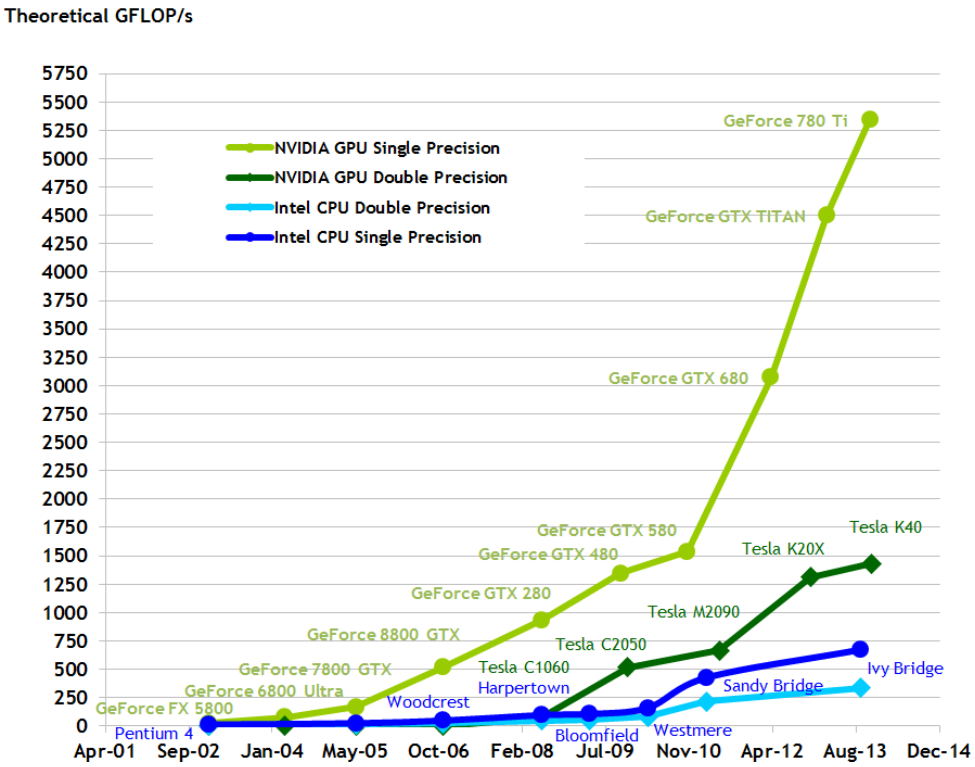
Because the NA62 was designed to have the largest acceptance and the greatest efficiency for $K^+ \to \pi^+\nu\bar{\nu}$ decay, the available bandwidth is fulfilled with data both main trigger and contral sample decay selected with the CHOD, MUV3, LKr, LAV and RICH, taken for the selection of the $K^+ \to \pi^+\nu\bar{\nu}$, and there is no much bandwidth available for collect data for the searches of other interesting decays. So the use of GPUs can in principle be useful in various way

- discarding the $K^+ \to \pi^+\pi^+\pi^-$ component in $K^+ \to \pi^+\nu\bar{\nu}$ selection, making a cut based on the number of rings in the RICH detector;
- discarding in the single-ring events the ones coming from to $K^+ \to \pi^+\pi^0$ using the closed kinematics of the decay;

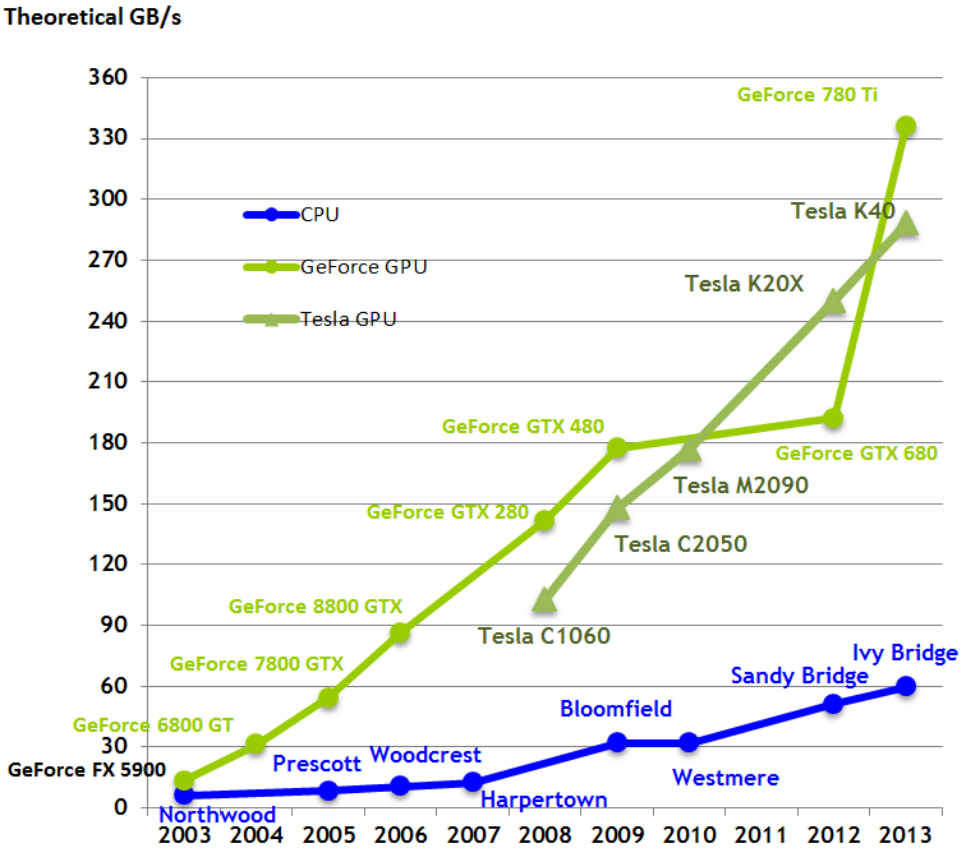
- select only the interesting multi-rings events, basing on the rings radius distributions .

Basically the GPUs can be useful in two way: discarding the background and make more bandwidth available or select more efficiently the events to save.

Moreover the future HEP experiment will have an increased amount of data, with respect to the actual ones. The amount of data collected at lowest level trigger will increase by a factor 1000/10000 and a it's impossible that a similar technologic improvement in data links bandwidth is achieved at the same time. So the computing stage needs to be moved necessarily at the first level trigger and the study on GPUs made at NA62 will be an useful starting point.



(a) Difference performances achieved in computing for CPUs and GPUs.



(b) Differences in memory bandwidth for the CPU and GPU.

Figure 3.1: Differences in performances 3.1(a) and memory bandwidth 3.1(b) between GPUs and CPUs.

3.1.1 NA62 GPU trigger

In a standard trigger system for a high energy physics experiment, the complexity in primitive generation and trigger decision is limited by the time available as defined by latency requirements. Usually in trigger levels with fixed small latency, the trigger primitives are quantities related to multiplicity and hit patterns. The trigger decision is defined with rough conditions, not allowing high rejection factors and selection power.

In many cases the definition of trigger primitives can be reduced to pattern recognition issues. This is the case for charged particle track identification in magnetic spectrometers, trajectories in silicon strip trackers or photon rings in Čerenkov detectors. The RICH detector in the NA62 experiment falls into this last category.

A project is being developed within the NA62 collaboration, which aims to

integrate GPUs into the lowest-level trigger for the first time in High Energy Physics[46]. The use of GPUs in such a hard real-time system has not been attempted so far, but it looks a realistic and challenging possibility. The online use of GPUs would allow the computation of complex trigger primitives at the L0 trigger level, with resolution comparable to offline analysis. There are two different way to insert GPUs in the NA62 context.

The first option is also the more challenging, and the GPUs perform two different works Fig 3.3:

- compute the data received from the TEL62 boards (see sec. 2.3.6) of the all the detector participating L0 making high quality primitives (rings, clusters, tracks)
- substitute the L0TP (see sec. 2.3.6), matching the primitives create in the previous first step and issue a trigger decision

In any case a FPGA is needed to send the trigger decision to the TEL62 boards, neither Host CPU or GPU have the necessary precision of 25 ns, for sending the trigger synchronously with the clock of the experiments.

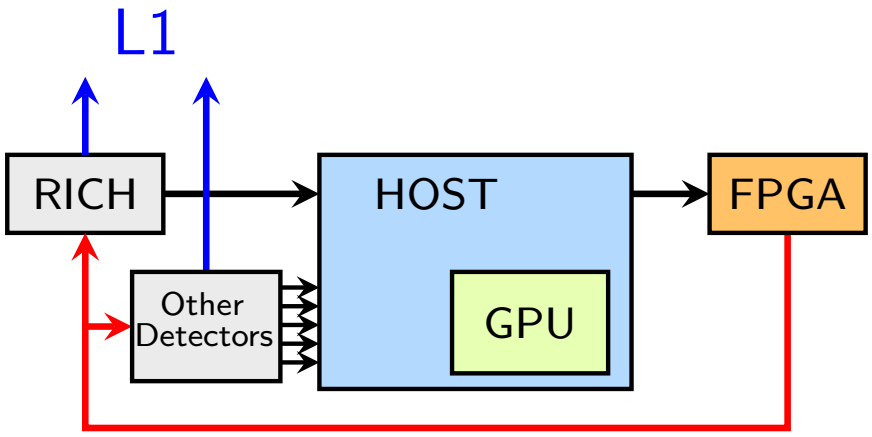


Figure 3.2: A GPU is located in a Host PC, and receive the RICH data and the primitives of the detector participating the L0, to issue a trigger decision, and send it to the TEL62. According to the trigger decision data will be discarded or sent to L1 PC farm.

The second option, described here, is illustrated in Fig. 3.3. The GPUs are inserted between the TEL62 boards and the L0TP. In this scenario the work of GPUs is simpler with respect to the one described above. The GPUs receives only the data from the RICH detector, process it and then sends a primitive compatible with the L0TP format, after this the L0TP issues a decision and sends it to the TEL62 boards.

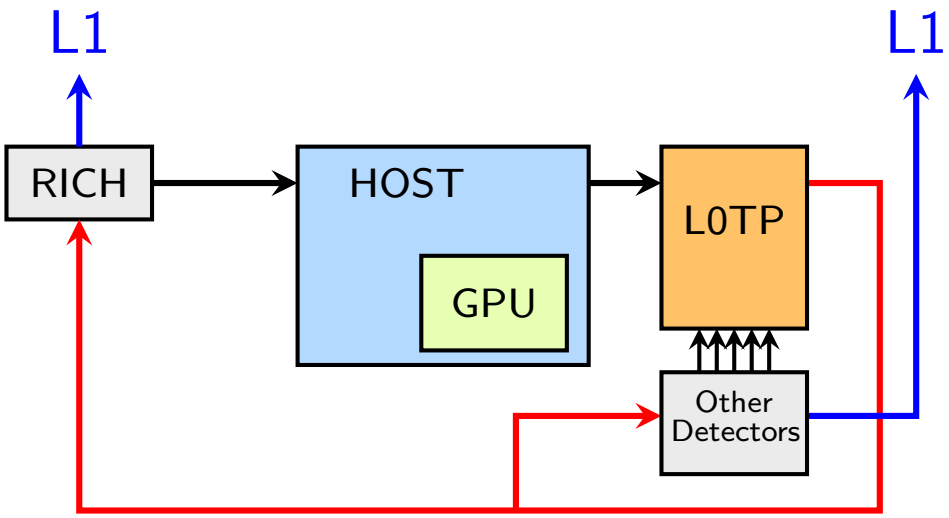


Figure 3.3: The GPU is located on a Host PC, and receives the data only from the RICH detector, and then sends primitives to the L0TP, where will be matched with the ones from the others detectors for issue a L0 trigger decision. According to the trigger decision data will be discarded or sent to L1 PC farm.

In both cases described above the time passed from when a primitives is sent by the TEL62 to a trigger decision is returned to the boards should be less than the maximum latency of 1 ms, so the computation time performed by the GPUs should a way better of 1ms (more specifically should be below 206μs, see Sec. 4.1).

It's important to remember that NA62 is considered a test-bench for the use the GPUs in lowest level trigger. In this work we aim at demonstrating that GPUs can be usefully employed in a low level trigger more than to prove that the computing power available in the present generation of video cards is enough for the NA62 needs.

In this work, is more important that the computation time of GPUs is low enough to use the graphic boards in the L0 trigger, and better result will be achieved surely in the future when better computing performance will be achieved, however if the informations provided by the GPUs increase the quality of triggered data with information more useful than the simple multiplicity is better for us.

For what concerns the higher trigger levels GPUs could be *easily* integrated in the L1 and L2 software trigger, devoted respectively to the processing of data coming from a single detector and from the entire experiment. The L1/L2 PC farms can benefit the advantages of heterogenous programming of a GPU-CPU system (Sec. 3.2.1) can be used for analyze data of certain detector,

for example in the L1 level GPUs can be used for measure the momentum of the track in the spectrometer as illustrated in [55]. In this way an increased rejection power be achieved.

More details of how GPUs are integrated in the L0 trigger level are given in section 4.1.

3.2 GPU architecture

Originally designed for the video-game market and the handling of on-screen graphics, GPUs are massively parallel multiprocessors equipped with large fast-access on-board memory . Unlike CPUs, the majority silicon area is devoted to computing units rather than to control structures (Fig. 3.4). The computing power of GPUs arises from the large number of processing cores installed on the device, rather than from clock speed (as for CPUs). The other main difference is in the purpose of the cores, while GPU cores are exclusively dedicated to computing, the CPU cores do also other functions like the handle of the peripheral device or the memory.

The main vendors of GPUs for video-gaming and scientific computing are NVIDIA and AMD . The two main toolkits available for programming GPUs are OpenCL(Open Computing Language)[35] and CUDA(Compute Unified Device Architecture)[27].

OpenCL is an open standard for parallel programming compatible with all the graphics board cited above.

CUDA is another platform for parallel programming and computing , developed for NVIDIA GPUs (like GeForce, Quadro and Tesla).

Both platform expose GPUs for computing just like any usual processor, through accelerated libraries and extension to the most popular programming languages. A set of C/C++libraries enables heterogeneous programming and provides straightforward APIs (Application Programming Interface) for device and memory management. GPUs can be embedded on the PC motherboard, or reside in dedicated graphics cards, connected to the CPU via PCI Express links.

We decided to use the CUDA toolkit, to exploit at best the characteristics of NVIDIA boards, which at this time is the leader in GPU computing. In any case porting the code to OpenCL, to adapt it to other architectures it's relatively a simple task.

From now on, I will call *host* the CPU and its memory and *device* the GPU. Serial functions, decorated with the `__host__` prefix are coded in standard C and execute on the CPU; the host can call a `__device__` function called

kernel at any moment, that will run on the GPU.

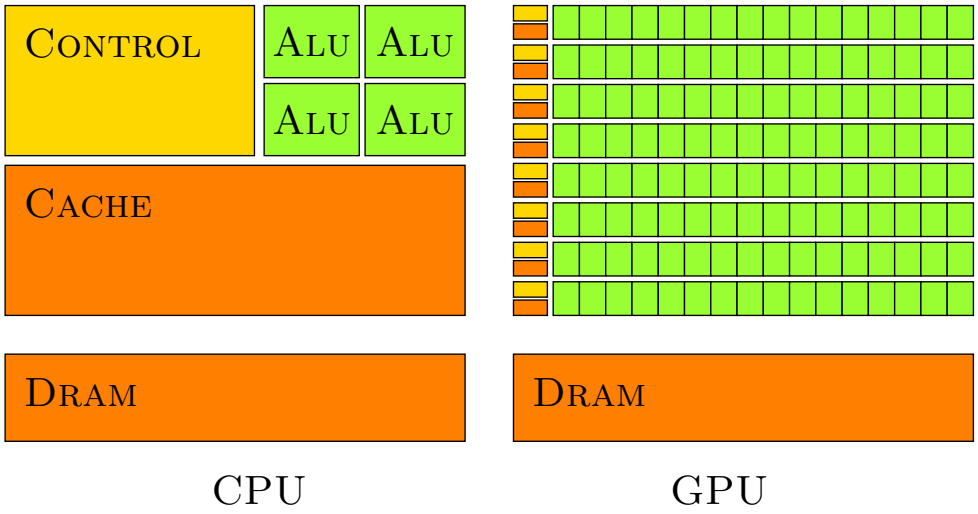


Figure 3.4: The GPU devotes more chip-area to data processing than the CPU[52].

A few definitions should be given before discussing the GPU architecture. From a programmer prospective this are:

Thread: a thread is the smallest sequence of programmed instructions that can be managed independently,

Block: the basic element of a GPU program. All the threads of a block execute concurrently.

Grid: a set of blocks. All blocks in a grid have the same number of threads.

and for what concerns the hardware implementation there are:

GPU: a entire grid is handled by a single GPU chip.

Stream Multiprocessor: the GPU chip is organized as a collection of Stream Multiprocessors(SM); each stream is responsible for handling one or more blocks in a grid. A block is never divided across multiple SMs.

Warp: the minimum *work group size*, or the maximum number of threads that can execute the same instructions simultaneously, in SIMD mode (*Single Instruction - Multiple Data*), within a single multiprocessor. All the threads in a warp can't be divided between more blocks, so to achieve the performance the number of threads into a block should be a multiple of the warp dimension. Currently, all NVIDIA GPUs feature warps of 32 threads.

The SIMD mode cited above is the architecture on which are based the GPUs, It describes computing machine with multiple processing elements that perform the same operation on multiple data. The main advantage with respect to the SISD(*Single Instruction - Single Data*) architecture of a standard CPU, to do the same operations on a given set of data points , in the SIMD mode the data are read all together from the memory, processed in parallel and written back in the memory. While in SISD mode the cycle read, process and write needs to be repeated a number of times equal to the data to process. So the SIMD architecture is good for the handle the pixels of a monitor.

When a kernel is launched all the blocks of the grid block can be scheduled in any order or any of the available SMs, allowing for program scalability: devices with more SMs automatically outperform older GPUs, as sketched in Fig. 3.5. The number of blocks running simultaneously on on SM depends also on the number of threads per block, the smaller is the number of threads per block, the larger is the number of block running concurrently.

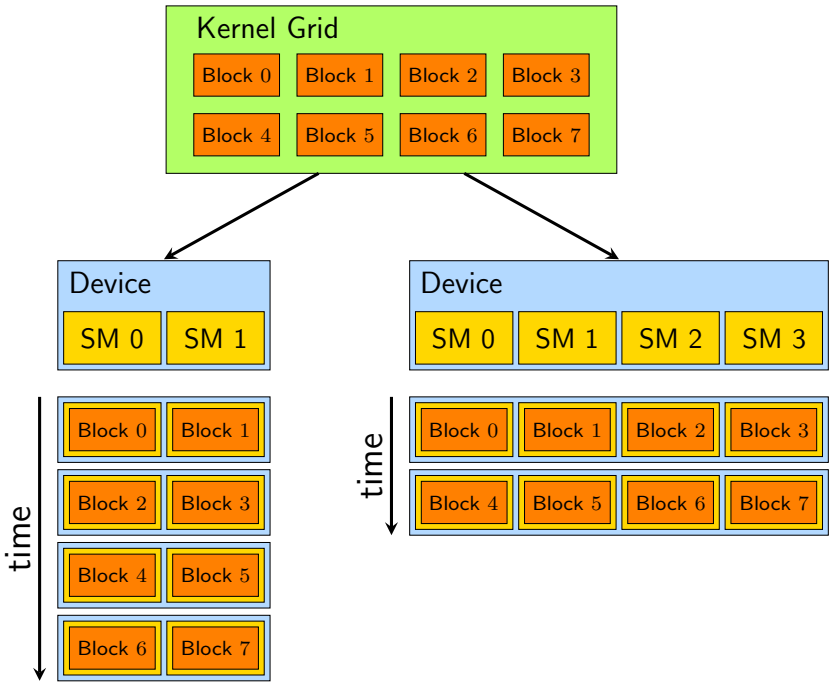


Figure 3.5: A GPU is built around an array of Streaming Multiprocessors (SMs). A multithreaded program is partitioned into blocks of threads that execute independently from each other, so that a GPU with more multiprocessors will automatically execute the program in less time than a GPU with fewer multiprocessors[51].

3.2.1 Heterogeneous Programming

As illustrated in Fig 3.6, the CUDA programming model assumes that the CUDA threads execute on a physically separate device that operates as a coprocessor to the host running the C program. This is the case, when the kernels execute on a GPU and the rest of the C program executes on a CPU. After the kernel launch the control is returned to the host, which continues the execution of the serial code, in this way selecting carefully the division of the code between GPU and CPU the best performance could be achieved. The host also provides to the device memory allocation (`cudaMalloc`, to be done before the kernel launch) and deallocation(`cudaFree`), and data transfer between GPU and CPU memories and vice versa(`cudaMemcpy`).

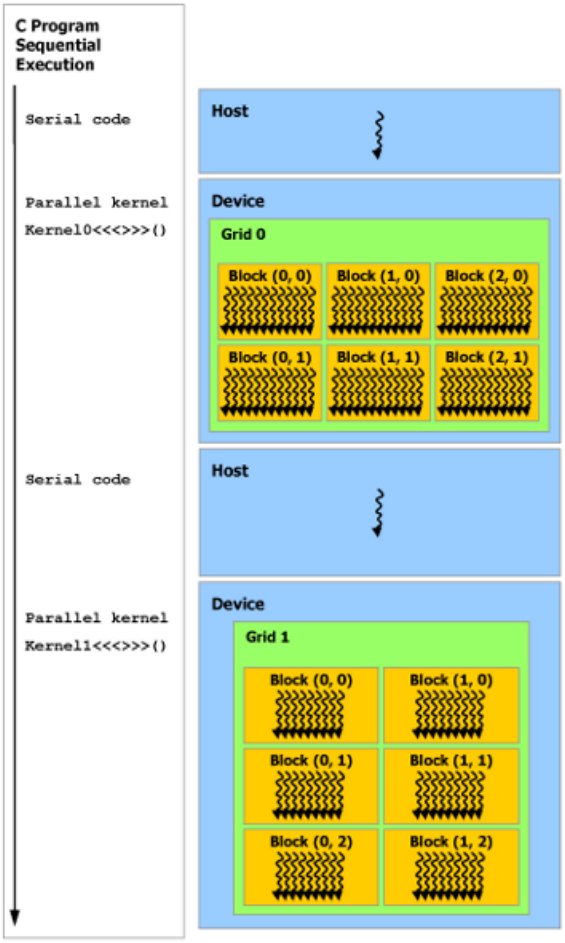


Figure 3.6: Serial code executes on the host(PC), while parallel code executes on device(GPU).

After all the definition given, a kernel is launched by the host with a call

```
1 mykernel<<<blockPerGrid,threadsPerBlock>>>(input* somestuff,
    output* someresults);
```

where:

- triple angle brackets denote a call to *device* function by *host* ;
- the input parameters point to device memory, that must be allocated before the kernel call;
- `mykernel` is executed on grid of `blockPerGrid` indipendents blocks;
- each block is split into `threadsPerBlock` threads that can synchronize and communicate to each other through access to shared memory.

3.2.2 CUDA memory hierarchy

The CUDA architecture features a hierarchy of memory spaces accessible for different purposes and with different performances.

global memory: all threads have read/write access to this large common memory space(some GB). This memory is persistent across subsequent kernel launches within the same application. This memory is also visible to all threads in a grid; its main disadvantage is the great access latency time ($400 \div 600$ clock cycles).

shared memory: this memory resides on each SM, is visible to all the threads in a block. The lifetime of data on shared memory coincide with the execution time of the block. The access time to the memory is 100 times faster ($20 \div 40$ clock cycles) than the global memory and is visible to all threads in a block, but this memory is not very large, only 48 KB for SM for the GTX TITAN (Appendix)board used during this analysis.

registers: small private local memory space owned by each single thread. The lifetime of data on registers coincide with the execution time of the kernel in that thread. Access to registers are the fastest($\sim$ 10 clock cycles) and their size is around one hundred of Byte.

The different characteristics in access time and size of the various memories determine how they are used.

The global memory is the best way to make data available to all the threads. But due to its long access time, if more than one read operation from global memory is necessary during kernel execution, it is better to copy data into shared memory or registers.

Due to the characteristics, shared memory it's mostly used for communication between the threads in a block or to store data which is requested repeatedly by different threads of the same block during the kernel execution.

The registers are used only for computation by a single thread , if the data stored in registers isn't copied to shared or global memory it is lost at the end of execution.

The choice of the memory in which data should be allocated is a crucial point in order to achieve the best performances in kernel computing time.

Thanks to the performances achieved in the last few years GPUs have demonstrate to be better than CPUs for various scientific problem as Lattice QCD Calculations[58], weather forecast[32] and solving linear system[53].

In this work we want study the characteristic of these processors in a real-time context.

Chapter 4

RICH Histograms algorithm

Contents

As a first implementation of a low-level L0 trigger on GPUs in NA62, we decided to focus on the fitting of rings on the RICH detector generated from charged particles crossing its volume.

This information can be employed at the trigger level to increase the purity and the rejection power for many triggers of interest. In the standard L0 trigger the RICH information is only used to generate a PMT hit multiplicity, this is information is barely connected to the number of rings in the detector and is not very useful.

Using a ring fitting one can have a better discrimination of between multi-track and single-track events, extracted parameters might be used in later software trigger levels to perform particle identification with spectrometer data. The input rate to the RICH trigger is expected to be $\sim$ 11 MHz with an average hits multiplicity of $\sim$ 20 hits for events, the amount of data to be

processed is enormous, and is a good bench test for a first level trigger based on GPUs.

So in order to be used in a lowest-level trigger, a ring fitting algorithm needs to be:

seedless : it will be fed with raw RICH data with no previous information on the ring position from other detectors;

fast : it will run concurrently with the hardware L0 trigger, with a maximum latency of 1 ms (decision making time) and an input event rate about of 10 MHz;

None of the usual offline multi-ring algorithms have the above characteristics. The fastest algorithms use the information on particle trajectories computed by other detectors as initial guesses for the centers of the rings, while trackless algorithm like fiTQun[2], APfit[2] or Metropolis-Hastings[3], based on maximum likelihood methods or similar, are usually slow with respect to the requirements imposed by the high intensity of the NA62 experiment.

The algorithm which I developed to overcome the above limitations and to have resolutions comparable with those obtained in the offline reconstruction.

4.1 Data input to GPUs

In order to use GPUs in the RICH L0 trigger two main aspects need to be defined:

Input: how the data from readout boards are sent to GPUs

Data format: how the data sent to GPUs are arranged

4.1.1 Input

The data from readout boards need to be transferred in the GPU memory. The copy process need to have a deterministic low latency: the contribution to the total latency has to be low enough in order to respect the requirement of latency < 1ms.

The data on RICH PMT hits is produced within the TEL62 boards (2.3.6) and they are made available to the GPU trigger system trough standard 1Gb/s ethernet links.

In the standard way, data are sent to a Network Interface Control (NIC) from the readout boards, then the NIC would copy the data via PCIExpress (PCIe) into the CPU memory and finally data would be copied in the GPU

memory to be processed.

This solution has two major problems:

- the multiple copies to write data in GPU memory: NIC → CPU → RAM → GPU;
- the non-deterministic latency time, due to CPU running concurrently many different processes.

One approach for addressing the first issue is to speed up the transfer by reducing the multiple copies using a non-standard driver software on the host, such as `PF_RING`[45]. The other approach, which addressed both issues , is to avoid the copy to host completely, and this is the one adopted for this thesis.

NaNet and GPUs

In order to overcome the above limitations an approach was considered in which data is transferred directly to the GPU without action from the host. This is possible because NVIDIA GPUs implement P2P(Peer to Peer)/RDMA (Remote Direct Memory Access) protocol, this means GPUs connected via the same PCIe bus can access to each others' memories without involving the CPU (Fig. 4.1).

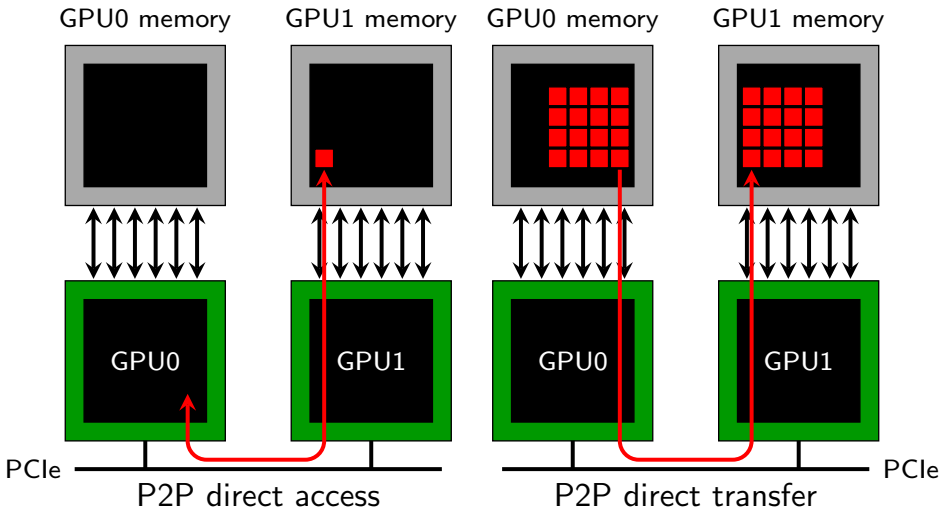


Figure 4.1: NVIDIA GPUDirect Peer-to-Peer (P2P) Communication Between GPUs on the Same PCIe Bus.

One implementation of this is the custom NaNet NIC, developed by INFN within the project APEnet+. NaNet is a FPGA-based Network Interface Card with GPUDirect P2P/RDMA capabilities [1]. This essentially means that NaNet can copy data directly into GPU memory, because it's seen by the video card as another GPU device and can use the data sharing mechanism between GPUs.

Using this solution data are sent to NaNet , then NaNet copies data directly into GPU memory, without involving the CPU in the process. In this way data are transferred with a low and deterministic latency time as intended[6]. Figure 4.2 shows how, for a buffer size smaller than 8KB, the transfer time is below 100 μs. The 8KB was chosen as the maximum buffer size of the data transmitted by NaNet .

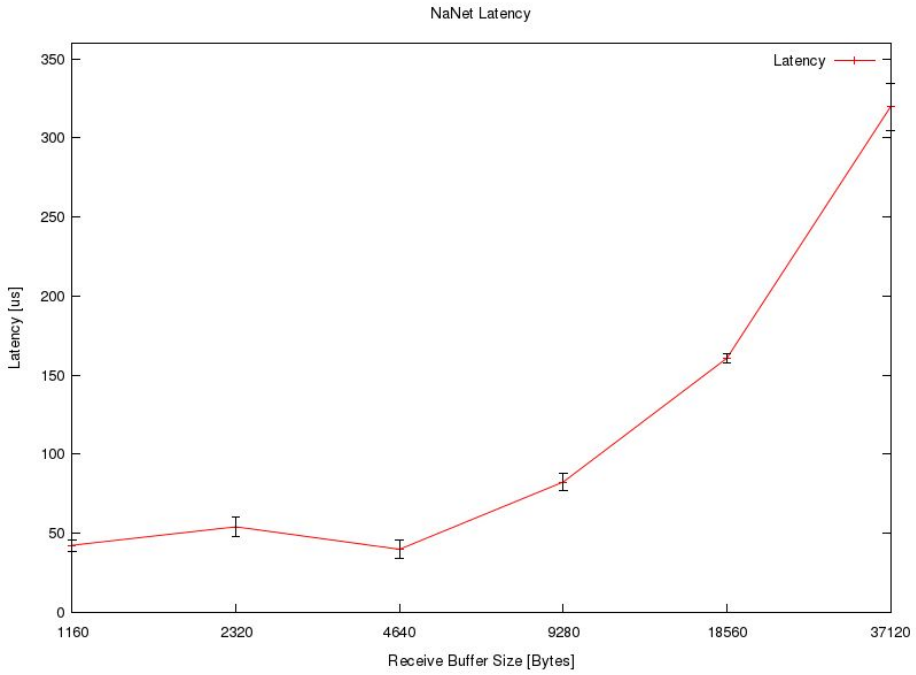
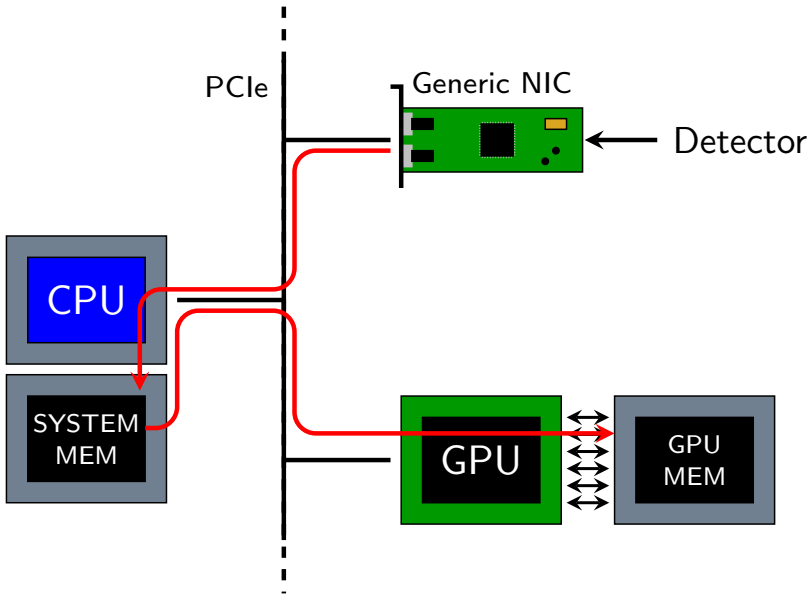
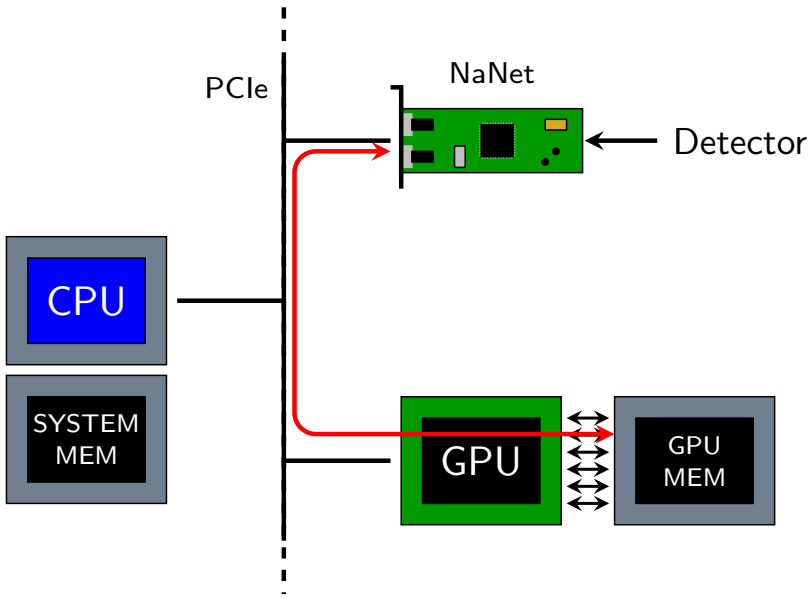


Figure 4.2: The latency time to transfer data from NaNet to GPU memory for different buffer sizes.

Fig.4.3 shows the difference in data copying between NaNet and a standard NIC.



(a)



(b)

Figure 4.3: Difference in data transfer between a generic NIC 4.3(a) and NaNet 4.3(b).

RICH data are read by on five different TEL62 boards (512 channel per board): the first four boards receive signals from single PMT; the fifth board receives data from the SuperCells (OR of 8 PMTs) and is the one used for the standard L0 trigger based on FPGA. A ring fitting algorithm based on GPUs needs to use the data from the first four boards to have all the information on the individual channel hits, so one of NaNet preprocessing tasks is to merge the data received to make it usable a by GPU kernel.

4.1.2 Data format

Each TEL62 board sends to NaNet the individual PMT hit data in a Multi GPU Packet (MGP) format shown in Fig 4.4. This format with PMT IDs coded with 9 bits has been chosen to optimize the bandwidth used by TEL62 boards.

SOURCE ID	COUNTER	FORMAT	TOTAL NUMBER OF HITS
SOURCE SUB-ID	NUM OF EVENTS		TOT MGP LENGHT
Event Data			
Event Data			
Event Data			
31...24	23...16	15...8	7...0

(a) Header of the MGP.

EVENT TIMESTAMP			
Reserved	EVENT FINE TIME	EVENT NUMBER OF HITS	
PADDING	HIT 2 PM ID (9 bits)	HIT 1 PM ID (9 bits)	HIT 0 PM ID (9 bits)
PADDING	HIT 5 PM ID (9 bits)	HIT 4 PM ID (9 bits)	HIT 3 PM ID (9 bits)
PADDING	HIT 8 PM ID (9 bits)	HIT 7 PM ID (9 bits)	HIT 6 PM ID (9 bits)
PADDING	...	...	...
31...24	23...16	15...8	7...0

(b) Event data format of MGP.

Figure 4.4: The MGP format.

The various field of the MGP format are described below

Source-ID =0x1C, is the RICH detector identifier

Source sub-ID 0x0, 0x1, 0x2, 0x3, is the identifier of the RICH TEL62 board sending the data

Total number of hits sum of all hits in the MGP(control purpose)

Counter progressive number of the MGP (4-bit, wrapping every 16 MGP)

Number of events number of events in the MGP

Event Timestamp : timestamp of the event with 25 ns LSB

Event Fine time : fine time of the event with 100 ps LSB

Event Number of hits : total number of hits in the event

Hit ID : PMT number (9 bit), the full identification number of the RICH PMT is obtained adding in front of the Hit ID the 7 LSB of the source SUB-ID field from the MGP header.

The data from the four boards are merged by NaNet according to the timestamp. NaNet takes the first events sent by each TEL62 board and searches for the one with the smaller Timestamp + Finetime, then it considers programmable time window around such time (5 Finetime units, ~ 500 ps) is open. All the events with a Timestamp + Finetime in the time window are merged in the same event, stored in a buffer called CLOP(Circular Lists Of Persistent receiving buffers) and sent to GPU for processing. The data are sent to GPU either after a certain programable timeout period (206μs at this time, the timeout start after the first MGP is arrived from the TEL62 boards) or when a buffer size of 8KB is reached. The NaNet timeout is also the maximum time available to a kernel for processing the events. If during the computation this limits it's exceed repeatedly, data would be overwrite while are read by the kernel, causing in most cases a crash. The only way to prevent the crash is keep the computing time of the kernel below 206μs.

The data format for each event is shown in Fig. 4.5

STR 3MGP	STR 2MGP	STR 1MGP	STR 0MGP	STR 3HIT	STR 2HIT	STR 1HIT	STR 0HIT	RESERVED	WINDOW	TOTAL HIT		TIMESTAMP			
STREAM 1; HIT 1		STREAM 1; HIT 0		STREAM 0; HIT 5		STREAM 0; HIT 4		STREAM 0; HIT 3		STREAM 0; HIT 2		STREAM 0; HIT 1		STREAM 0; HIT 0	
STREAM 2; HIT 0		STREAM 1; HIT 8		STREAM 1; HIT 7		STREAM 1; HIT 6		STREAM 1; HIT 5		STREAM 1; HIT 4		STREAM 1; HIT 3		STREAM 1; HIT 2	
STREAM 2; HIT 8		STREAM 2; HIT 7		STREAM 2; HIT 6		STREAM 2; HIT 5		STREAM 2; HIT 4		STREAM 2; HIT 3		STREAM 2; HIT 2		STREAM 2; HIT 1	
STREAM 3; HIT 4		STREAM 3; HIT 3		STREAM 3; HIT 2		STREAM 3; HIT 1		STREAM 3; HIT 0		STREAM 2; HIT 11		STREAM 2; HIT 10		STREAM 2; HIT 9	
PADDING										STREAM 3; HIT 7		STREAM 3; HIT 6		STREAM 3; HIT 5	
127...120	119...112	111...104	103...96	95...88	87...80	79...72	71...64	63...56	55...48	47...40	39...32	31...24	23...16	15...8	7...0

Figure 4.5: The M^{2}EGP data format

The data format called Merged Multi Event GPU Packet(M^{2}EGP) has an 128 bit long header containing

- the TIMESTAMP (32bit) of the merged event corresponds to the 24 LSB bits of MGP Timestamp + Finetime of the event with smaller value. Only events with a Timestamp + Finetime value within a programmable time window are used for the event
- the WINDOW (8bit) field contains the size of time window used for merging, with 100 ps LSB

- the TOTAL HIT (16bit) field contains the total number of hits of the merged event
- the fields STR X HIT(8 bit) have the information on the hits received from board X
- the fields STR X MGP(8 bit) contains the information on the number of MGP received from board X
- the fields STREAM (TEL62 board) X; HIT Y (16 bit) contain the ChannelID of individual hits

4.2 Implementation on GPUs

The algorithm I developed uses high number of initial guesses for the ring center, then searches the most probable radius for each center. An algorithm with this logic is intrinsically parallelizable, then adequate to be implemented on GPUs, while its implementation on CPUs would be impratical due to the large number of operations required.

4.2.1 Algorithm description

I briefly describe how the Histograms algorithm works. I consider a square grid of possible ring centers, and for each square of the grid I compute the distance between its center and the position of all the hits in the event, and fill a histogram with this quantity, (the histogram is represented by a vector of integer number in register). After this step, I search the bin with maximum number of entries in the associated histogram, and if such maximum is above a certain threshold, the bin and its contents are saved and considered for further analysis. The data are saved in the *shared* memory, in this way the results of each center are visible to all threads processing the events and a faster access to the data is possible with respect to the one obtained using the *global* memory (around 100 times slower than global memory), so at least one integer variable is needed for each square of the grid, but because the limited size of the *shared* there's a limitation on the grid size. Figure 4.6 shows how the basic Histograms algorithm works in two different cases.

All the tests were performed on a NVIDIA GeForce GTX Titan, the specifications of the board are given in Appendix B. The computation time for the Histogram kernel were obtained using the CUDA functions for the time measurement, while the simulated data are obtained with Geant4 based MonteCarlo of the NA62 experiment.

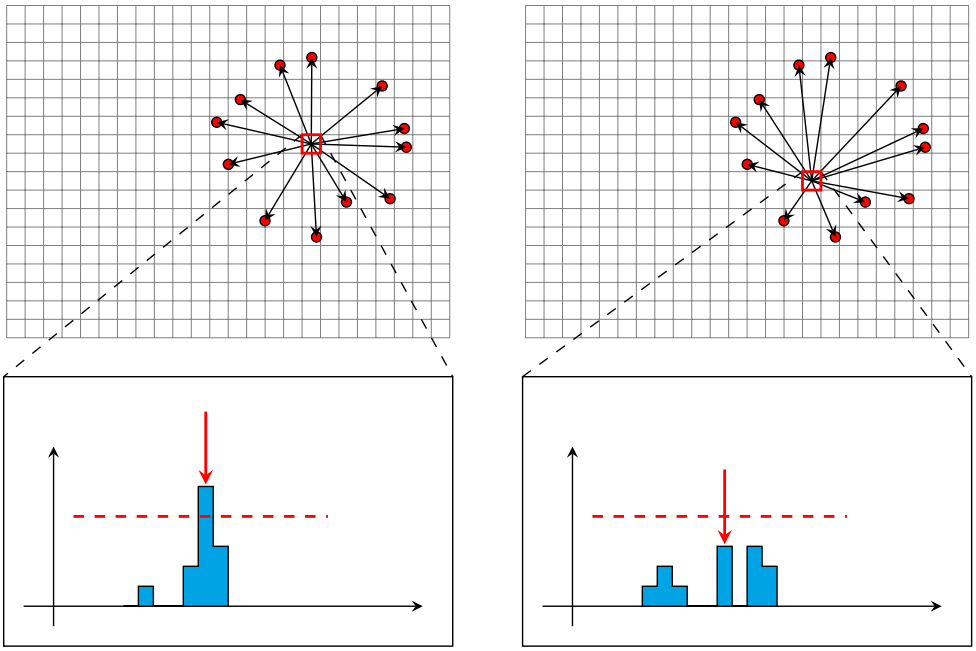


Figure 4.6: The figures show how two different squares of the grid are affected by the same set of hit. On the left side of the figure, the maximum associated to the red square is above the threshold, while this doesn't happen on the right side. So only the square on the left is considered for further analysis while the one on the right is discarded.

The more natural way to implement this algorithm on GPUs is to assign each block to an event and each square of the grid to a thread. In this way one can exploit all the parallel computing power of GPUs. If a greater number of threads is used a better precision can be achieved on the center resolution, but a larger number of threads implies a lower number of blocks executed concurrently and so a larger amount of time to process all the events.

4.2.2 First implementation

For the first test I implemented the algorithm using a GPU block for each event, with each block having 256 threads arranged in a 16×16 2-dimensional grid. Each thread of a block is associated to a grid point and works as described above. After all the threads have computed their maximum, they compare it with the maximum of the four adjacent squares. In case the maximum found is smaller than any of the one found by the nearly square, it is discarded. All the squares surviving after this step are considered as ring centers.

Figure 4.7 shows a map of the PMTs for the two flanges of the RICH with the grid superimposed and Table 4.1 summarizes the relevant parameters for

the algorithm:

Parameter	Value [mm]	Granularity [mm]
X_MIN	-477	40.625
X_MAX	173	
Y_MIN	-300	37.5
Y_MAX	300	
STEP_R	12	

Table 4.1: Parameters of the grid used in the first version of Histograms. X_MIN, X_MAX, Y_MIN and Y_MAX are the grid edges respectively along the x and y axis. STEP_R is the bin width.

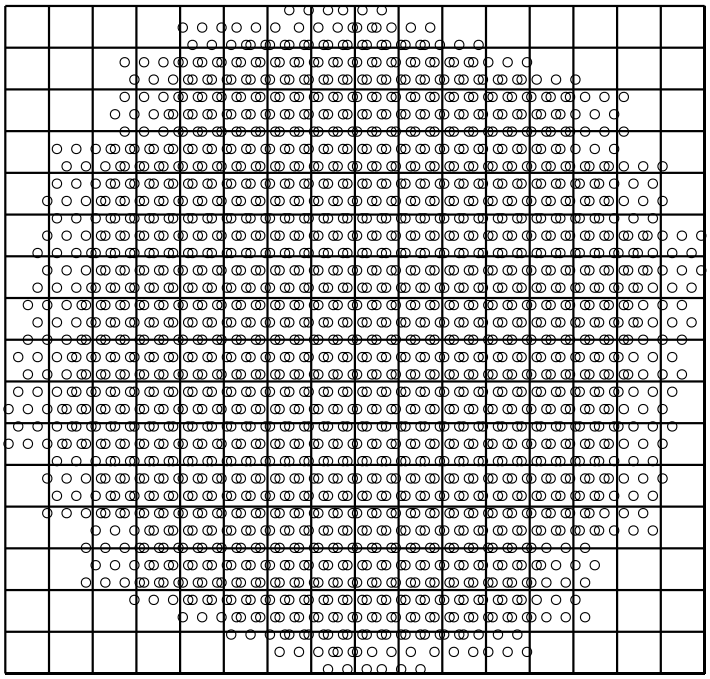


Figure 4.7: Grid used in the first version of the algorithm superimposed to the 2 PMTs maps, one for each flange of the RICH.

The computation time for this first implementation of the algorithm was too high to use the kernel in the level 0 trigger, as shown in Fig.4.8. The computing time was above 206μs (NaNet timeout) for all CLOP size considered.

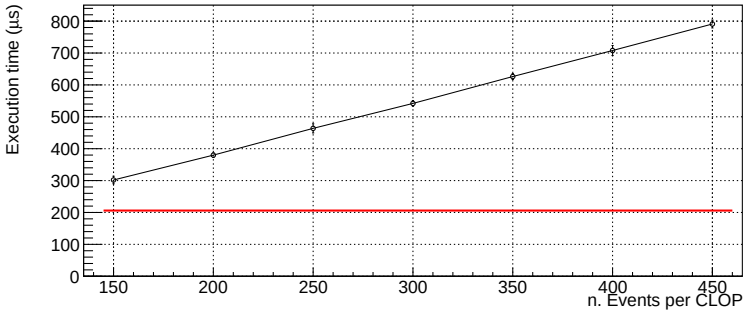


Figure 4.8: Computation time for the first algorithm version: the red line shows the limit at 206μs. This version wasn't suitable to be used in the L0 trigger. Data obtained with a MC simulation and ran on GPU GeForce GTX TITAN installed in a desktop PC.

4.2.3 Optimization

The kernel implemented in the way described above is too slow to be used in the L0 trigger of the experiment. So two major improvements were made. The first one concerns the histogram of distances: instead of using a single histogram with a bin width of 12 mm, I use two histograms with the same bin width but offset a relative shift of $1/2$ bin width. With this trick it is possible to find rings which would have been discarded using the previous version: Fig 4.9 shows how the same hit distribution doesn't satisfy the threshold condition in the first case while does in the new version.

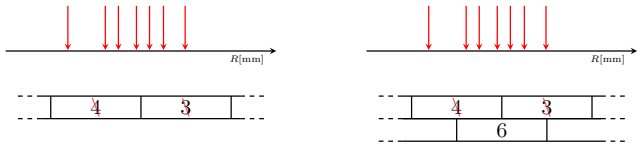


Figure 4.9: An example of how the *trick* works. The figure shows how the previous (left) and new (right) version of the kernel works. The red arrows indicate the distance values computed for different hits with respect to a given point and the rectangle below represent the bins of the histogram with their own count. In the old version of the algorithm, the two bins count 3 and 4 and aren't above the threshold and they are discarded. In the new version, there is a second histogram offset a relative shift of $1/2$ bin width, so there is one bin of the second histogram with 6 counts and the point will be considered.

The second improvement was related to the structure of the algorithm, I use a two step algorithm, the first step works in the same way described in section 4.2.2, but on a 8×8 grid, and the centers with maxima above the threshold are considered for the second step. In the second step the interesting candidate centers found in the first step are analyzed on a 4×4 sub-grid covering the region of interest (Fig. 4.10).

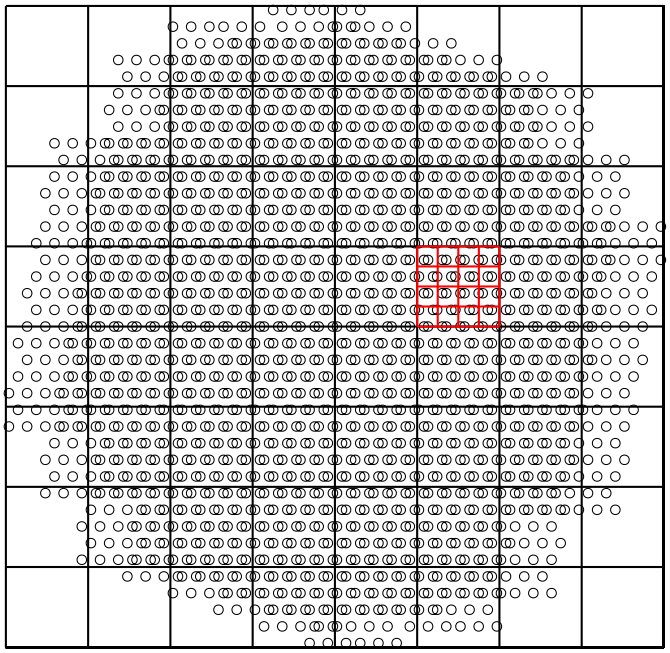


Figure 4.10: The figure shows the grid used in the first step of the algorithm in black, and the sub-grid used for the second step in red, superimposed to the 2 PMTs maps, one for each flange of the RICH.

The solution discussed above has two advantages:

- only 64 threads are used instead of 256, so the number of blocks which run concurrently is increased by factor 4, for a given number of SMs on a board;
- the space resolutions on the centers along x and y axis are improved by a factor 2.

This solution has a disadvantage in that only 64 threads are available per block, a maximum of 4 of the squares found by the first step can be analyzed

simultaneously. Fig 4.12 shows the number of square found at the first step for single and multi-ring events, the fraction of events with 5 or more ring are 2.2% and 3.5% for single-ring and multi-ring events respectively. Despite this fact a speed up in computing time is achieved with respect to the first version, but it is not yet enough to use the algorithm in L0 trigger as shown in Fig. 4.11.

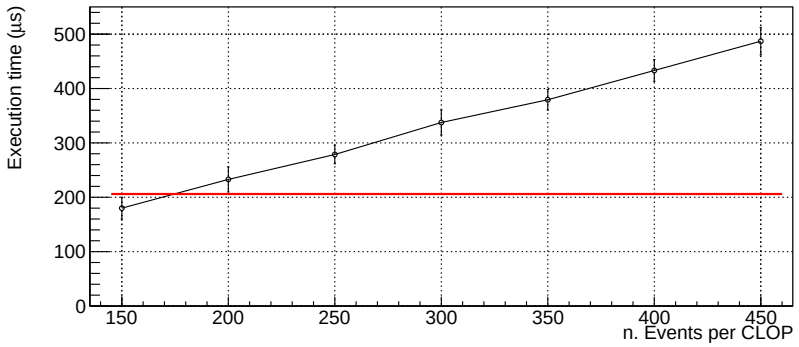


Figure 4.11: Computation time for the second version of algorithm, only clop with less than 150 events. Data obtained with a MC simulation and ran on GPU GeForce GTX TITAN installed in a desktop PC.

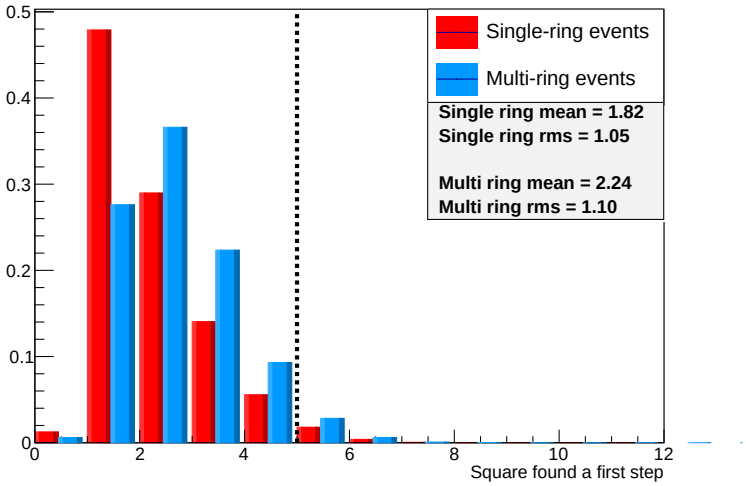


Figure 4.12: Number of the square found by the kernel at first step, for single and multi ring events, the black dashed line separates the events with 5 or more squares found during the first step from the ones with 4 or less squares. Data simulated with MC and normalized to unity.

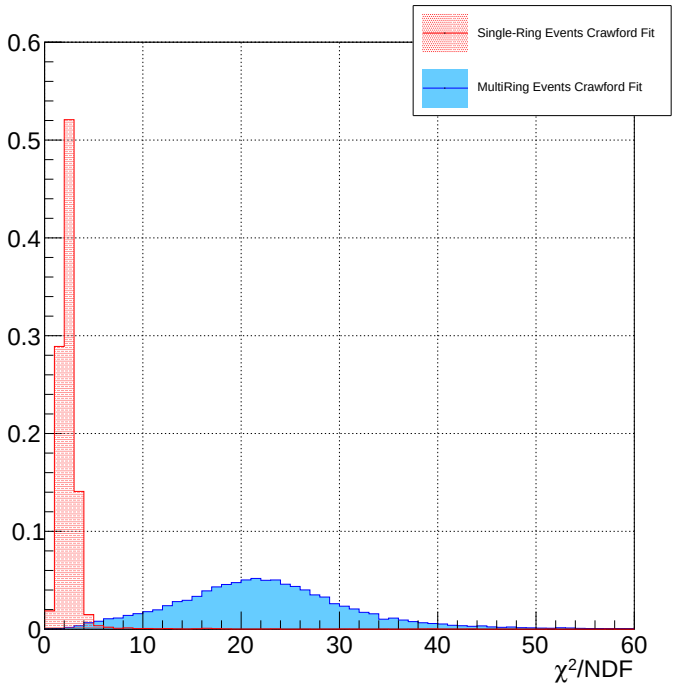
4.2.4 A different approach : Single ring vs Multi rings

Because most of K^+ decays ($\sim 95\%$) have only one charged particle in the final state (and so only one ring in the RICH), a possible way to speed up the algorithm is to use a fast single-ring fitting algorithm for one ring events and use Histograms for the multi-rings event. One does not know *a priori* if an event has one or more rings. To face this problem a fast single-ring fit algorithm is used, then a decision based on the χ^2 of the fit is made: if χ^2 doesn't exceed a certain threshold the event is considered as a one ring candidate, otherwise data are analyzed with the Histogram algorithm.

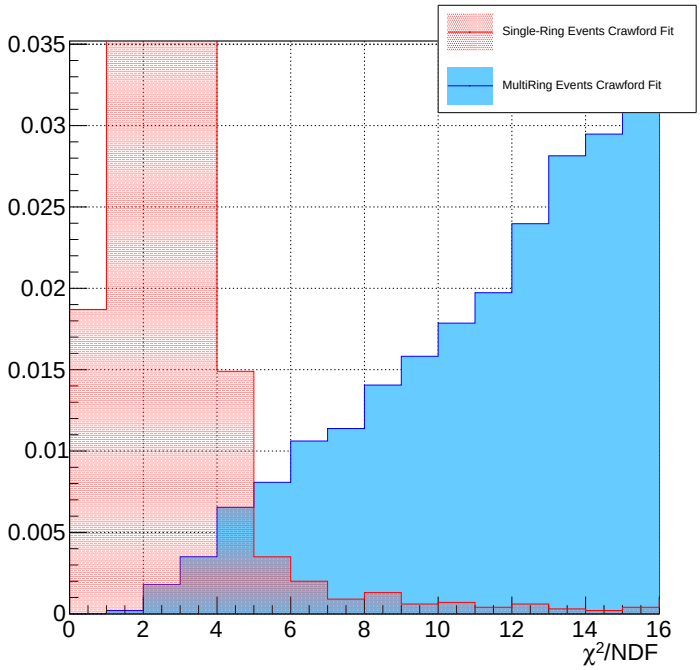
The single-ring fit algorithm used is the Crawford algorithm[28]: the hits centroid is translated to the origin $O(0,0)$, in this system a least square method can be used to fit a ring. The condition can be reduced to a linear system , analytically solvable, without any iterative procedure (for more details see Appendix A). Using the Crawford algorithm on both one and multi ring events and evaluating the χ^2 the plots shown in Fig 4.13 were obtained. The threshold value on χ^2/NDF (Number Degree of Freedom) was chosen to minimize the misidentification of multi rings events seen as single-ring event, and to maximize the correct identification single-ring events. The value of the threshold and the fraction of misidentified events are summarized in Tab 4.2.

χ^2/NDF threshold	5
Multi Ring $\Rightarrow$ One Ring	1.4%
One Ring $\Rightarrow$ Multi Ring	1.6%

Table 4.2: Chosen χ^2 threshold and the corresponding fraction of misidentified events.



(a)



(b)

Figure 4.13: (a): χ^2/NDF distributions for one and multi rings events fitted with Crawford algorithm. (b): zoom on the interesting zone of the top plot. Data obtained with a MC simulation, in red $K^+ \to \pi^+\nu\bar{\nu}$ events , in blue $K^+ \to \pi^+\pi^+\pi^-$ events with at least 2 rings in acceptance.

Because only the processing of multi-ring events is left to the Histogram algorithm, I studied with MC the center distributions for different types of these events. Figures 4.18(a) and 4.18(b) show the distributions of the center coordinate respectively on x and y axis. One can see in Fig.4.18(a), the multi-ring events are distributed asymmetrically along x-axis with respect to the center of the detector; this is due to the detector geometry, chosen to maximize acceptance as possible for $K^+ \to \pi^+ \nu\bar{\nu}$ decays.

I changed the grid size according to these multi-ring distributions, (Tab. 4.3 summarizes the new parameters), and since the new grid is smaller than the previous one, I decided to use a 2×2 sub-grid in the second step instead of a 4×4 sub-grid: this change allows 16 firing squares to be considered at the same, avoiding the problem described in section 4.2.3, which is also more relevant in this kernel version (Fig. 4.16). Figure 4.17 shows the new grid superimposed to the PMTs.

Parameter	Value [mm]	Granularity [mm]
`X_MIN`	-150	18.75
`X_MAX`	150	
`Y_MIN`	-150	18.75
`Y_MAX`	150	
`STEP_R`	12	6

Table 4.3: Parameter of the grid used in the final version of Histograms, the granularity in the third column for the radius is due to the optimization explained in section 4.2.3

The performance of new algorithm version is fast enough to be used in L0 trigger. Figure 4.14 shows the execution time for the kernel on a GPU in a desktop PC and Fig. 4.15 shows the measurements performed simulating the chain TEL62 $\to$ NaNet $\to$ GPU with data collected at the NA62 experiment: in both cases the execution time don't exceed the 206μs. The data shows in Fig. 4.15 are collected only from one flange, while those simulated in Figure 4.14 came from both sides. This difference in the data set affect the computation time of the kernel, this is due to the larger number of hits in the events, this difference can be seen in Figures 4.14 and 4.15 watching the CLOPs with 300 events, in the first case the computation time is over 150 μs while doesn't exceed this value in second case.

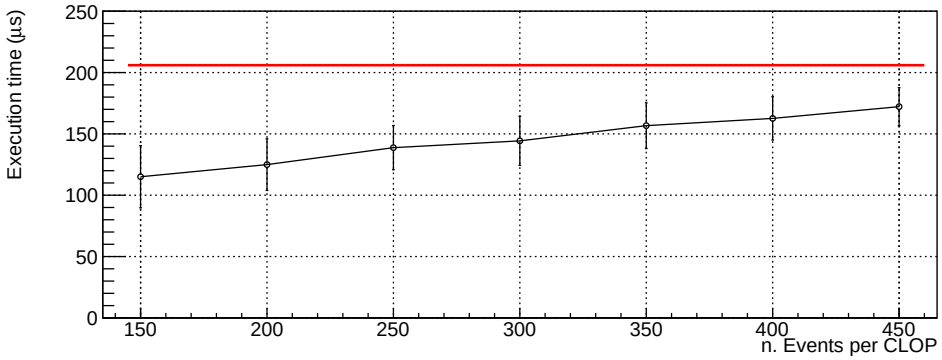


Figure 4.14: After the split of single and multi ring events, the kernel performance is good enough to be used in the L0 trigger. Data obtained with a MC simulation and ran on GPU GeForce GTX TITAN installed in a desktop PC.

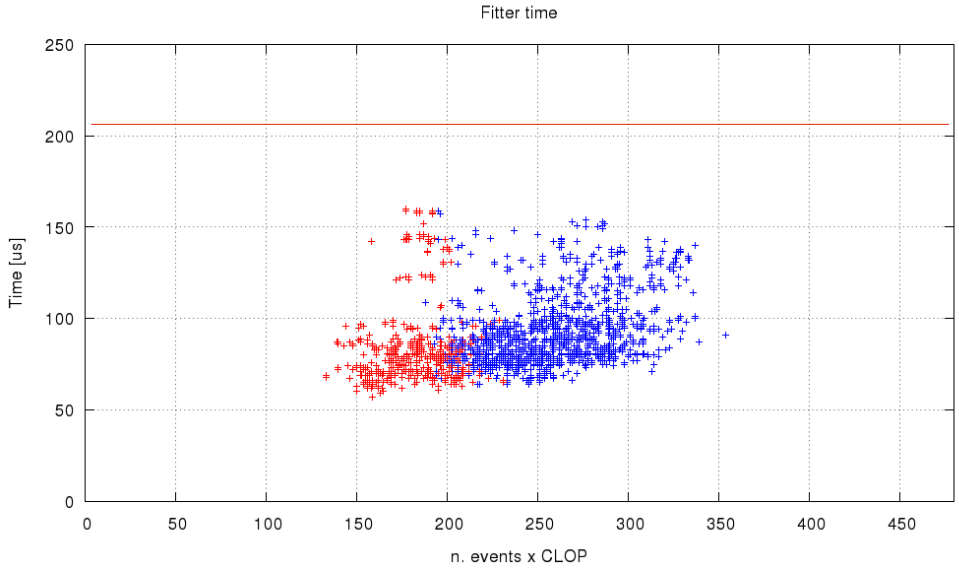


Figure 4.15: The red points are Clop with size smaller than 4KB, the blue ones for Clop with a size larger than 4KB. Data obtained in Pisa simulating the chain TEL62→ NaNet → GPU with data collected at the NA62 experiment, the data are collected only by one flange; the GPU used is a GeForce GTX TITAN.

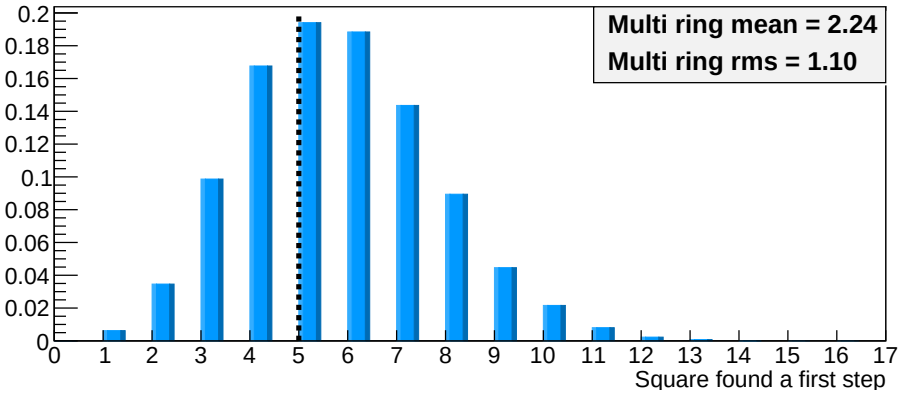


Figure 4.16: Number of the square found by the kernel at first step for multi ring events after the χ^2/NDF cut. I haven't considered the multi-ring events because they are discarded by the cut. Data simulated with MC and normalized to unity.

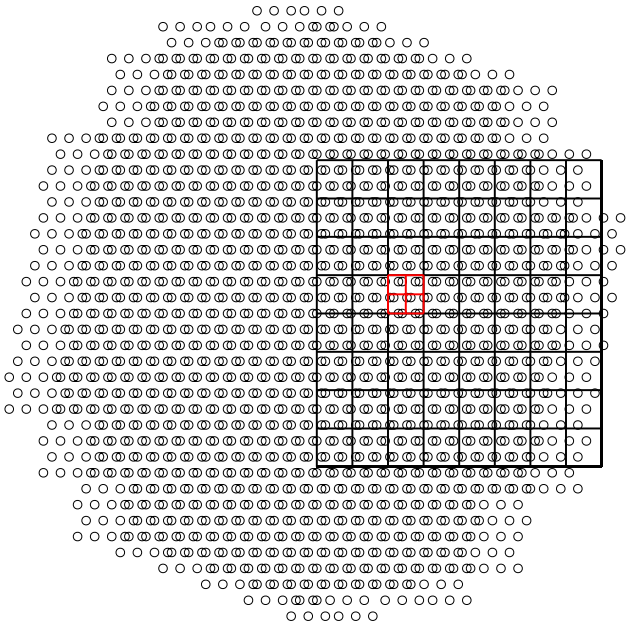


Figure 4.17: The figure shows the grid used in the first step in black, and the sub-grid used for the second step in red, superimposed to the 2 PMTs maps, one for each flange of the RICH. The grid doesn't cover the totality of PMTs. Rings with center out of the grid can't be identified, but how one can see by Fig. 4.18, only a small fraction of events have ring center out of the grid. Moreover a finer grid improves the resolutions on the ring centers.

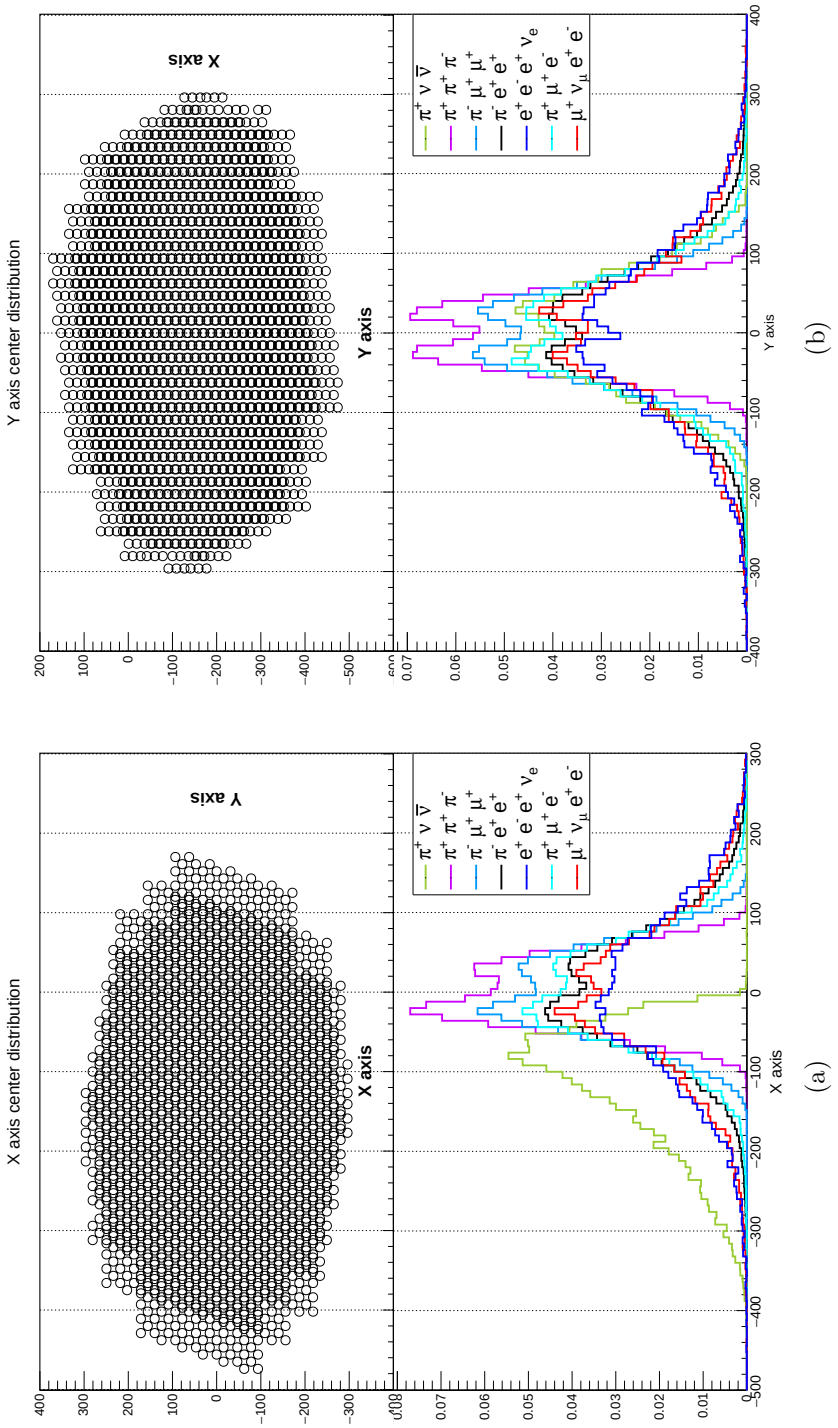


Figure 4.18: Center distributions for different kaon decays. One can see how the distribution of centers is offset to achieve a larger acceptance for $K^+ \to \pi^+ \nu\bar{\nu}$ decays.

4.3 Algorithm resolution

With a good time performance achieved, now we can see the resolution obtained by the algorithm for the center coordinate and radius. Because of the separation between single ring and multi ring events, I show the results obtained by the algorithm, separating the two event classes . The achieved results are shown in Tab. 4.4, moreover a finer grid improve the result.

Parameter	σ [mm]
r	2.56
x	3.75
y	3.67

Parameter	σ [mm]
r	5.33
x	9.03
y	7.63

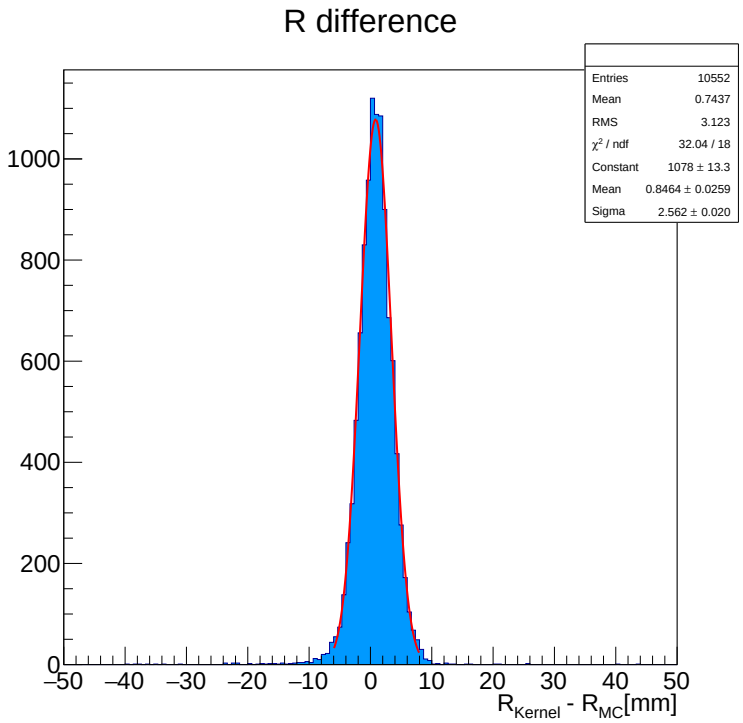
Table 4.4: Left side: resolutions achieved for the single ring event. Right side: resolution achieved for multi-ring events. Data obtained with a Gaussian fit of the difference between reconstructed and true r,x and y variables.

The Histogram results for multi rings fit are not so satisfactory; a better resolution could be achieved using Crawford algorithm on the rings found by Histogram. Tab 4.5 shows the final results.

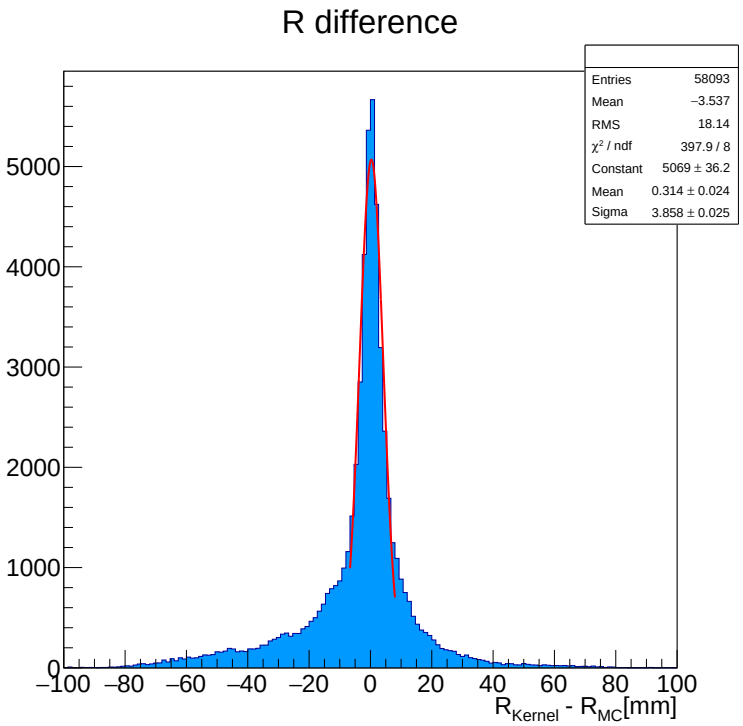
Parameter	σ [mm]
r	3.85
x	6.18
y	6.46

Table 4.5: Resolutions for multi rings event in the final version of the Histograms.

Figures 4.19,4.20 and 4.21 show the resolutions obtained for r, x and y variables for single and multi-ring events. Figure 4.22 shows three different events fitted by the algorithm

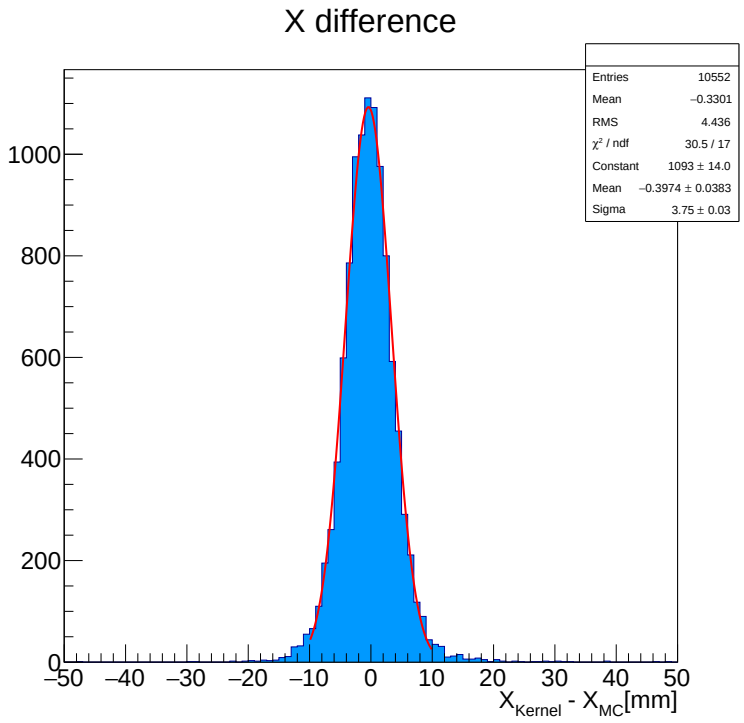


(a)

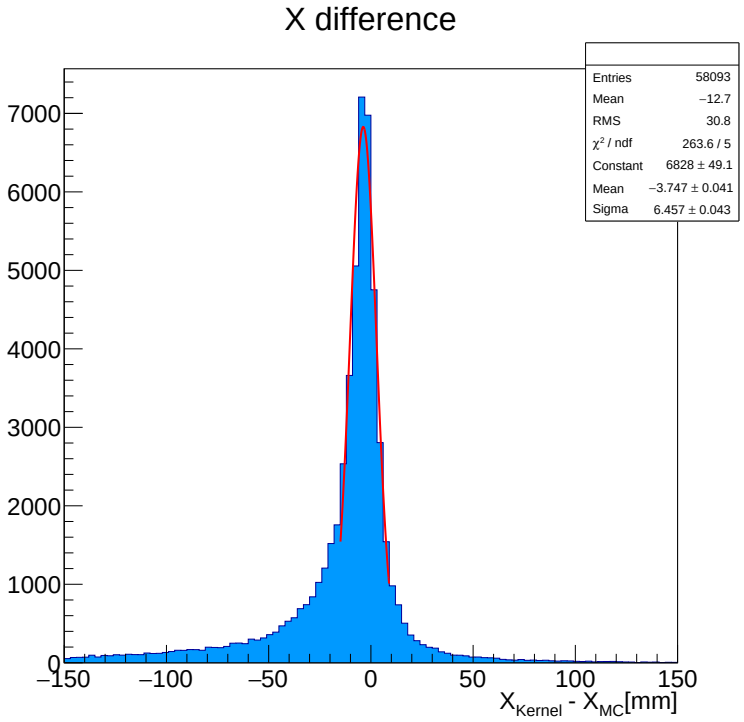


(b)

Figure 4.19: Difference between true and reconstructed r variable for single-ring (a) and multi-ring (b) events .

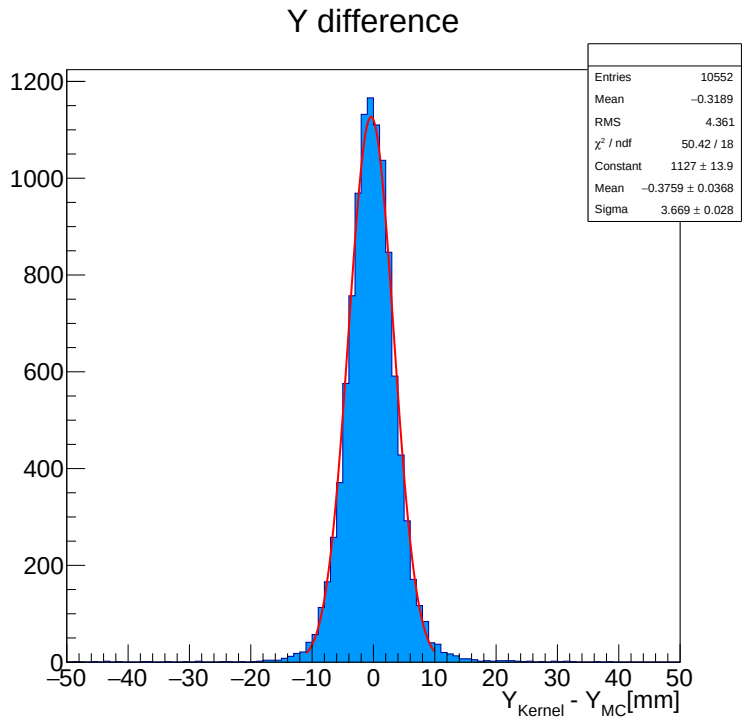


(a)

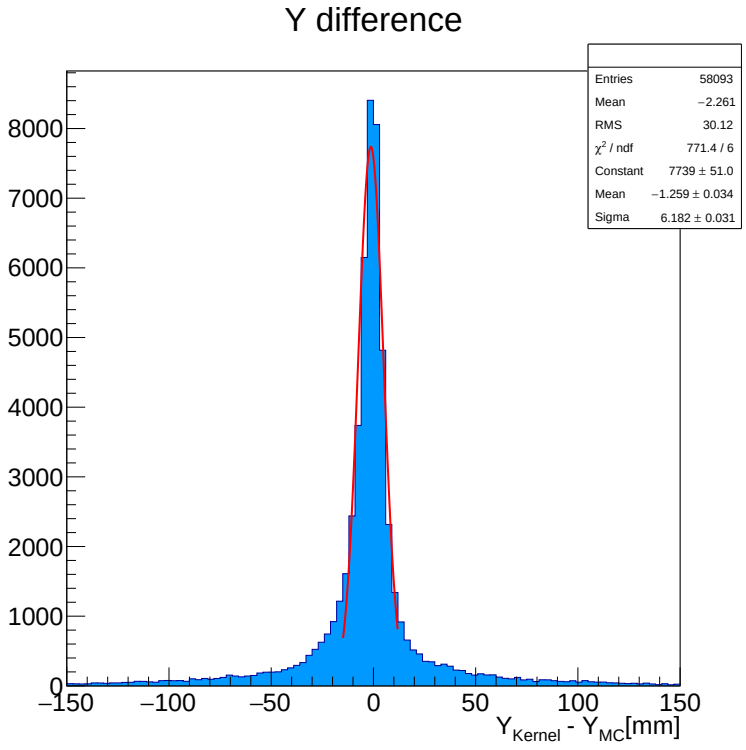


(b)

Figure 4.20: Difference between true and reconstructed x variable for single-ring (a) and multi-ring (b) events. The non-symmetric tails (b) of the distributions are due the non-centered grid used shown in Figure 4.17.

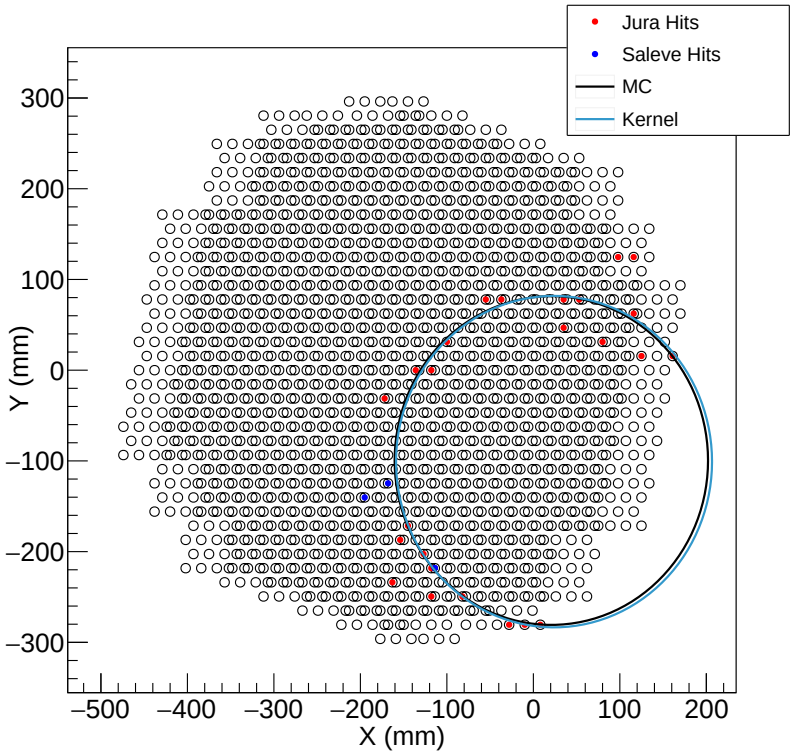


(a)

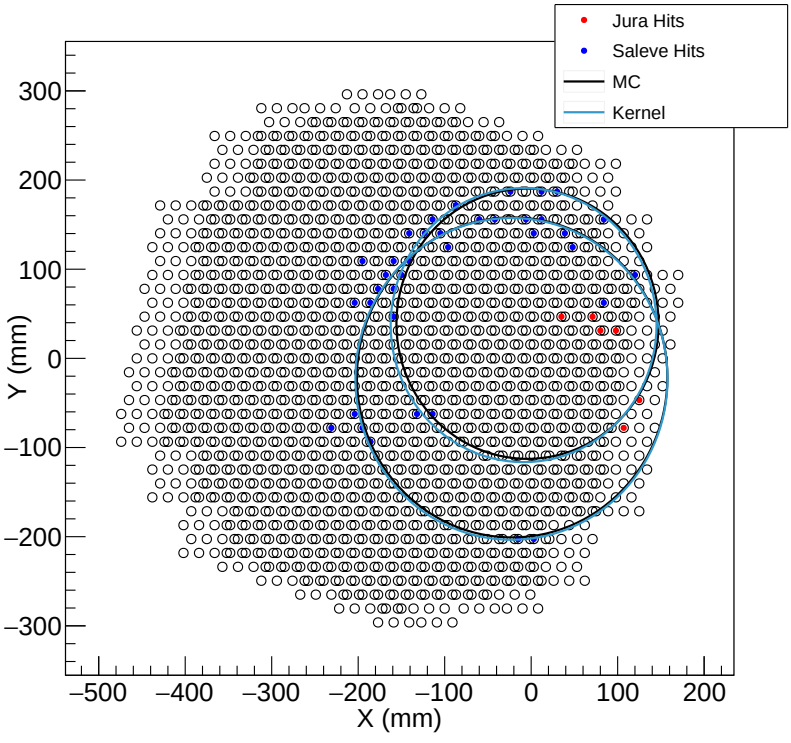


(b)

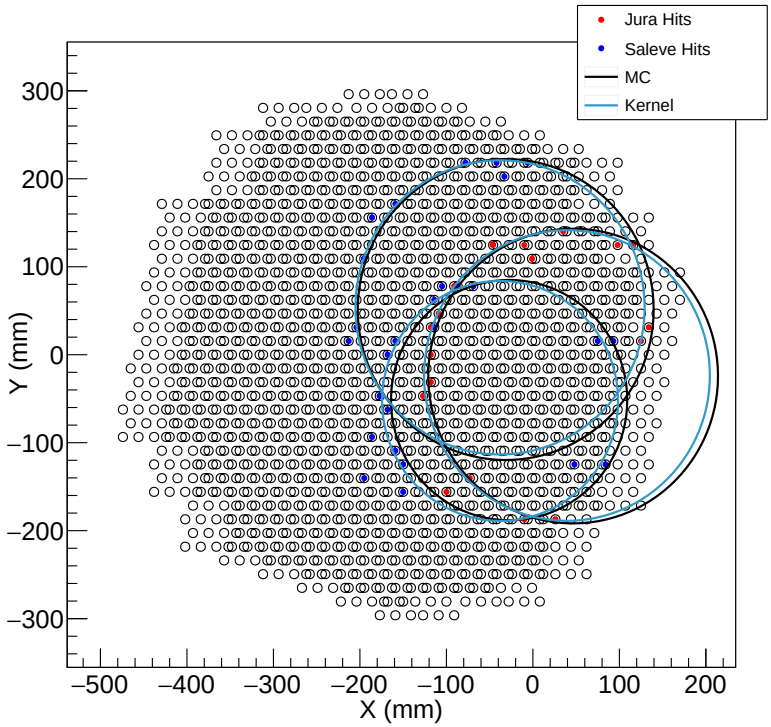
Figure 4.21: Difference between true and reconstructed y variable for single-ring (a) and multi-ring (b) events.


Jura Hits
Saleve Hits
MC
Kernel
300
200
100
0
−100
−200
−300
Y (mm)
−500
−400
−300
−200
−100
0
100
200
X (mm)

(a)


Jura Hits
Saleve Hits
MC
Kernel
300
200
100
0
−100
−200
−300
Y (mm)
−500
−400
−300
−200
−100
0
100
200
X (mm)

(b)



(c)

Figure 4.22: Three different events fitted by the algorithm, with one (a), two (b) and three (c) rings respectively. In black the truth MC ring and in azure the ring fitted by the kernel.

4.3.1 Comparison with Almagesto

The above results must be compared with the ones obtained by the offline reconstruction. The algorithm used for the offline reconstruction is a multi ring algorithm called Almagesto[44]; this algorithm was also implemented on GPU, so I'm going to compare also the computation time of the two algorithms. The algorithm is based on Ptolemy's theorem about four-side polygon inscribed in a circle. The Ptolemy theorem states that *for a cyclic quadrilateral (a four-side polygon whose vertices all lie on a single circle), the sum of the products of the two pairs of opposite sides equals the product of diagonal.* In formula (see Fig.4.23 for vertices and segment names):

$$\overline{AB} \cdot \overline{CD} + \overline{AD} \cdot \overline{BC} = \overline{AC} \cdot \overline{BD} \tag{4.1}$$

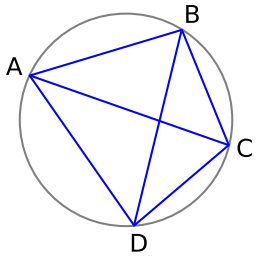


Figure 4.23: Ptolemy's theorem is a relation among the lengths of the sides and those of the diagonals in a cyclic four-side polygon.

The Almagesto algorithm has two steps: in the first one a search is performed for the hits belonging to a ring, then each set of hits is fit using the single-ring fit algorithm of Crawford. For the first pattern recognition step the Ptolemy's theorem is used: first of all the algorithm selects 8 triplets of points as follows:

- the leftmost three points
- the rightmost three points
- the three points at the bottom
- the three points at the top
- the leftmost three points, after a 45° rotation
- the rightmost three points, after a 45° rotation
- the three points at the bottom, after a 45° rotation
- the three points at the top, after a 45° rotation

The triplets are chosen at the edges of the x, y, u and v axis because in this way the chance that the points belong to the same ring is higher.

The resolution obtained from Almagesto algorithm are shown in Tab 6.1, while Figures 4.25,4.26 and 4.27 compare the resolutions between Almagesto and Histogram for r,x and y variables respectively; one can see the non Gaussian tails of Almagesto offline reconstruction are smaller than the ones of Histograms.

Parameter	σ_{Alm} [mm]	σ_{Hist} [mm]	Parameter	σ_{Alm} [mm]	σ_{Hist} [mm]
r	2.05	2.56	r	3.08	3.85
x	2.07	3.75	x	4.92	6.18
y	2.69	3.67	y	4.12	6.46

Table 4.6: Left side: resolutions achieved for single ring events. Right side: resolutions achieved for multi-ring events. The second column shows the Almagesto resolutions, the third shows the Histogram resolutions.

Because both Histogram and Almagesto use the Crawford method to fit the ring, shouldn't be any difference in the resolution obtained by the two algorithms.

I'll try to explain the reason of these differences.

The main difference between the two algorithms is in the selection of the points used for the Crawford fit. Almagesto use 8 triplets and Ptolemy theorem for search the point belonging to a ring. For the single-ring events there are very high chance that one of the triplets lie on the ring and selects efficiently all the hit to use in the fit. While Histogram use all the hits in the events, so also points due to the noise, and this affect the resolution achievable. This effect can be seen in Fig 4.24.

For what concerns the multi-rings events, the difference are in the approach to the events, Almagesto like said above searches the hits belonging to the ring and then fits the points, the worse resolutions respect to the single-ring events are due only to the major complexity of these type of events.

While the rings found by the Histogram algorithm are strongly related to the grid used, if the grid is too loose, the resolutions couldn't be very good and are slightly improved by the use of Crawford algorithm.

Summarizing, the main difference is: Almagesto searches the hits belonging to a ring, Histogram searches in a high set of possible candidate rings (equal to the number of grid squares $\times$ bin of the histogram associated to each square) the combinations of ring which is more plausible for a given set of firing PMTs, so if the set of ring isn't .

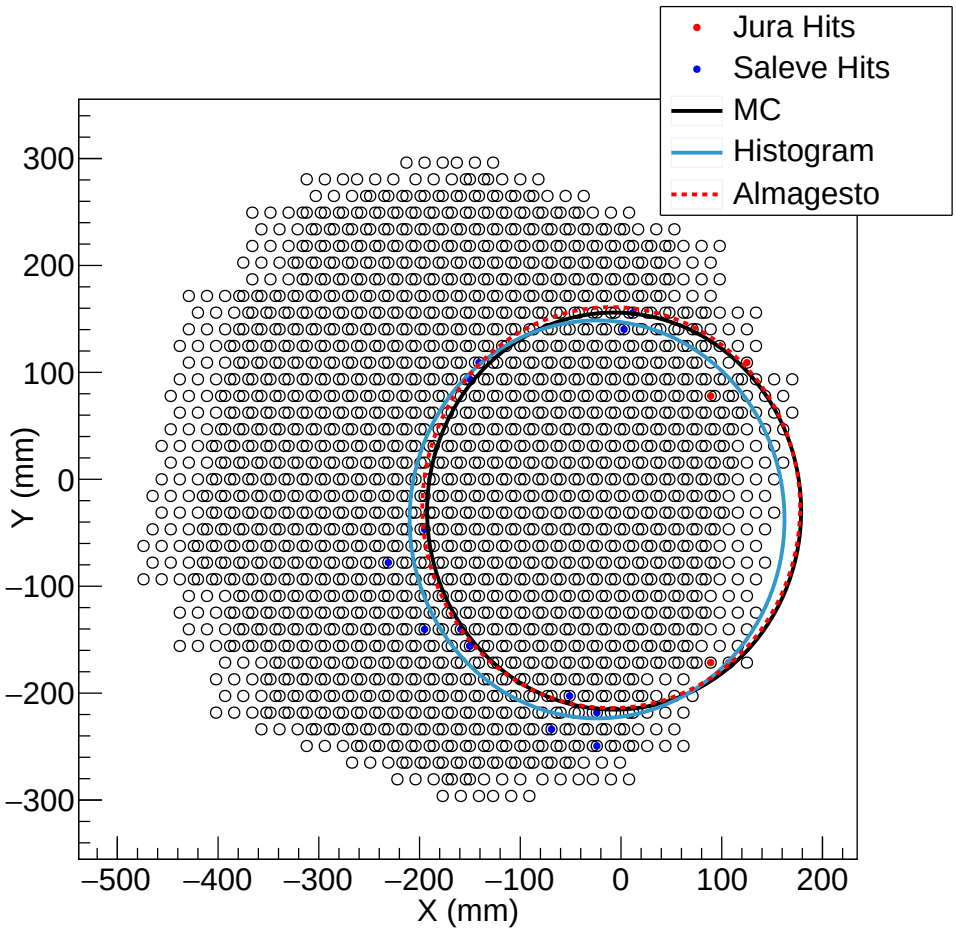


Figure 4.24: The same single-ring event fitted by Almagesto and Histograms. Almagesto selects only the good point and then the resulting fit is very close to the true MC ring. Histogram use all the firing hits, also the one non belonging to a ring and so the fit is worse with respect to Almagesto.

The GPU version of Almagesto is described in detail in [33], because the fitting method is the same of the offline reconstruction, also the resolution will be the same. For the computing time per event of GPUs version of Almagesto is $\Delta t_{evt} = 0.97 \pm 0.02 \mu s$ to be compared with the one obtained with Histograms. Using the computation time measured for the Clops with 450 events which is 172 ± 26 μs, we obtain for a single event

$$\Delta t_{evt} = 0.38 \pm 0.06 \ \mu s \tag{4.2}$$

So an improvement by more than a factor 2 in computing time has been obtained with respect to GPU version of Almagesto; by the way the resolution achieved by Histograms are worse, but they are still good enough to use

Histograms in the L0 trigger of the RICH as intended.

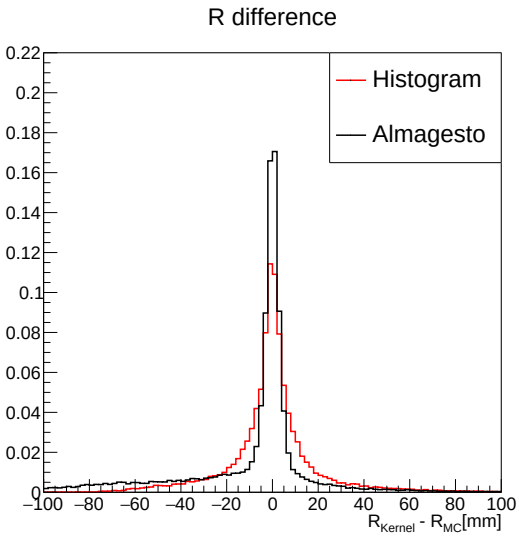


Figure 4.25: Difference between true and reconstructed r variable for Almagesto offline reconstruction (black) and Histogram kernel(red). Both plots are normalized to unity.

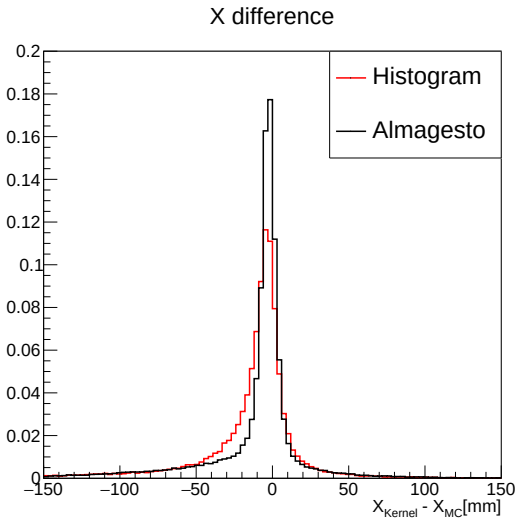


Figure 4.26: Difference between true and reconstructed x variable for Almagesto offline reconstruction (black) and Histogram kernel (red). Both plots are normalized to unity.

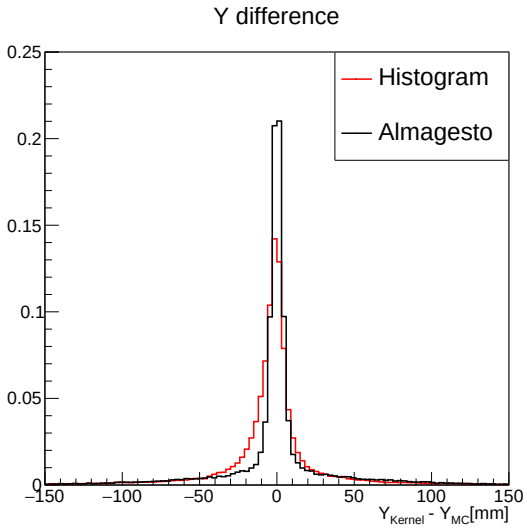


Figure 4.27: Difference between true and reconstructed y variable for Almagesto offline reconstruction (black) and Histogram kernel (red). Both plots are normalized to unity.

Chapter 5

The $K^+ \to \pi^- \ell^+ \ell^+$ decay.

Contents

Lepton flavour and lepton number are accidentally conserved quantities in the Standard Model. Unlike other quantum numbers, e.g the electric charge, their conservation is not imposed by theory as consequence of a global gauge symmetry (Noether's theorem).

While lepton flavour violation (LFV) has been observed in neutrino mixing [54], leading to first evidence to physics Beyond the Standard Model, Lepton Number Violation has never been observed. In particular, process violating lepton number by two units ($\Delta L = 2$) are unique tools to probe the Dirac or Majorana nature of neutrinos and their origin of its masses.

5.1 Massive neutrinos

Neutrinos are strictly massless in SM, and a neutrino mass is not allowed in the SM lagrangian $\mathcal{L}_{SM}$, due to the absence of right-handed neutrino states. However since the observation of neutrino oscillations has demonstrated their

massive nature, right-handed neutrinos must be included. This extension leads to an ambiguity about neutrino nature. In fact neutrino can be either Dirac or Majorana particles, and the description of right-handed neutrino states, as well as the neutrino mass term, depends on this.

In the Standard Model, neutrino masses can be introduced in exactly same way as for the up-type quarks using the conjugate Higgs doublet. In this case, after symmetry breaking, the gauge invariant *Dirac mass* term for a neutrino is:

$$\mathcal{L}_D = -m_D \left(\bar{\nu}_R \nu_L + \bar{\nu}_L \nu_R\right) \tag{5.1}$$

If this is the origin of the masses, then a right-handed chiral neutrino exist. However the neutrino masses are very much smaller than the masses of the other fermions, suggesting that another mechanism for generating neutrino mass might be possible. Because the *right-handed neutrinos* and *left-handed antineutrinos* transform as a singlets under the Standard Model gauge transformation, any additional terms in the Lagrangian formed from these field alone can be added to the Lagrangian without breaking the gauge invariance.

So we can introduce the Majorana mass term

$$\mathcal{L}_M = -\frac{1}{2} M \left(\overline{\nu_R^c} \nu_R + \overline{\nu_R} \nu_R^c\right) \tag{5.2}$$

where ν_R is the solution of the Majorana equation and $\overline{\nu_R^c}$ corresponds to the left-handed antineutrino. The Majorana mass term is formed from right handed-neutrino fields and left-handed antineutrino field, so it respects the local invariance of the Standard Model.

No terms of the type $\bar{\nu}_L \nu_L^c$ can be added, because no renormalizable singlet under hypercharge and weak isospin can be formed using these doublet components in the SM Lagrangian[48].

5.1.1 The seesaw mechanism

The most general renormalisable Lagrangian for neutrino masses includes both Dirac and Majorana terms is:

$$\mathcal{L}_{DM} = -\frac{1}{2} \begin{pmatrix} \overline{\nu}_L & \overline{\nu_R^c} \end{pmatrix} \begin{pmatrix} 0 & m_D \\ m_D & M \end{pmatrix} \begin{pmatrix} \nu_L^c \\ \nu_R \end{pmatrix} + h.c. \tag{5.3}$$

The physical states of this system can be obtained from the basis in which the mass matrix is diagonal. Hence, in this model, the masses of the physical neutrinos would be:

$$m_\pm = \frac{M \pm \sqrt{M^2 + 4m_D^2}}{2} = \frac{M \pm M\sqrt{1 + 4m_D^2/M^2}}{2} \tag{5.4}$$

Now if the Majorana mass M is much greater than the Dirac mass m_D

$$m_{\pm} \approx \frac{1}{2}M \pm \frac{1}{2}\left(M + \frac{2m_D^2}{M}\right) \tag{5.5}$$

giving a light neutrino state[1] (ν) and heavy neutrino state (N) with masses

$$m_\nu \approx -\frac{m_D^2}{M} \quad \text{and} \quad m_N \approx M \tag{5.6}$$

There are various models supporting which mass to use for the Majorana neutrino M. In GUT-seesaw model[49], the Majorana mass term is of the order of $M_{GUT} \sim 10^{14}$ GeV/c^2, while in ElectroWeak seesaw model[10] the Majorana mass is related to (unknown) electroweak symmetry breaking physics, and M is of the order of $0.1 \div 1$ TeV/c^2. In each case, if a Majorana mass term exists the seesaw mechanism predicts that for each of the three neutrino generations, there is a very light neutrino with a mass much smaller than the other Standard Model fermions and a very massive neutrino state $m_N \simeq M$.

In this scheme $m_\nu \propto m_D^2$ and if we use for each neutrino family the correspondent lepton mass for m_D

$$m_{\nu_e} : m_{\nu_\mu} : m_{\nu_\tau} = m_e^2 : m_\mu^2 : m_\tau^2 \tag{5.7}$$

So equations 5.7 predict a hierarchy of neutrino mass, so in this scenario we have

$$m_{\nu_e} < m_{\nu_\mu} < m_{\nu_\tau} \tag{5.8}$$

Moreover the physical neutrino state obtained from the eigenvalues of mass matrix are:

$$\nu = \cos\theta(\nu_L + \nu_L^c) - \sin\theta(\nu_R + \nu_R^c) \tag{5.9a}$$
$$N = \cos\theta(\nu_R + \nu_R^c) + \sin\theta(\nu_L + \nu_L^c) \tag{5.9b}$$

where $\tan\theta = m_D/M$, so the effect of introducing a Majorana mass term is to reduce the weak-charged current couplings of light neutrino states by a factor $\cos\theta$. However for $m_D \ll M$, the neutrino states are:

$$\nu \approx (\nu_L + \nu_L^c) - \frac{m_D}{M}(\nu_R + \nu_R^c) \tag{5.10a}$$

$$N \approx (\nu_R + \nu_R^c) + \frac{m_D}{M}(\nu_L + \nu_L^c) \tag{5.10b}$$

[1] The minus sing for the mass of the light neutrino in 5.6 can be absorbed in field definition.

and the couplings of light neutrinos are essentially the same of those of Standard Model. This is the

Because Majorana mass term provides coupling between a particle and an antiparticle, and a $|\Delta L = 2|$ transition is possible. The corresponding Majorana mass term for the electron would allow the $e^+ \leftrightarrow e^-$ transition, violating charge conservation. Because neutrinos are neutral, this problem doesn't exist, and they can be their own antiparticles.

In the specific case of K^+, the existence of a Majorana neutrino allows the decays $K^+ \to \pi^-\ell^+\ell^+$(the lowest order Feynman diagram contributing to this decay are shown in Fig. 5.1).

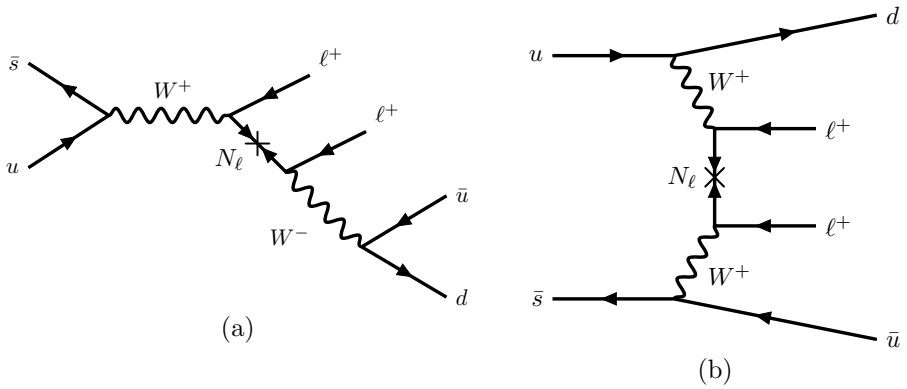


Figure 5.1: The two main Feynman diagram describing the massive Majorana neutrinos contributions to the process $K^+ \to \pi^-\ell^+\ell^+$. Because Majorana neutrinos are their own antiparticles they carrie a $\Delta L = 2$

5.2 Previous searches for $K^+ \to \pi^-\ell^+\ell^+$

In discuss previous years, many searches for this rare decay were performed, I separately the events with muons and those with the positrons.

5.2.1 Searches for $K^+ \to \pi^-\mu^+\mu^+$

The first upper limit for the $K^+ \to \pi^-\mu^+\mu^+$ decay was set at the Brookhaven National Laboratory as[47]:

$$K^+ \to \pi^-\mu^+\mu^+ < 1.5 \cdot 10^{-4} \quad (90\% C.L.) \quad (1992) \tag{5.11}$$

After 8 years the limit was improved by the E865 experiment to[9]:

$$K^+ \to \pi^-\mu^+\mu^+ < 3.0 \cdot 10^{-9} \quad (90\% C.L.) \quad (2000) \tag{5.12}$$

The most recent upper bound limit for this branching ratio was set by the NA48/2 collaboration at[31]:

$$K^+ \to \pi^-\mu^+\mu^+ < 1.1 \cdot 10^{-9} \quad (90\% C.L.) \quad (2011) \tag{5.13}$$

5.2.2 Searches for $K^+ \to \pi^- e^+ e^+$

The first limit for this decay is dated 1968 and was based of the CP conjugate $K^- \to \pi^+ e^- e^-$ decay[26]:

$$K^+ \to \pi^- e^+ e^+ < 1.5 \cdot 10^{-4} \quad (90\% C.L.) \quad (1968) \tag{5.14}$$

The first dedicated search for the $K^+ \to \pi^- e^+ e^+$ decay dates to 1976, at CERN PS experiment devoted to the measurement of K_{e4} decay[29]:

$$K^+ \to \pi^- e^+ e^+ < 9.2 \cdot 10^{-9} \quad (90\% C.L.) \quad (1976) \tag{5.15}$$

The limit on this branching ratio was improved only after 24 years by the E865 experiment at Brookheaven National Laboratory, the new limits is [9]:

$$K^+ \to \pi^- e^+ e^+ < 6.4 \cdot 10^{-10} \quad (90\% C.L.) \quad (2000) \tag{5.16}$$

NA62 $K^+ \to \pi^-\ell^+\ell^+$ sensitivity

Due to this high K^+ statistic the NA62 experiment could investigate the existence of these decays. With 10^{13} kaon decay collected in two years and with a fraction of 3 ring events in RICH acceptance of 31.87% and 32.14% for $K^+ \to \pi^+\mu^+\mu^+(K_{\mu\mu\pi})$ and $K^+ \to \pi^+ e^+ e^+(K_{ee\pi})$ respectively, if the the trigger efficiency is equal to 1 and without considering the background, the single event sensitivity aspected for the two decays is $K_{\mu\mu\pi} = 2.6 \cdot 10^{-13}$ and $K_{ee\pi} = 3.1 \cdot 10^{-13}$.

If no event are detected an upper limit of the two decays could be set which is

$$K_{\mu\mu\pi} < 9.57 \cdot 10^{-13} \quad (95\% C.L)$$

and

$$K_{ee\pi} < 11.41 \cdot 10^{-13} \quad (95\% C.L)$$

improving in the best case scenario the actual limits at least by a factor 100.

5.3 Trigger for $K^+ \to \pi^-\ell^+\ell^+$

The NA62 geometry and the standard trigger used for the experiment are designed for collect the largest sample as possible of $K^+ \to \pi^+\nu\bar{\nu}$ decay, so the standard RICH trigger select mostly events with only one charged track and few events with more than one charged track. So the selection of decays like the $K^+ \to \pi^-\ell^+\ell^+$ with the standard trigger is difficult, for this scope I studied if the use GPUs can improves it.

5.3.1 Separation between single and multi-ring events

Before all I studied how Histogram Kernel separates multi-ring and the single-ring events, this is a fundamental request for the searches of this rare decay, Tabs. 5.1 show the fraction of collected events for the main single-charge decays of kaon and the multi-charge decays of interest in the RICH acceptance. The efficiency on multi-ring events isn't very high, but combined with the rejection power of the single ring events, make the use of GPUs advantageous to separate efficiently the two types of events.

Decay	3-Rings
$\mu^+\nu_\mu$	0.62
$\pi^+\pi^0$	0.82
$\pi^0 e^+\nu_e$	0.91
$\pi^0\mu^+\nu_\mu$	0.78
$\pi^+\pi^0\pi^0$	0.76

Decay	3-Rings
$\pi^+\pi^+\pi^-$	59.44
$\pi^-\mu^+\mu^+$	71.38
$\pi^- e^+ e^+$	59.84

Table 5.1: Left: fraction of single-track events reconstructed with 3-rings according to the Histogram kernel. Data obtained with 15000 MC events for each decays. Right: fraction of the events with 3 tracks reconstructed as 3-ring according to Histogram kernel. Data obtained with 50000 MC events for each decays.

The results show above need to be compared with the ones obtained by the standard RICH L0 trigger for what concern the single-ring and multi-ring separation.

So I studied the performances of the standard L0 trigger of the RICH. The Multi-Ring trigger conditions R2 OR R3, where R2 and R3 are the number of SuperCell (digital OR of PMTs) hits, $9 \div 32$ for $R2$ and $33 \div 59$ for $R3$, Figure 5.2 shows the SC hits, for the principal kaon decays and for $K^+ \to \pi^-\ell^+\ell^+$ channels. Tab. 5.2 and Tab. 5.3 show the results of the simulations. One can see in Tab 5.2, Tab 5.3 and Fig. 5.2 if data are triggered with only R2 condition, the multi-rings event will be overwhelmed by the ones with one ring in the RICH due to the limited rejection power of the RICH. If trigger

is done with the R3 condition only the rejection on single-rings event is good, but the efficiency on the multi-ring events is too low. Comparing the results of Tab. 5.2, Tab. 5.3 and Tab.5.1 the use of GPUs improve the separation of single and multi ring events respect to the standard trigger of the experiment.

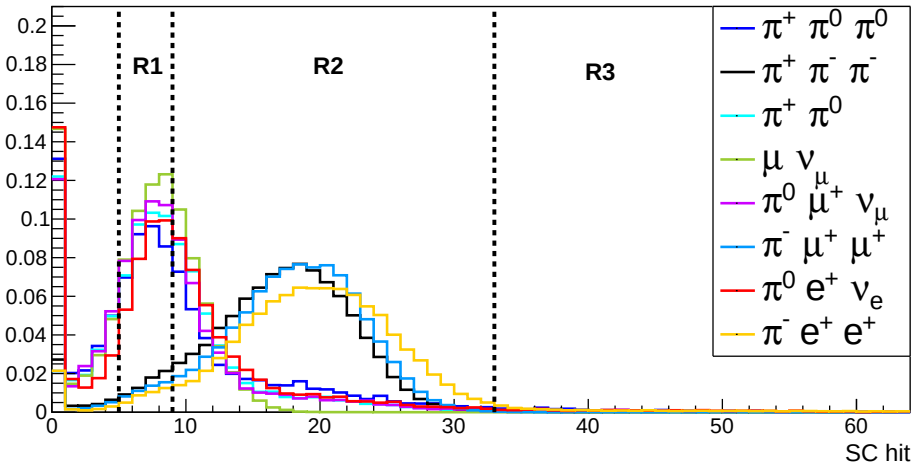


Figure 5.2: SuperCells hit multiplicity for principal kaon decays and the two $K^+ \to \pi^-\ell^+\ell^+$ channels. The dashed lines separate the R1,R2 and R3 trigger conditions. Data obtained with MC. All histograms are normalized to unity.

Decay	% R2	%R3
$\mu^+\nu_\mu$	27.32	<0.004
$\pi^+\pi^0$	43.48	2.09
$\pi^0 e^+\nu_e$	47.67	3.03
$\pi^0\mu^+\nu_\mu$	34.67	1.87
$\pi^+\pi^0\pi^0$	50.56	3.62

Table 5.2: Fraction of the single ring events satisfying the multi-ring condition, data obtained with 15000 MC events for each decays.

Decay	%R2	%R3	% True R2	% True R3	% 3-rings $\to$ R2
$\pi^+\pi^+\pi^-$	86.77	<3.7	43.78	<1.38	39.51
$\pi^-\mu^+\mu^+$	89.88	2.09	39.38	<1.85	51.34
$\pi^-e^+e^+$	89.44	2.25	44.43	<0.009	35.93

Table 5.3: The 2nd and 3rd column show fraction of the multi-ring events satisfying the R2/R3 conditions, 4th and 5th column show the fraction events triggered with R2/R3 which have 2/3 rings, 6th column fraction of 3-ring events in R2 triggered events .Data obtained with 50000 MC events for each decays.

5.3.2 The $K^+ \to \pi^+\pi^+\pi^-$ background

For distinguish in the multi-ring event $K^+ \to \pi^-\ell^+\ell^+$ from the main multi-ring background $K^+ \to \pi^+\pi^+\pi^-(K_{3\pi})$, I try to used as discriminating variable the sum of the radii of the ring reconstructed, but one can see in Fig.5.3 that the distributions (for 2 or 3 rings found within the RICH acceptance) are nearly the same,and any selection on the sum of radii couldn't separate efficiently the events.

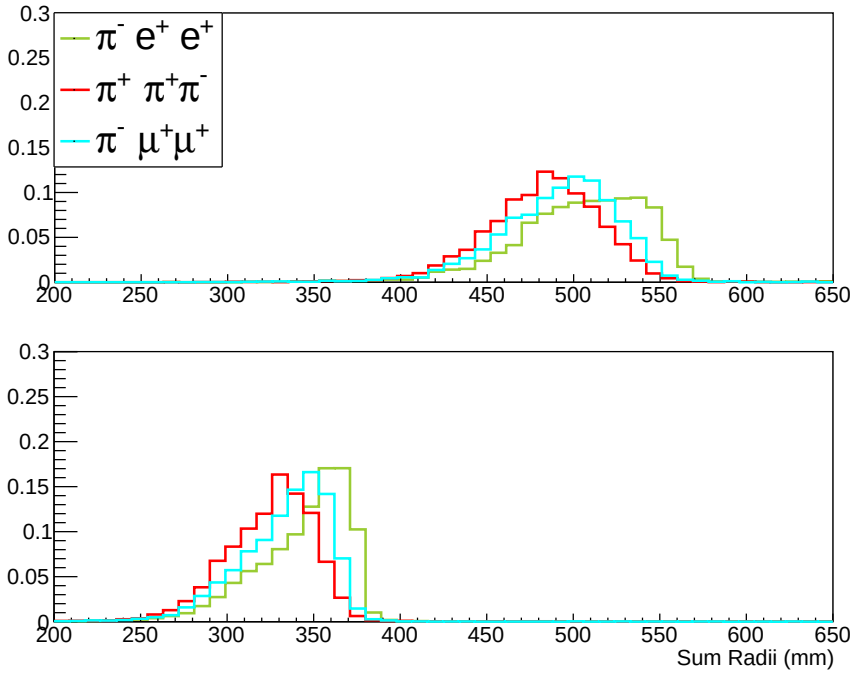


Figure 5.3: Top: sum of radii for events with 3 rings identified in the RICH acceptance, data obtained with MC. Bottom: sum of radii for events with 2 rings, data obtained with MC. All the histograms are normalized to unity.

I try to use the maximum radius of events with 3 rings, also in this case the distribution are too similar (see Fig. 5.4) to be used for a very selective reduction of the $K_{3\pi}$ background at L0 level.

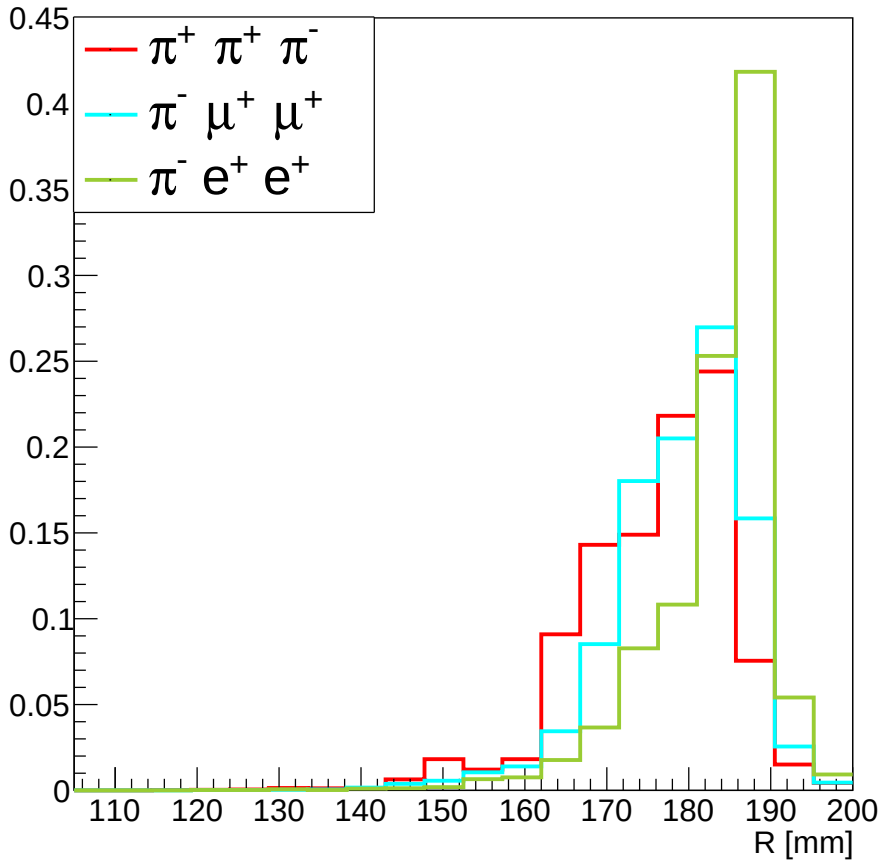


Figure 5.4: Distribution for the maximum radius in the 3 rings events. All the histograms in figure are normalized to unity.

As last solutions I select all the events with 3 rings, whit the 2 maximum radius exceeding 180mm the results obtained are show in Tab 5.4, ever in this case a efficiently rejection of the background can be performed.

Decay	%3-Rings
$\pi^+\pi^+\pi^-$	4.71
$\pi^-\mu^+\mu^+$	8.21
$\pi^-e^+e^+$	28.21

Table 5.4: Fraction of 3-rings events included in the RICH acceptance with the maximum radius and the second maximum radius exceeding the 180mm and 170mm respectively.

At the end I studied how the combined use of GPUs and the standard L0 trigger respect to only L0 trigger multiplicity or only GPUs trigger affect the multi-ring trigger rate of the RICH for the main kaon decays, the results are shown in Tab 5.5. One can see how the use of GPUs allows to reduce the rate with respect to the standard L0 trigger at least by a factor 20. The results obtained shows how using only the informations of the RICH it is not possible distinguish efficiently $K^+ \to \pi^-\ell^+\ell^+$ from $K^+ \to \pi^+\pi^+\pi^-$.

Decay	L0 (kHz)	GPUs (kHz)	L0|GPU (kHz)
$\mu^+\nu_\mu$	1909	43	11
$\pi^+\pi^0$	1035	18	8
$\pi^0e^+\nu_e$	282	5	3
$\pi^0\mu^+\nu_\mu$	142	3	1
$\pi^+\pi^0\pi^0$	105	15	1
$\pi^+\pi^+\pi^-$	533	110	96
Total	4026	194	120

Table 5.5: The table show the different trigger rates for the main kaon decays select with the multi-ring condition, for an input rate of 11 MHz. The use of only GPUs or GPUs combined with L0 reduce by a factor 20 the trigger rate for the multi-ring events.

The result obtained in this chapter are summarized in Tab 5.6, which compare GPUs and the standard L0 trigger of the RICH for the multi-ring events. One can see how the use of the GPUs reduce the trigger rate for multi-rings event, improve the efficiency and increase the purity of the data collected. An improvement on the various aspect of the trigger is obtained with the GPUs.

	L0 trigger	GPU
$K_{\mu\mu\pi}$ efficiency%	51.34	71.38
$K_{ee\pi}$ efficiency %	35.93	59.84
Trigger 3-rings (kHz)	4026	194
Purity 3-rings %	5.22	39.37

Table 5.6: The 1st and 2nd rows show the different fraction of events collected with 3 rings in acceptance for the $K^+ \to \pi^-\ell^+\ell^+$ decays, the 3rd row shows the trigger rate with the multi-ring condition, 4th row shows the fraction of the event triggered with multi-ring conditions which have effectively 3 rings, the last two rows are based on the $K_{\pi\pi\pi}$ data only.

Chapter 6

Conclusions

This thesis invstigates one of the first approach for a real-time trigger based on GPUs.

For using the algorithm in the L0 trigger level of NA62 experiments a computation time < 1 ms per event and resolutions comparable to the ones achieved by the offline reconstruction are required .

The results

$$\Delta t_{evt} = 0.38 \pm 0.06 \ \mu\text{s} \tag{6.1}$$

improving the previous results obtained in [33] which was:

$$\Delta t_{evt} = 0.97 \pm 0.02 \ \mu\text{s} \tag{6.2}$$

and the measured resolutions for the rings parameters are

Parameter	σ_{Alm} [mm]	σ_{Hist} [mm]
r	2.05	2.56
x	2.07	3.75
y	2.69	3.67

Parameter	σ_{Alm} [mm]	σ_{Hist} [mm]
r	3.08	3.85
x	4.92	6.18
y	4.12	6.46

Table 6.1: Left side: resolution achieved for the single ring event. Right side: resolution achieved for the multi ring events. The second column shows the Almagesto(offline) resolutions, the third shows the Histogram resolutions(online).

The results above show how the algorithm fulfill the requirements and is suitable to be used in the L0 trigger level of the experiment.

I also try to use the GPUs as trigger to $K^+ \to \pi^-\ell^+\ell^+$ decays, using GPUs to separate these decays from the main background $K^+ \to \pi^+\pi^+\pi^-$, the results obtained in Chapter 5 show how with GPUs is possible separate single-ring and multi-rings events, but the resolutions are not sufficient to separate the $K^+ \to \pi^-\ell^+\ell^+$ decays from the $K^+ \to \pi^+\pi^+\pi^-$. In any case the

trigger based on GPUs improve the selection of the multi-ring events.

This work shows that alternative trigger designs are feasible for the NA62 experiment and is a starting point to introduce the use of flexible GPU-based real-time triggers in High Energy Physics.

6.1 Possible improvements and outlook

As seen in section 4.3 and section 5.3 the resolution of the algorithm need to be improved in order to obtain a more efficient trigger, which could not only separate better multiple ring events from single-ring events, but also discriminate the multi-ring events based on kinematical constraints. A way to do this might be to change the shape or the geometry of the grid used. Such as the use of a grid with squares of different sizes, according to the frequency of the ring centers, finer where there are more and larger elsewhere.

The algorithm was optimized, but surely there is room for improvement, especially when new generation of GPUs (Maxwell or Pascal) will be used.

The time performances of the trigger, satisfactory up to 450 events per CLOP, need to be improved. The plots in section 4.2 show how the computing time of the events scale almost linearly with CLOP size, and at full intensity, the number of events per CLOPs will be above than 450. In order to achieve better timing performance a multi-GPUs trigger system should be implemented. A simple option could be used CLOPs smaller by a factor equal to the number of GPUs used, and each of these smaller CLOPs will be sent to a different GPUs for computing. In this way the times would be reduced by a factor equal to the number of GPUs used.

Appendices

Appendix A

Crawford Algorithm

Let us define

$$f_i \equiv (x_i - a)^2 + (y_i - b)^2 - R^2 \tag{A.1}$$

The simplest algebraic method to fit a circle $(x_i - a)^2 + (y_i - b)^2 + R^2 = 0$ to a set of points is to minimize the algebraic expression [19]

$$\mathcal{F}_1 = \sum_i f_i^2 \tag{A.2}$$

We will solve the problem in a suitable (u, v) coordinate system, and then transform the solutions back to the original system (x, y) [19]. Writing

$$\bar{x} = \frac{1}{n}\sum_i x_i \qquad \bar{y} = \frac{1}{n}\sum_i y_i \tag{A.3}$$

$$u_i \equiv x_i - \bar{x} \qquad v_i \equiv y_i - \bar{y} \tag{A.4}$$

we obtain:

$$\mathcal{F}_1{}' = \sum_i g_i^2 \tag{A.5}$$

$$g_i = (u_i - u_c)^2 + (v_i - v_c)^2 - R^2 \tag{A.6}$$

where (u_c, v_c) are the coordinates of the centre of the circle, as computed in the (u, v) frame:

$$u_c = a - \bar{x} \tag{A.7}$$
$$v_c = b - \bar{y} \tag{A.8}$$

In order to minimize $\mathcal{F}_1{}'$ we compute its derivatives with respect to the parameters u_c, v_c and R^2:

$$\frac{\partial \mathcal{F}_1{}'}{\partial R^2} = 2 \sum_i g_i \frac{\partial g_i}{\partial R^2} = -2 \sum_i g_i \tag{A.9}$$
$$\frac{\partial \mathcal{F}_1{}'}{\partial u_c} = 2 \sum_i g_i \frac{\partial g_i}{\partial u_c} = -4 \sum_i u_i \, g_i + 4 u_c \sum_i g_i \tag{A.10}$$
$$\frac{\partial \mathcal{F}_1{}'}{\partial v_c} = 2 \sum_i g_i \frac{\partial g_i}{\partial v_c} = -4 \sum_i v_i \, g_i + 4 u_c \sum_i g_i \tag{A.11}$$

and then set them to zero. This system gives the unique solution

$$\sum_i g_i = 0 \qquad \sum_i u_i \, g_i = 0 \qquad \sum_i v_i \, g_i = 0 \tag{A.12}$$

Now let

$$S_u \equiv \sum_i u_i \qquad S_{uu} \equiv \sum_i u_i^2 \qquad S_{uuu} \equiv \sum_i u_i^3 \quad \text{etc.} \tag{A.13}$$

and similarly for S_v, S_{uv} and so on, where $S_u = S_v = 0$ by definition.

Adopting this notation, if we expand Eqns. A.12 we obtain the system

$$u_c S_{uu} + v_c S_{uv} = \frac{1}{2} \left(S_{uuu} + S_{uvv} \right) \tag{A.14}$$
$$u_c S_{uv} + v_c S_{vv} = \frac{1}{2} \left(S_{vvv} + S_{uuv} \right) \tag{A.15}$$
$$n \left(u_c^2 + v_c^2 - R^2 \right) + S_{uu} + S_{vv} = 0 \tag{A.16}$$

which in turn yields the solutions

$$u_c = \frac{\frac{S_{uv}}{2}\left(S_{vvv} + S_{uuv}\right) - \frac{S_{vv}}{2}\left(S_{uuu} + S_{uvv}\right)}{S_{uv}^2 - S_{vv}^2} \tag{A.17}$$

$$v_c = \frac{\frac{1}{2}\left(S_{uuu} + S_{uvv}\right) - u_c S_{uu}}{S_{uv}} \tag{A.18}$$

$$R^2 = u_c^2 + v_c^2 + \frac{S_{uu} + S_{vv}}{n} \tag{A.19}$$

The centre of the circle in the original coordinate system will be $(a, b) = (u_c, v_c) + (\bar{x}, \bar{y})$.

Appendix B

GeForce Titan Specifications

Device	GeForce GTX TITAN
CUDA Driver Version / Runtime Version	7.0 / 6.5
CUDA Capability Major/Minor version number	3.5
Total amount of global memory	6143 MBytes (6441730048 bytes)
(14) Multiprocessors, (192) CUDA Cores/MP	2688 CUDA Cores
GPU Clock rate	876 MHz (0.88 GHz)
Memory Clock rate	3004 Mhz
Memory Bus Width	384-bit
L2 Cache Size	1572864 bytes
Total amount of constant memory	65536 bytes
Total amount of shared memory per block	49152 bytes
Total number of registers available per block	65536
Warp size	32
Maximum number of threads per multiprocessor	2048
Maximum number of threads per block	1024
Max dimension size of a thread block (x,y,z)	(1024, 1024, 64)
Max dimension size of a grid size (x,y,z)	(2147483647, 65535, 65535)
Maximum memory pitch	2147483647 bytes
Texture alignment	512 bytes
Concurrent copy and kernel execution	Yes with 1 copy engine(s)
Run time limit on kernels	Yes
Integrated GPU sharing Host Memory	No
Support host page-locked memory mapping	Yes
Alignment requirement for Surfaces	Yes
Device has ECC support	Disabled
Device supports Unified Addressing (UVA)	Yes
Device PCI Bus ID / PCI location ID	3 / 0

Table B.1: Technical characteristics of the GPU on which this project was implemented and tested. Data obtained with the `deviceQuery` script provided with the CUDA libraries.

Bibliography

[1] NaNet overview. http://apegate.roma1.infn.it/mediawiki/index.php/NaNet_overview.

[2] New Event Reconstruction Algorithm for Super-Kamiokande Water Cherenkov Detector. http://indico.ipmu.jp/indico/getFile.py/access?contribId=20&sessionId=10&resId=0&materialId=slides&confId=10.

[3] Trackless ring identification and pattern recognition in ring imaging cherenkov (rich) detectors. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 560(2):621 – 632, 2006.

[4] Buras A.J. Weak Hamiltonian, CP Violation and Rare Decays. *ArXiv High Energy Physics - Phenomenology e-prints*, pages 204–210, June 1998.

[5] Wolfgang Altmannshofer, Andrzej J. Buras, Stefania Gori, Paride Paradisi, and David M. Straub. Anatomy and Phenomenology of FCNC and CPV Effects in SUSY Theories. *Nucl. Phys.*, B830:17–94, 2010.

[6] Roberto Ammendola, Andrea Biagioni, Riccardo Fantechi, Ottorino Frezza, Gianluca Lamanna, Francesca Lo Cicero, Alessandro Lonardo, Pier Stanislao Paolucci, Felice Pantaleo, Roberto Piandani, Luca Pontisso, Davide Rossetti, Francesco Simula, Marco Sozzi, Laura Tosoratto, and Piero Vicini. Nanet: a low-latency nic enabling gpu-based, real-time low level trigger systems. *Journal of Physics: Conference Series*, 513(1):012018, 2014.

[7] B Angelucci, E Pedreschi, M Sozzi, and F Spinella. Tel62: an integrated trigger and data acquisition board. *Journal of Instrumentation*, 7(02):C02046, 2012.

[8] V. V. Anisimovsky et al. Improved measurement of the $K \to \pi\nu\bar{\nu}$ branching ratio. *Phys. Rev. Lett.*, 93:031801, 2004.

[9] R. Appel, G. S. Atoyan, B. Bassalleck, D. R. Bergman, N. Cheung, S. Dhawan, H. Do, J. Egger, S. Eilerts, H. Fischer, W. Herold, V. V. Issakov, H. Kaspar, D. E. Kraus, D. M. Lazarus, P. Lichard, J. Lowe, J. Lozano, H. Ma, W. Majid, W. Menzel, S. Pislak, A. A. Poblaguev, P. Rehak, A. Sher, J. A. Thompson, P. Truöl, and M. E. Zeller. Search for lepton flavor violation in k^+ decays into a charged pion and two leptons. *Phys. Rev. Lett.*, 85:2877–2880, Oct 2000.

[10] Thomas Appelquist and Robert Shrock. Neutrino masses in theories with dynamical electroweak symmetry breaking. *Physics Letters B*, 548(3-4):204 – 214, 2002.

[11] A. V. Artamonov et al. New measurement of the $K^+ \to \pi^+\nu\bar{\nu}$ branching ratio. *Phys. Rev. Lett.*, 101:191802, 2008.

[12] A. V. Artamonov and other. Study of the decay $K^+ \to \pi^+\nu\bar{\nu}$ in the momentum region $140 < p_\pi < 199$ MeV/c. *Phys. Rev. D*, 79:092004, May 2009.

[13] Y. Asano, E. Kikutani, S. Kurokawa, T. Miyachi, M. Miyajima, Y. Nagashima, T. Shinkawa, S. Sugimoto, and Y. Yoshimura. Search for a rare decay mode $K \to \pi\nu\bar{\nu}$ and axion . *Physics Letters B*, 107(1-2):159–162, December 1981.

[14] Henry W Atherton, Claude Bovet, Niels T Doble, L Piemontese, Alfredo Placci, Massimo Placidi, David E Plane, Max Reinharz, Edouard Rossa, and G Von Holtey. *Precise measurements of particle production by 400 GeV/c protons on beryllium targets*. CERN, Geneva, 1980.

[15] A. Badalov, D. Cámpora, N. Neufeld, and X. Vilasís-Cardona. Lhcb gpu acceleration project. *Journal of Instrumentation*, 11(01):P01001, 2016.

[16] Monika Blanke, Andrzej J. Buras, Bjorn Duling, Stefan Recksiegel, and Cecilia Tarantino. FCNC Processes in the Littlest Higgs Model with T-Parity: a 2009 Look. *Acta Phys. Polon.*, B41:657–683, 2010.

[17] Joachim Brod, Martin Gorbahn, and Emmanuel Stamou. Two-loop electroweak corrections for the $K \to \pi\nu\bar{\nu}$ decays. *Phys. Rev. D*, 83:034030, Feb 2011.

[18] Gerhard Buchalla and Andrzej J. Buras. QCD corrections to the anti-s d Z vertex for arbitrary top quark mass. *Nucl. Phys. B*, 398:285–300, 1993.

[19] R. Bullock. *Least-Squares Circle Fit.* Developmental Testbed Center, 2006.

[20] Andrzej J. Buras, Fulvia De Fazio, and Jennifer Girrbach. $\Delta I = 1/2$ rule, ε'/ε and $K \to \pi\nu\bar{\nu}$ in $Z'(Z)$ and G' models with FCNC quark couplings. *Eur. Phys. J.*, C74(7):2950, 2014.

[21] Andrzej J. Buras, Bjorn Duling, Thorsten Feldmann, Tillmann Heidsieck, Christoph Promberger, and Stefan Recksiegel. Patterns of Flavour Violation in the Presence of a Fourth Generation of Quarks and Leptons. *JHEP*, 09:106, 2010.

[22] Nicola Cabibbo. Unitary symmetry and leptonic decays. *Phys. Rev. Lett.*, 10:531–533, Jun 1963.

[23] G. D. Cable, R. H. Hildebrand, C. Y. Pang, and R. Stiening. Search for rare K^+ decays. ii. $K^+ \to \pi^+\nu\bar{\nu}$. *Phys. Rev. D*, 8:3807–3812, December 1973.

[24] U. Camerini, D. Ljung, M. Sheaff, and D. Cline. Experimental search for semileptonic neutrino neutral currents. *Phys. Rev. Lett.*, 23:326–329, August 1969.

[25] Augusto Ceccucci. NA62/P-326 Status Report. Technical report, SPS Experiments Committee, CERN, November 2007.

[26] C. Y. Chang, G. B. Yodh, R. Ehrlich, R. Plano, and A. Zinchenko. Search for double beta decay of K^- meson. *Phys. Rev. Lett.*, 20:510–513, Mar 1968.

[27] "NVIDIA Corporation". CUDA toolkit. https://developer.nvidia.com/cuda-toolkit.

[28] J.F. Crawford. A non-iterative method for fitting circular arcs to measured points. *Nuclear Instruments and Methods in Physics Research*, 211(1):223 – 225, 1983.

[29] A.M. Diamant-Berger, P. Bloch, B. Devaux, N. Do-Duc, G. Marel, R. Turlay, P. Extermann, J. Fischer, O. Guisan, R. Mermod, L. Rosselet, and R. Sachot. Study of some rare decays of the k+ meson. *Physics Letters B*, 62(4):485 – 490, 1976.

[30] J. M. Flynn, M. Paulini, and S. Willocq. WG2 Conveners' Report: V_{td} and V_{ts}, B-Bbar mixing, radiative penguin and rare (semi)leptonic decays. Technical report, November 2003.

[31] Evgueni Goudzovski. Searches for lepton flavour and lepton number violation in kaon decays. pages 255–262, 2011.

[32] M. W. Govett, J. Middlecoff, and T. Henderson. Running the nim next-generation weather model on gpus. In *Cluster, Cloud and Grid Computing (CCGrid), 2010 10th IEEE/ACM International Conference on*, pages 792–796, May 2010.

[33] Elena Graverini. A GPU-based real time trigger for rare kaon decays at NA62. Master's thesis, Università di Pisa, Ottobre 2013.

[34] Yuval Grossman and Yosef Nir. $K_L^0 \to \pi^0 \nu\bar{\nu}$ beyond the standard model . *Physics Letters B*, 398:163 – 168, 1997.

[35] "Khronos group". Open CL. https://www.khronos.org/opencl/.

[36] TDAQ Working group. NA62 data formats. https://twiki.cern.ch/twiki/pub/NA62/TdaqSystem/DataFormats.pdf.

[37] F Hahn, F Ambrosino, A Ceccucci, H Danielsson, N Doble, F Fantechi, A Kluge, C Lazzeroni, M Lenti, G Ruggiero, M Sozzi, P Valente, and R Wanke. NA62: Technical Design Document. Technical Report NA62-10-07, CERN, Geneva, December 2010.

[38] Gino Isidori, Federico Mescia, and Christopher Smith. Light-quark loops in $K \to \pi\nu\bar{\nu}$. *Nucl. Phys. B*, 718:319–338, 2005.

[39] E. Iwai. Status and prospects of J-PARC KOTO experiment. *Nucl. Phys. Proc. Suppl.*, 233:279–283, 2012.

[40] Jaffe, David E. and Youssef, Saul. Bayesian estimate of the effect of $B^0\bar{B}^0$ mixing measurements on the CKM matrix elements. *Comput. Phys. Commun.*, 101:206, 1997.

[41] S Kama, J Augusto Soares, J Baines, M Bauce, T Bold, P Conde Muino, D Emeliyanov, R Goncalo, A Messina, M Negrini, L Rinaldi, A Sidoti, A Tavares Delgado, S Tupputi, and L Vaz Gil Lopes. Triggering events with GPUs at ATLAS. *Journal of Physics: Conference Series*, 664(9):092014, 2015.

[42] S A Kholodenko, A A Khudyakov, I Mannelli, V F Obraztsov, V D Samoylenko, V K Semenov, and V P Sugonyaev. Time resolution measurements of scintillating counters for a new na62 trigger charged hodoscope. *Journal of Instrumentation*, 9(09):C09002, 2014.

[43] Makoto Kobayashi and Toshihide Maskawa. CP Violation in the Renormalizable Theory of Weak Interaction. *Prog. Theor. Phys.*, 49:652–657, 1973.

[44] G. Lamanna. Almagest, a new trackless ring finding algorithm. *Nucl. Instrum. Meth.*, A766:241–244, 2014.

[45] G Lamanna, R Ammendola, M Bauce, A Biagioni, R Fantechi, M Fiorini, S Giagu, E Graverini, G Lamanna, A Lonardo, A Messina, F Pantaleo, P S Paolucci, R Piandani, M Rescigno, F Simula, M Sozzi, and P Vicini. Gpus for real-time processing in hep trigger systems. *Journal of Physics: Conference Series*, 513(1):012017, 2014.

[46] Gianluca Lamanna, Gianmaria Collazuol, and Marco Sozzi. GPUs for fast triggering and pattern matching at the CERN experiment NA62. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 628(1):457 – 460, 2011.

[47] Laurence S. Littenberg and Robert E. Shrock. Upper bounds on lepton-number violating meson decays. *Phys. Rev. Lett.*, 68:443–446, Jan 1992.

[48] Peter Minkowski. $\mu \to e\gamma$ of one out of 10^9 muon decays? *Physics Letters B*, 67(4):421 – 428, 1977.

[49] Rabindra N. Mohapatra and Goran Senjanović. Neutrino mass and spontaneous parity nonconservation. *Phys. Rev. Lett.*, 44:912–915, Apr 1980.

[50] NA48 Collaboration, A. Lai, D. Marras, L. Musa, A. Nappi, R. Batley, A. Bevan, R. S. Dosanjh, R. Galik, T. Gershon, and et al. The beam and detector for the NA48 neutral kaon CP violation experiment at CERN. *Nucl. Instrum. Meth. A*, 574:433–471, May 2007.

[51] NVIDIA Corporation. *NVIDIA CUDA C Best Practise Guide*, July 2013.

[52] NVIDIA Corporation. *NVIDIA CUDA C Programming Guide*, July 2013.

[53] B. Oancea, T. Andrei, and R. Mariana Dragoescu. Improving the performance of the linear systems solvers using CUDA. *ArXiv e-prints*, November 2015.

[54] K. A. Olive et al. Review of Particle Physics. *Chin. Phys.*, C38:090001, 2014.

[55] Jacopo Pinzino. Algoritmi di Trigger Paralleli per la Ricerca di Decadimenti Rari del Mesone K. Master's thesis, Università di Pisa, 2010/2011.

[56] Antonino Sergi. Recent results from Kaon Physics. Technical report, March 2013.

[57] Bob Velghe. GigaTracker, a Thin and Fast Silicon Pixels Tracker. In *RD13 - 11th International Conference on Large Scale Applications and Radiation Hardness of Semiconductor Detectors*, July 2013.

[58] F. T. Winter, M. A. Clark, R. G. Edwards, and B. Joó. A Framework for Lattice QCD Calculations on GPUs. 2014.

[59] Lincoln Wolfenstein. Parametrization of the Kobayashi-Maskawa matrix. *Phys. Rev. Lett.*, 51:1945–1947, Nov 1983.

Acknowledgements

I would like to express my gratitude to my supervisor for the patience and the attention he has shown me during this work.

A big thanks to Gianluca and Luca for all the helps and advices they give to me, during the development of the algorithm.

Thanks to Roberto and Jacopo for all the funny moments.

I would thanks Paolo for all the company and coffee-break we shared during these long months.

I cannot thank all my friend which have supported me during these years, thanks to Marta, Matteo, Alberto, Niccolò, Claudia, Simone, Cristina, Andrea, Alessandro, Mario and Leonardo.

A big thanks to the friends of my hometown, every time I see you it's like we had left the day before, so thanks to Alessio, Simone, Peppe, Chiara, Matteo, Bruscone and Valentina.

Thanks to my friend Pietro, which stimulate me with every type of question and for all the *Maxi-Calzoni* we take together.

Two special thanks, the first one is for Fabio, my are roommate by 5 years, we have passed many good moments and funny times in Via Parini 8, the one with the firefighters is one of my favourite.

The latter one is for Lorenzo, which I started this adventure together 6 years ago, thanks for all the Mexico-Switzerland during these years, for me will forever be the $Match$.

Thanks to my gramps for all the love they show me during these years, and a special thoughts to grandma Maria

Last but not least I would like to express my profound gratitude to my family, and especially to my parents. During these years they have encouraged and sustained me, with only one advice, the most valuable: *"You know..."*.

It is to them that this thesis is dedicated.

Ringraziamenti

Vorrei ringraziare il mio relatore per tutta la pazienza e l'attenzione che ha dimostrato nei miei confronti durante questo lavoro.

Ringrazio Gianluca e Luca per tutto l'aiuto ed i consigli che mi hanno durante lo sviluppo dell'algoritmo.

Grazie anche a Roberto e Jacopo per tutti i momenti divertenti passati insieme.

Grazie a Paolo, per tutta la compagnia e per le varie pause caffé passate insieme durante questi mesi.

Non posso non ringraziare tutti i miei amici che mi hanno su(o)pportato durante questi anni, grazie a Marta, Matteo, Alberto, Niccolò, Claudia, Simone, Cristina, Andrea, Alessandro, Mario e Leonardo.

Grazie ai miei amici di Francavilla, ogni volta che vi rivedo e come se ci fossimo lasciati solo il giorno prima, e quindi grazie ad Alessio, Simone, Peppe, Chiara, Matteo, Bruscone e Valentina.

Grazie al mio amico Pietro che mi stimola sempre con domande di tutti i tipi e per tutti i *Maxi-Calzoni* che abbiamo preso insieme.

Mi sento di fare due ringraziamenti speciali, il primo è per Fabio mio coinquilino da 5 anni ormai, abbiamo passato tanti bei momenti in Via Parini 8, quello che preferisco è sicuramente quello con i pompieri.

Il secondo è per Lorenzo con cui ho iniziato questa avventura insieme 6 anni fa, grazie per tutti i Messico-Svizzera di questi anni, per me resterà per sempre la *Partita*.

Grazie ai miei nonni per tutto l'affetto che mi hanno dimostrato in questi anni, ed un pensiero speciale a nonna Maria.

Infine per ultimo ma non per importanza, un grandissimo immenso grazie alla mia famiglia, specialmente ai miei genitori. Durante tutti questi anni mi hanno incoraggiato e sostenuto, sempre con un solo consiglio, il più prezioso: *"Tu sai..."*.

È a loro che dedico questa tesi.