



University of Pisa, Department of Physics
PhD course XXXIV

Measurements of the $\pi^0 \rightarrow e^+e^-\gamma$ decay with application of Deep Learning techniques

Candidato

Enrico Lari

Relatore

Prof. M. Sozzi

Abstract

We present the measurement of the transition form factor of the π^0 that we performed analyzing the Run1 data, collected by the NA62 experiment during the year 2017-2018. It accounts for the largest worldwide sample of π^0 Dalitz decays with 8.7×10^6 events, allowing to reduce the statistical uncertainty by a factor ≈ 1.8 with respect to the previous measurement [36]. While the statistical uncertainty is dominated by the statistics in the available MC sample, the largest uncertainty to the final result arises from the systematics due to the trigger emulation in the MC sample. Our result is

$$a \pm \delta a_{stat} \pm \delta a_{sys} = (3.56 \pm 0.29 \pm 1.35) \times 10^{-2} \quad (1)$$

We also discuss the measurement of the π^0 Dalitz branching ratio which is on hold due to the need of data processing at the experiment level.

We also report some studies of a new data-analysis technique based on Deep Learning methods which were performed while considering different approaches for the measurement, to investigate the possibility of fully performing an event selection using reconstructed quantities while minimizing any human input on the properties of the events being searched for.

A byproduct of this work is a new fine calibration method for the energy clusters of the electromagnetic calorimeter, now the default method used in the official data-processing of the NA62 experiment. We show how our method reached similar performances with respect to previous ones, without introducing any biases and with the significant advantage of requiring much less computational power and time.

As this work has been developed during the NA62 experiment shut-down, before the restarting of the new data-taking period in 2021, as an additional byproduct of the investigation of Deep-Learning techniques in high-energy physics, we also studied the possibility of a new algorithm for the L1 software trigger, implementing Deep Learning techniques to perform an unbiased particle identification exploiting the information coming from the NA62 Rich Cherenkov Counter. While the results of this study did not reach the efficiency levels required for its actual implementation in the experiment before the start of the new data-taking period, they represent an important basis for futures studies.

*To my grandmother,
who would have always wanted to study math*

Contents

1	The NA62 Experiment	10
1.1	Physics Motivations	10
1.2	The NA62 Detector	11
1.2.1	Tracking system	13
1.2.2	Veto system	14
1.2.3	Particle identification system (PID)	18
1.3	NA62 Trigger and Data Acquisition System (TDAQ)	21
1.4	NA62 reconstructed data structure and analysis framework	23
1.5	NA62 MC package	24
1.5.1	Overlaid MC samples	25
2	Measurement of the π^0 Transition Form Factor	26
2.1	The π^0 Dalitz decay	26
2.1.1	Dalitz MC generator	30
2.2	Experimental status	31
2.3	Analysis strategy	33
2.4	Event selection	35
2.4.1	Data sample	35
2.4.2	MC samples	36
2.4.3	Vertex selection	36
2.4.4	Cluster selection	37
2.4.5	Kinematic cuts	39
2.4.6	Offline trigger conditions	42
2.4.7	Background suppression	42
2.4.8	Trigger Efficiency	46
2.5	TFF Fit	56
2.5.1	Equipopulated bins	56
2.5.2	First Measurement	58
2.5.3	Inclusion of the biased signal sample	58
2.5.4	Improved Measurement and final result	59
2.6	Systematics	61
2.6.1	Trigger efficiency	61
2.6.2	Spectrometer track corrections	63
2.6.3	Accidental Vertices	63
2.6.4	Accidental Clusters	64
2.6.5	LKr cluster fine calibration	64
2.6.6	Kinematic Particle identification	64
2.6.7	Neglected MC sources	64

2.6.8	Overlay treatment	65
2.6.9	Stability Checks	65
2.7	Conclusions	75
2.8	The $\text{BR}(\pi^0 \rightarrow e^+e^-\gamma)$ measurement	75
2.9	Event filtering	76
2.9.1	Prospects	78
3	LKr energy fine calibration based on $\pi^0 \rightarrow e^+e^-\gamma$ decay	80
3.1	Introduction	80
3.2	Calibration computation and motivation for a π_D^0 based calibration .	80
3.3	π_D^0 selection	82
3.4	Fitting procedure	85
3.5	Comparison with standard calibration	85
3.6	Feedback from analyses	89
3.7	Status and prospects	92
4	Feasibility study of a Deep Learning algorithm for event classification by a seedless RICH circle detection	93
4.1	Introduction	93
4.2	Theory behind Deep Learning	94
4.3	Event generator	95
4.4	Classifier models	96
4.4.1	Fully connected Deep Neural Network classifier	97
4.4.2	Convolutional Neural Network classifier	112
4.5	Autoencoders	124
4.5.1	Conclusions	134
5	Perspectives of an automatized study of $\pi^0 \rightarrow e^+e^-\gamma$ decay through deep learning techniques	135
5.1	Introduction	135
5.2	Datatest design	135
5.3	Conclusions	140
6	Conclusions	145
A		147
A.1	Data/MC comparison	147
B		149
B.1	Performance indicators	149
B.2	Generalization of the Crawford fit	149

List of abbreviations

In the following Table we collect together some of the frequently used abbreviations that are used in the text.

Abbreviation	Meaning
General	
TFF	Transition Form Factor
BR	Branching Ratio
MC	Monte Carlo
ML	Machine Learning
DL	Deep Learning
NN	Neural Network
CNN	Convolutional Neural Network
L0 trigger	Level 0 trigger
L1 trigger	Level 1 trigger
π^0 Decays	
π_D^0	$\pi^0 \rightarrow e^+e^-\gamma$
$\pi_{\gamma\gamma}^0$	$\pi^0 \rightarrow \gamma\gamma$
Kaon decays	
$K_{2\pi}$	$K^+ \rightarrow \pi^+\pi^0$
$K_{2\pi D}$	$K^+ \rightarrow \pi^+\pi_D^0$
$K_{2\pi(\gamma\gamma)}$	$K^+ \rightarrow \pi^+\pi_{\gamma\gamma}^0$
$K_{3\pi}$	$K^+ \rightarrow \pi^+\pi^+\pi^-$
$K_{3\pi 0}$	$K^+ \rightarrow \pi^+\pi^0\pi^0$
$K_{3\pi 0 D}$	$K^+ \rightarrow \pi^+\pi^0\pi_D^0$
K_{e3}	$K^+ \rightarrow e^+\pi^0\nu_e$
$K_{\mu 3}$	$K^+ \rightarrow \mu^+\pi^0\nu_\mu$
K_{e3D}	$K^+ \rightarrow e^+\pi_D^0\nu_e$
$K_{\mu 3D}$	$K^+ \rightarrow \mu^+\pi_D^0\nu_\mu$

Table 1

Introduction

The present work has been carried out within the NA62 experiment, the current CERN research program for testing the Standard Model (SM) through the study of the rare kaon decays.

The thesis has a common thread: the $\pi^0 \rightarrow e^+e^-\gamma$ decay, also known as Dalitz decay, a well-studied decay mode described by QED together with some experimental information whose measurement we performed, but with large uncertainties on the quantities describing it, such as the branching ratio (with a relative uncertainty of 3%) and the transition form factor slope parameter, whose uncertainty is still dominated by small data samples available for the measurement. The large statistics of kaon decays ($O(10^{12})$), which was collected by NA62 in three years of running, and the fact that the second most-frequent decay channel of the K^+ is the $K^+ \rightarrow \pi^+\pi^0$ decay, provides a large sample of tagged π^0 decays, exceeding that of previous experiments, allowing a more precise measurements for the above quantities. For both studies we developed a common event selection, free from backgrounds. Then for what concerns the TFF measurement, we reused the technique of [36] to perform the final fit. However the use of a biased Monte Carlo sample, necessary to reduce the uncertainties related to the available statistics of simulated events, required some modification of their approach to reach the final result. Unfortunately, systematic uncertainties appear to limit the quality of the result, and further investigations are needed to further reduce these to fully exploit the statistical reach of the data samples. For what concerns the π^0 Dalitz branching ratio, in the intent of reducing the systematics effect on the measurement, we faced technical issues that force us to wait for a new processing effort of the whole data sample by the collaboration.

Besides the physical interest, the nature of the final state of the Dalitz decay channel, with a single photon, led to the development of a new technique for the fine calibration of the energy clusters of the electromagnetic calorimeter of the experiment. We took care of the development of a new data selection algorithm, somewhat complementary to the one developed for the branching ratio and the TFF measurement, and of the implementation in the NA62 data processing system.

In parallel to the study of the above decay, we investigated on general grounds the possible use of machine-learning approaches in high-energy physics, focusing on different areas from those which are already well studied (such as multivariate particle-identification algorithms). These investigations spanned two different topics. In the context of the possible improvements to be implemented before the Run2 data-taking period in July 2021, we took care of the design of a new L1 trigger algorithm to reduce the expected increase of the L0 trigger rates due to the higher beam intensity with respect to the Run1 period. In particular, we focused on the study of a Deep Learning algorithm exploiting only the information on the rings generated by particles crossing the Rich Cherenkov Counter (RICH), which to date

is only partially exploited to generate a trigger decision. In particular the interest for such an algorithm meets the interest of all the NA62 analyses with a final state including more than one charged particle, such as the Dalitz decay, whose data samples are limited to date because the need of downscaling factors in the current trigger selections. Starting from the implementation of simple Deep Learning models, we explored more complex techniques, eventually obtaining good results with a hybrid approach, mixing the machine-learning pattern recognition of rings with analytical fit techniques. We stress that few examples of such approaches appear in the literature (for example [25]), as usually information from multiple detector are used. As a consequence, we developed the technical framework from scratch, starting from the dataset design, the preprocessing steps where needed and the model tuning. The final goal of such an approach is to allow the hardware implementation of a simple and fast machine-learning trigger approach which might be successfully implemented online. A second novel line of investigation of machine-learning techniques was the preliminary feasibility study of the ambitious program of performing a complete event classification at the analysis level with limited human intervention, which might be seen as an alternative cross-check to standard approaches, if not as an ultimate technique to best exploit the available data. Some of the issues which need to be overcome for such a long-range goal were studied on the Dalitz decay analysis.

In the first Chapter we give a general overview to the NA62 experiment, its main physics goal and a description of some of the detectors and the reasons for design choices.

In the second chapter we describe the measurement of the π_D^0 Dalitz decay transition form factor measurement, briefly introducing the theory describing the process, describing the event selection, and the final result with the report on all the studied uncertainties. In the final part of the chapter we report about the prospects for the branching ratio measurement.

In the third Chapter we report about the new calibration technique for the energy clusters of the electromagnetic calorimeter based on the π^0 Dalitz decays.

In the fourth chapter, we describe the study on the possible new L1 software trigger algorithm for the RICH detector, discussing all the tested approaches and their results.

In the final chapter, we describe the first result from the studies of the implementation of a fully automatized pipeline to perform the selection of π_D^0 decay events and a first "toy" result.

*La frase più pericolosa in assoluto è:
Abbiamo sempre fatto così.
(Grace Murray Hopper)*

Chapter 1

The NA62 Experiment

1.1 Physics Motivations

The NA62 experiment aims to detect ~ 100 events of the ultra rare kaon decay $K^+ \rightarrow \pi^+ \nu \bar{\nu}$ to measure its BR with a 10% precision, to perform a stringent test of the Standard Model or expose evidence of New Physics. $K^+ \rightarrow \pi \nu \bar{\nu}$ is a Flavour Changing Neutral Current process and as such, in the Standard Model, forbidden at tree level and occurring through box or penguin diagrams (Figure 1.1). Due to the GIM mechanism and the transition from the t to d quark, it is an ultra rare decay, that probes higher mass scales for possible virtual particles contributing in loops than other rare meson decays, and is affected by unusually small theoretical uncertainties. The main source of these are simply related to the CKM matrix element V_{cb} , to the NLO and NNLO QCD corrections to the t and c quark contribution respectively, and NLO electroweak corrections. The hadronic matrix element, which is usually the model-dependent limiting factor in the computation of hadron decay properties, is extracted from the semileptonic decay $K^+ \rightarrow \pi^0 e^+ \nu_e$, which is well measured in [45]. The electroweak amplitude is dominated by top-quarks loops, resulting in a theoretically clean dependence on the product of CKM matrix elements $V_{ts}^* V_{td}$.

After the Run1 data-taking period which took place in the years 2016-2018, the NA62 experiment collected the largest world sample of $K^+ \rightarrow \pi^+ \nu \bar{\nu}$ decays, corresponding to 20 candidates, which already improved the BR measurement for this very rare decay [12]. The most updated theoretical predictions and experimental measurements are shown in Table 1.1.

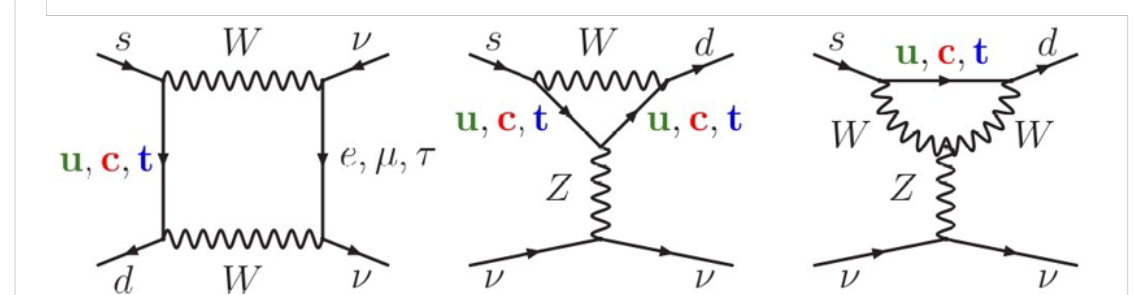


Figure 1.1: Feynman diagrams for the $K^+ \rightarrow \pi^+ \nu \bar{\nu}$ process. The Box diagram on the left and Penguin on the center-right.

Theory [9]	Experiment [12]
$(8.4 \pm 1.0) \times 10^{-11}$	$(10.6^{+4.0}_{-3.4} _{stat} \pm 0.9_{syst}) \times 10^{-11}$

Table 1.1: Most recent values of the $K^+ \rightarrow \pi^+ \nu \bar{\nu}$ branching ratio.

1.2 The NA62 Detector

According to the SM prediction of the $BR(K^+ \rightarrow \pi^+ \nu \bar{\nu})$ ($\sim O(10^{-10})$), a way to reach the main goal of the experiment is to have a detector acceptance of 10%, a background rejection factor of 10^{12} for other kaon decay modes, and an integrated flux of $O(10^{13})$ kaon decays. These requirements are reached with a experimental setup based on:

- the use of a high-energy hadron beam;
- the exploitation of a decay in flight technique.

Both represent an innovation with respect to the experiments which have searched for the $K^+ \rightarrow \pi^+ \nu \bar{\nu}$ decay so far, exploiting protons with low momentum beams and kaon decays at rest [2]. The use of a high energy proton beam (400 GeV/c) from the CERN SPS accelerator enabled the production of the required number of kaon decays in a few years of data-taking [13]. This occurs because the kaon production cross section increases at increasing values of the proton energy. The proton beam produces a 75 GeV/c secondary beam, after impinging on a beryllium target. The disadvantage of using a high energetic beam is represented by the impossibility of separating the pion and protons from the kaons. The particle rate of 750 MHz, composing the secondary hadron beam, contains a 6% (45 MHz) kaon composition. Among the produced kaons, only $\sim 10\%$ (Table 2 in [13]) decay in the 65 m-long fiducial decay volume (Figure 1.2), for a total of $\sim 5\%$ MHz kaon decays [13].

The decay in flight technique profits from the Lorentz boost of 75 GeV/c allowing to concentrate most of the detectors in the forward region. Moreover, the detection of some backgrounds is made simpler, especially for those containing photons in the final state. The large number of kaon decays collected by the experiment allows many other studies to be performed and measurements to be improved. While this work effectively focuses on a different decay from the one described here, we briefly describe the measurement of $K^+ \rightarrow \pi^+ \nu \bar{\nu}$ to illustrate the design of the experiment and the experimental apparatus.

The main difficulty lies in the limited signature of the signal, which is not kinematically closed and contains only two detectable particles: the incoming K^+ and the outgoing π^+ .

The rejection of the most abundant kaon decay modes is performed through the quantity

$$m_{miss}^2 = (P_K - P_\pi)^2 \quad (1.1)$$

(P_K is the 4-momentum of the parent particle, assumed to be a kaon, and P_π is the 4-momentum of the decay particle, assumed to be a charged pion): two different

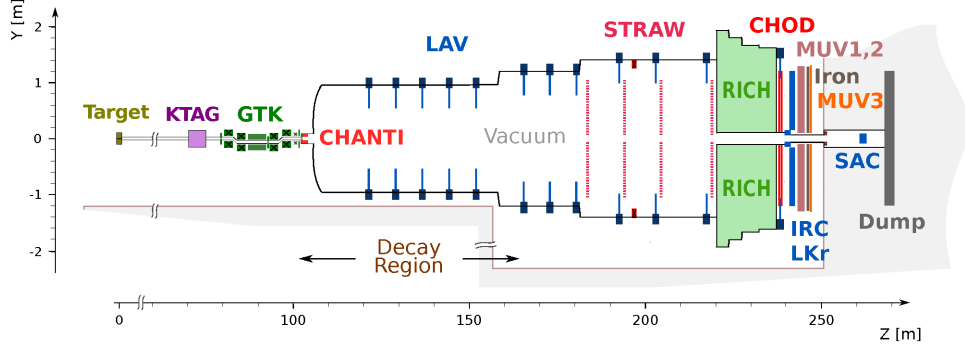


Figure 1.2: Sketch of the NA62 detector in the YZ plane [13]. The Z coordinate follows the beam propagation and the origin is at the position of the beryllium target. The Y axis is vertical and points upwards. The X axis is oriented to form a right-handed coordinate system. Note the difference in scale along the two axis. The decay region, starting from 102.4 m, is indicated. Some of the sub-detectors are shown.

Signal Regions (SR1 and SR2) are defined by looking at the distribution of (1.1) for background and signal events, plotted in Figure 1.3. The same plot is shown

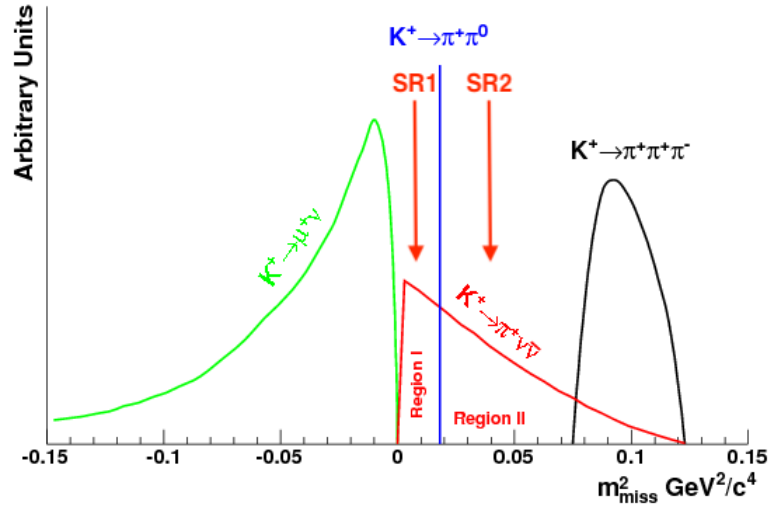


Figure 1.3: Distribution of m_{miss}^2 for some background kaon decay modes ($K_{\mu 2}$, $K_{2\pi}$, $K_{3\pi}$) and signal (red line). The two signal regions for $K^+ \rightarrow \pi^+ \nu \bar{\nu}$ are indicated. Negative values of the variable are present because we are imposing the mass of π^+ to the outgoing track to build P_π ; if the mass assignment is wrong, the computed missing mass can be negative.

in Figure 1.4 but with the finite width of the $K_{2\pi}$ peak due to the resolution. The choice of (1.1) as the discriminating variable determined part of the structure of the detector, since simulations show that to have a kinematic rejection factor of $O(10^4 - 10^5)$, a control on the $K_{2\pi}$ peak at a level of 0.001 (GeV/c^2) is needed and, to accomplish this result, a precise tracking system was inserted in the detector, measuring the momentum and angle of the kaon with a precision of $\sim 0.2\%$ and 0.016 mrad respectively, and 1% and 0.06 mrad for the downstream track [13].

The missing rejection factor of $\sim O(10^7) - O(10^8)$, to reach the desired level of

at a temperature of $\approx -15^\circ\text{C}$ [13]. The average beam particle rate on the GTK is 750 MHz.

The STRAW spectrometer measures the trajectories of the particles downstream of the decay region, starting from ~ 20 m at the end of it and covering a length of 35 m in the longitudinal direction. The guidelines used in its construction were to minimize multiple scattering and to give a uniform space resolution over the full active area. The former is accomplished using light material for the straws (36 μm thick polyethylene terephthalate, coated with 50 nm of copper and 20 nm of gold on the inside and gold-plated tungsten anode wires of 30 μm in diameter) and placing them into vacuum. The total amount of material corresponds to 1.8% X_0 .

The straws are arranged into four chambers, two before and two after a dipole magnet (MNP33), with an integrated field of ≈ 0.9 Tm. Each chamber is composed by two modules and each module by two views (or planes) (Figures 1.6 (a)). The views in the first module measure X and Y coordinates, while those in second module measure the U and V orientations, angled to -45° and 45° respectively (Figure 1.6 (b)). The resulting circular shape has a diameter of 2.1 m. In the middle of each view a gap, 12 cm in diameter, is present, forming a free passage for the beam. Overlaying the four views an octagon shaped central hole with 6 cm apothem is formed. Since the beam comes out the GTK with a displacement towards positive X values and receives a kick in the middle of the MNP33 magnet, bringing it towards the detector longitudinal axis, the hole of each chamber is displaced with respect to the neighbouring ones (Figure 1.7). To reconstruct a track, hits in all but one modules are required. The track reconstruction efficiency is reduced if the track is close to the central region of a chamber. The overall momentum resolution is [13]:

$$\frac{\sigma(p)}{p} = 0.30\% \oplus 0.005\% * p \quad (1.2)$$

with p in GeV/c .

The MNP33 dipole magnet is the same of the previous NA48 experiment [16]. The magnetic field points in the negative Y direction and the integral of the other components is smaller by a factor of 10^{-3} . The momentum kick is 270 MeV/c . A sketch of the tracking system is shown in Figure 1.8.

1.2.2 Veto system

Among the common kaon decay modes, the ones containing a π^0 or a muon represent $\approx 89\%$ of the kaon decays as shown in Table 1.2. These decays are the main background sources to the signal when the two photons of the π^0 escape the detector geometric acceptance and the μ^+ is misidentified as a π^+ . Two veto systems are used to reject these decays.

The Photon Veto (PV) is composed by the NA48 Liquid Krypton calorimeter (LKr [23]), the Large Angle Veto (LAV) detector, the Intermediate Ring Calorimeter (IRC) and Small-Angle Calorimeter (SAC). The system is designed to suppress the kaon decays with a π^0 in the final states, the most abundant of which is $K_{2\pi}$. To give an overall 10^8 rejection factor, the π^+ momentum of a signal decay is required

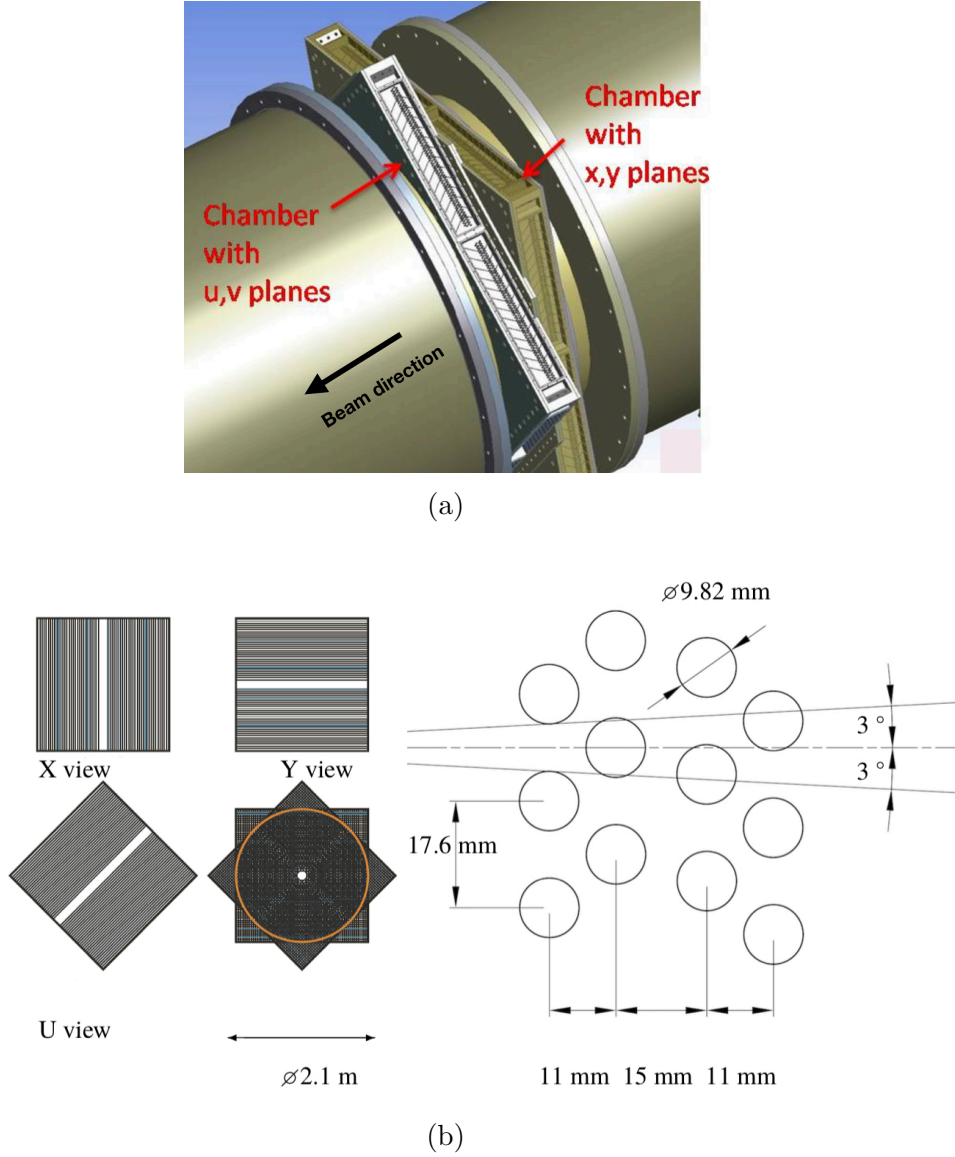


Figure 1.6: Details of the STRAW spectrometer [13]. In (a) a single chamber with two modules is shown. In (b), on the left, the arrangement of the straw tubes in the different views of a chamber is shown. On the right the overlap of the two double layers per view is sketched. This arrangement is required to have at least two straw crossing per view per track, to solve the left-right ambiguity.

to lie in the interval $15 - 35 \text{ GeV}/c$. With this requirement the π^0 of $K_{2\pi}$, with an energy at least of 40 GeV , cannot escape the angular coverage of the PV because of the anti-correlation between energies and emission angles [13]. The polar angular range with respect to the beam axis is subdivided as follows:

- from 8.5 mrad to 50 mrad covered by the LAV detector;
- from 1 mrad to 8.5 mrad covered by LKr;
- from 0 mrad to 1 mrad covered by the IRC and SAC;

These sub-detectors are highlighted in Figure 1.9 together with their angular coverage.

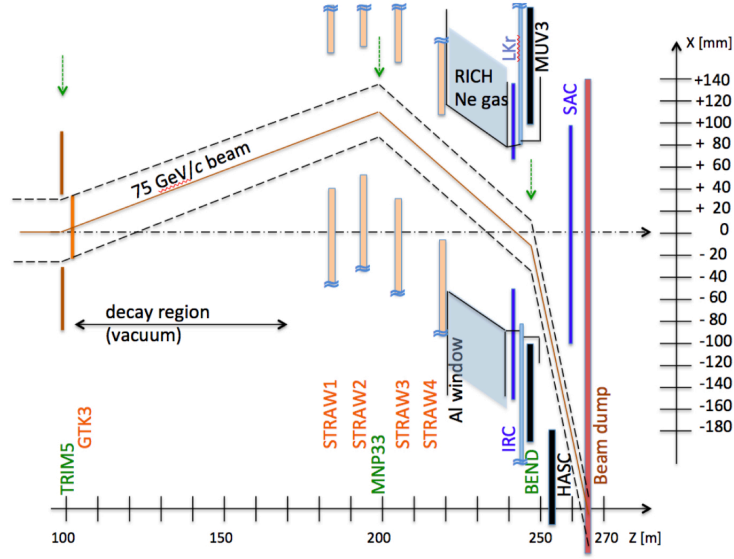


Figure 1.7: The displacement of the STRAW chamber centers in the X direction [13] is visible in this schematic drawing of the XZ plane of the detector. Note that the scale is different on the two axes.

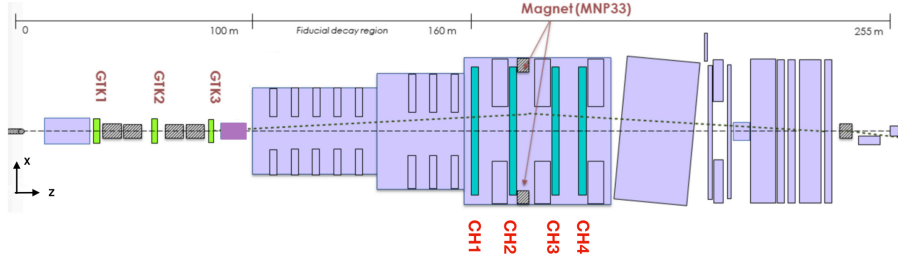


Figure 1.8: Sketch of the NA62 detector. The hadronic beam propagates towards the positive direction of the Z direction. The GTK and the STRAW spectrometers are highlighted in green with respect to the other sub-detectors (in purple). The GTK stations (GTK1, GTK2, GTK3) and STRAW chambers (CH1, CH2, CH3, CH4), with the MNP33 magnet are shown. The angled dashed line represents the trajectory of a charged particle receiving the kick of the downstream spectrometer magnet.

The LAV detector is composed by 12 ring-shaped stations (Figure 1.10 (a)), 11 installed in the vacuum tank and the last one in air, just 3 m upstream of the LKr front face (Figure 1.9). Each station is an arrangement of 4 – 5 layers of lead-glass blocks. Layers are separated by 1 cm along Z and are staggered in the XY plane (see Figure 1.10). In this way, particles crossing the detector intercept an average of three blocks, with a total effective material depth of $21X_0$. The overall inefficiency of the LAV detector must be less than 10^{-4} to reach the required 10^8 rejection power.

The LKr electromagnetic calorimeter is the one used in the NA48 experiment [23]. The only change has been applied to its read out system to satisfy the demanding rate requirements of NA62. It is a quasi-homogeneous ionization chamber, filled

Decay	BR
$\mu^+\nu$	$\sim 63\%$
$\pi^0\mu^+\nu$	$\sim 3\%$
$\pi^+\pi^0$	$\sim 21\%$
$\pi^+\pi^0\pi^0$	$\sim 2\%$

Table 1.2: Branching ratios of the main kaon decay channels with a π^0 or a μ^+ in the final state [45].

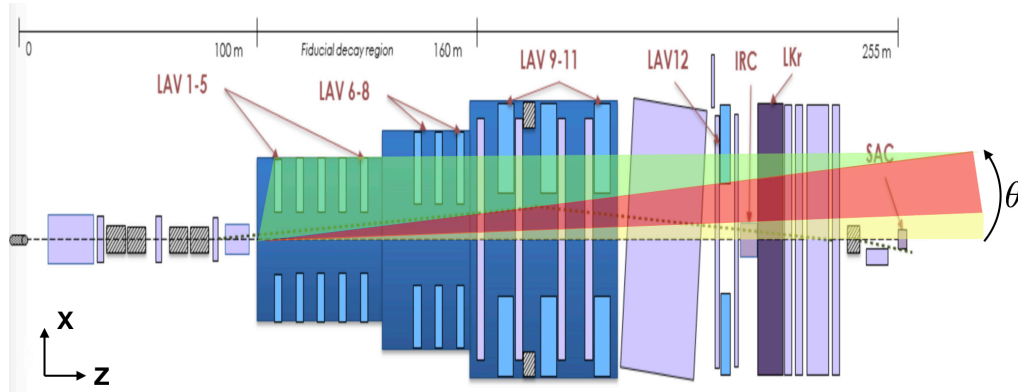


Figure 1.9: Sketch of the photon veto system (in blue). The beam propagates along the Z direction. The three intervals of angular coverage are shown from 0 mrad (the beam line) to 50 mrad (at the beginning of the decay region). Yellow area: IRC and SAC. Red area: electromagnetic calorimeter. Green area: LAVs.

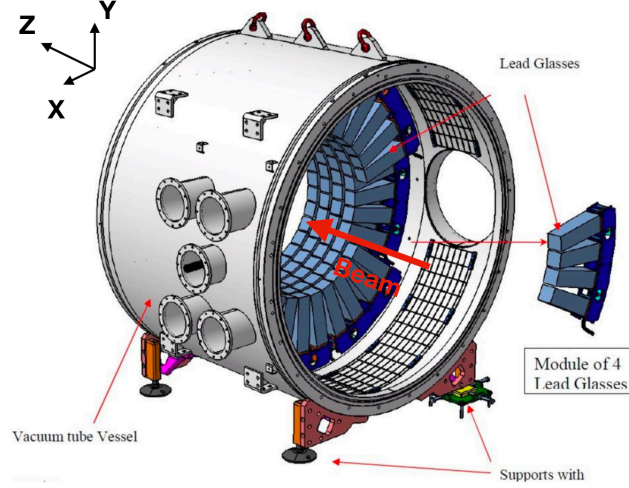


Figure 1.10: Drawing of one LAV station with 4 layers of lead-glass blocks [13].

by 9000 [13] litres of liquid krypton at 120 K, housed in a cryostat. Geometrically it has a octagonal shape and extends from the beam pipe, with a inner radius of ≈ 8 cm, to an outer one of 128 cm. The total length along the beam direction is 127 cm, corresponding to $27X_0$. The active area is divided in 13248 longitudinal cells with a cross section of 2×2 cm². A drawing of the LKr is shown in Figure 1.11. A cluster is formed by a cell with a deposited energy > 250 MeV and larger than the sum of those in the 8 neighboring cells. All the energies measured from cells within 11 cm from the cluster center are summed to build the total energy of the cluster. The energy resolution of the LKr is [13]:

$$\frac{\sigma(E)}{E} = \frac{4.8\%}{\sqrt{E}} \oplus \frac{11\%}{E} \oplus 0.9\% \quad (1.3)$$

with E in GeV. The cluster position is an energy-weighted average of the X and Y coordinates of all the cells participating in the cluster.

The LKr is also exploited for particle identification together with the STRAW spectrometer information on the track momentum, by computing the "energy over momentum ratio" (E/p).

Finally the two sub-detectors IRC and SAC compose the small angle veto (SAV) system, deputed to detect photons not hitting the previous PV detectors, due to their proximity to the beam pipe. The IRC and SAC are are shashlyk calorimeters.

The muon veto (MUV) system is used to discriminate between π^+ and μ^+ and is composed by three sub-detectors: two hadron calorimeters, MUV1 and MUV2, and, after a 80 cm thick iron wall, a muon veto detector (MUV3) composed by 140 plastic scintillator tiles. The whole MUV veto system is sketched in Figure 1.12.

1.2.3 Particle identification system (PID)

The PID system is composed by two Cherenkov detectors: the CEDAR (coupled with the Kaon TAGger or KTAG), and the Ring Image Cherenkov counter (RICH). The position of each detector is shown in Figure 1.13.

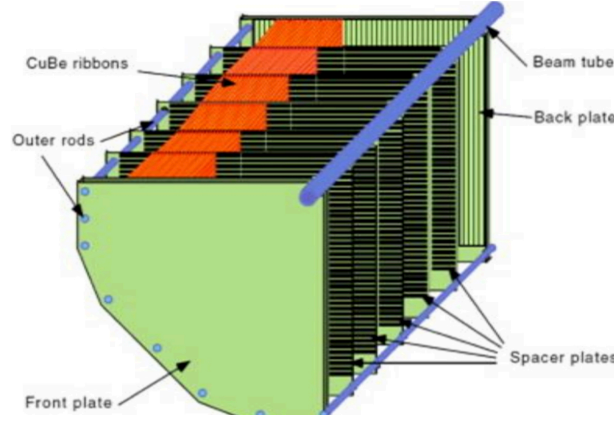


Figure 1.11: Representation of a quarter of the LKr and of its inner side with the cell arrangement [13].

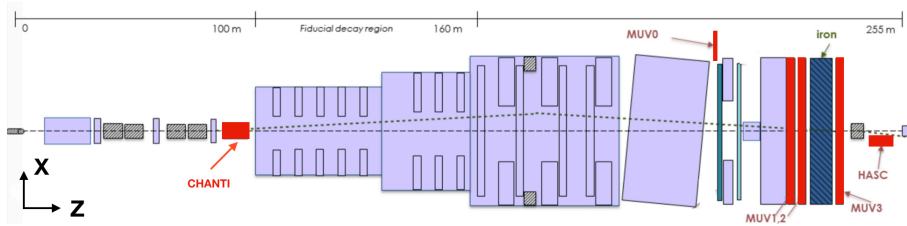


Figure 1.12: Sketch of the MUV system and other veto sub-detectors (in red). The beam is propagating along the Z direction. CHANTI, MUV0, HASC are described in detail in [13].

The CEDAR, is used to identify kaons among the beam particles. The CEDAR is a differential Cherenkov counter filled with nitrogen (N_2) at 1.75 bar at room pressure [13]. This represent a total of $3.5 \times 10^{-2} X_0$ in the path of the beam. The pressure of the radiator is chosen so that only light coming from the desired particle passes through an annular diaphragm. The original CEDAR gas volume and optics are suitable for use in NA62, but the original photodetectors and readout electronics are not capable of sustaining the nominal 45 MHz kaon rate of the NA62 beam line and of reaching the desired 100 ps time resolution, so that the new KTAG was developed. The light reflected back by the CEDAR mirrors passes through eight quartz windows and is focused onto eight spherical mirrors. These mirrors reflect the light outwards, in the radial direction, and this is collected by a light guide and driven into eight light boxes (or sectors) containing PMTs (Figure 1.14). Requiring a time coin-

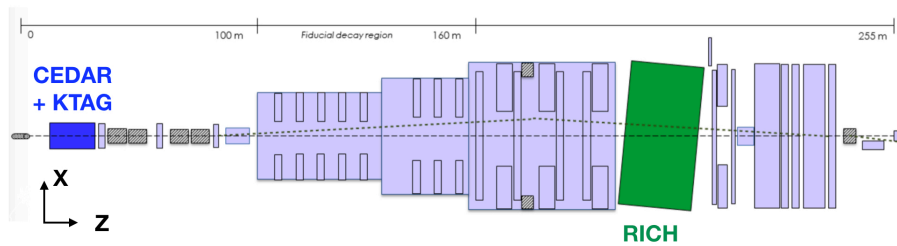


Figure 1.13: The PID system (highlighted in blue and green) with respect to the other sub-detectors which are in purple.

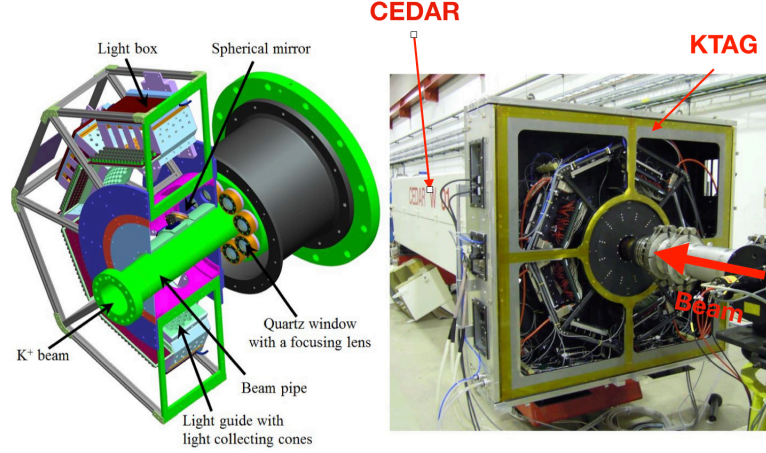


Figure 1.14: On the left 3D representation of the KTAG detector with its main parts highlighted. On the right a picture of the system CEDAR + KTAG as is installed in the NA62 experimental hall. The direction of the beam is indicated [13].

cidence of signals in at least 5 sectors, gives an overall kaon tag efficiency of 98% [13].

The Ring Image CHerenkov counter (RICH) is designed to separate muons and pions in the $15 - 35 \text{ GeV}/c$ momentum interval, providing a factor 100 rejection for kaon decays with a muon in the final state.

It is composed by a 17.5 m long cylindrical vessel, filled with neon gas used as radiator. The Cherenkov light, reflected by a mosaic of 20 spherical mirrors (Figure 1.15 (a)), placed at the end cap of the vessel, is collected by two photomultiplier (PMTs) arrays placed at the upstream of the RICH, on the left and right of the beam pipe crossing the detector. 2000 PMTs are used to reach the required angular resolution. The RICH structure is shown in Figure 1.15 (b). To get a good discrimination efficiency in the lower part of the momentum range, the minimum pion momentum to generate Cherenkov radiation must be $12.5 \text{ GeV}/c$.

Two plastic scintillator Charged particle HODoscopes (CHOD) complete the NA62 detector. One originates from of the previous NA48 experiment while the other was built specifically for NA62. For this reason they are generally called "Old CHOD", or simply CHOD, and "New CHOD". They are employed especially to generate trigger signals and, in association with the spectrometer and the KTAG, to give precise time measurements of the track and of the matching with the GTK track. They are exposed to a particle flux of 13 MHz [13], mostly kaon decay products and a small fraction of muon halo and pion decays. The two detectors are separated longitudinally by $\approx 70 \text{ cm}$.

The NewCHOD is placed just after the RICH end cap and is composed by 152 plastic scintillator tiles of 30 mm thickness ($0.13X_0$) covering a disk surface with inner and outer radius of 140 mm and 1070 mm respectively (Figure 1.16 (a)). The different-sized tile structure is used to equalize the hit rate distribution, shown in Figure 1.16 (b). The time resolution is $O(1 \text{ ns})$.

The CHOD detector (Figure 1.17) is composed by two planes (H and V) of 64 vertical and 64 horizontal plastic scintillator slabs of 20 mm ($0.10X_0$) thickness. Each slab is read by PMTs through light guides. The 64 slabs are assembled into

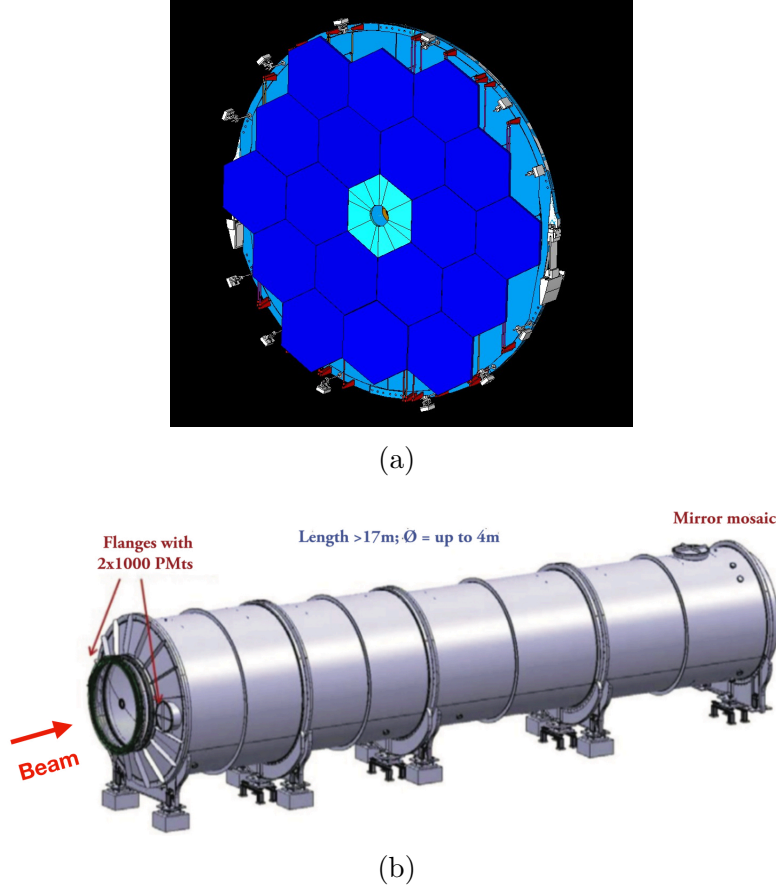


Figure 1.15: The NA62 RICH detector. (a) The mosaic of mirrors with the structure used to sustain its weight. The total thickness is 20% of a radiation length. (b) The whole structure of the sub-detector. The direction of the beam is indicated, as well as the two PMTs spots, and the back plane, housing the spherical mosaic of mirror.

4 quadrants with 16 slabs in each one. Two independent time measurements are provided for each charged particle crossing a vertical and a horizontal slab, reducing the possible tails given by out-of-time events, reaching a time resolution of 0.3 ns. The incident flux reaches 13 MHz and reaches almost 35 MHz for each plane, due to the activity of the upstream and downstream detectors, such as the LAV12 induced activity or the particle back-scattering from LKr and pion decays.

1.3 NA62 Trigger and Data Acquisition System (TDAQ)

The high intensity of the NA62 beam requires an high performance trigger and data acquisition system, with the main purposes of rejecting background events, minimizing the dead time and maximizing the data collection processes, while keeping a time resolution as high as possible. One of the features of the NA62 experiment is that the trigger and data acquisition system are unified. This is somewhat different to the most common case in which a restricted sub-set of informations is processed by a dedicated trigger system. Improvements introduced by this approach are the elimination of a specific trigger data path, consequently reducing the needed hard-

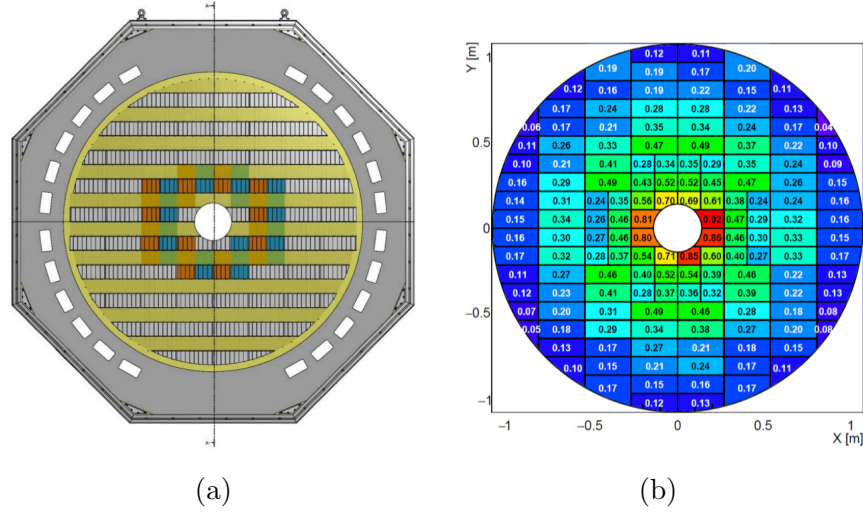


Figure 1.16: (a) The NewCHOD with its tile structure. The direction of the beam in this case is perpendicular to the paper. (b) Incident flux of particles in the NewCHOD. The number in each tile is the expected rate in MHz at nominal beam intensity [13]. Due to the presence of the beam, the central tiles receive a higher rate of incident particles [13].

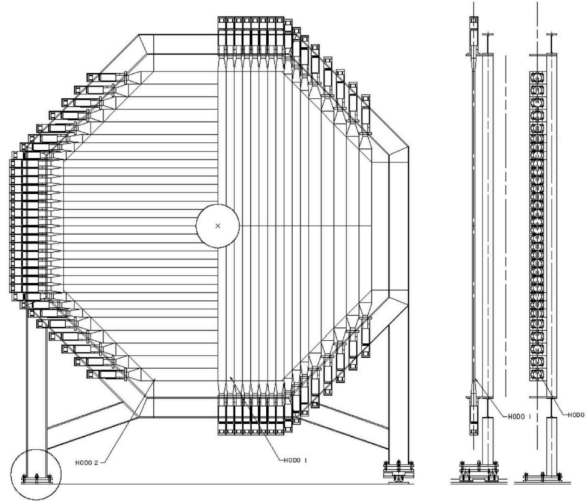


Figure 1.17: The CHOD cross section with the two arrangement of slabs in the two planes H, V. At the end of each slab a PMT is present. The hole at the center of the detector houses the beam pipe passage. On the right a side view of the detector [13].

ware, and the exact offline reproducibility of the trigger algorithms.

The timing to the whole system is given by the Timing, Trigger and Control (TTC) system developed for the LHC experiments. The TTC is driven by a common free-running synchronous 40.079 MHz [13] clock which is the unique reference for time measurements.

The data-taking unit in NA62 is called "burst" and corresponds to the SPS beam spill, which is a few seconds long. All the elements of the NA62 TDAQ system are synchronously reset by a Start Of Burst (SOB) and stopped at each End Of Burst (EOB) signal.

The trigger system is composed by three levels: the first (L0) is hardware based, the second (L1) and the third (L2) are software based and are implemented on dedicated PCs. The whole trigger chain permits to decrease the event rate from 10 MHz to 10 kHz.

Each detector participating to the L0 trigger produces primitives, encoded in a 32 bit word containing sub-25 ns time information and an identifier which indicates which conditions are satisfied at that time. If two different primitives are produced with a time difference smaller than the sub-detector resolution, they are merged into one. The detectors involved in the primitive generation are CHOD, NewCHOD, RICH, MUV3, LKr.

Primitives are sent from each system to the L0 Trigger Processor (L0TP), which stores the primitives in a RAM. Primitives are stored in the RAM according to the primitive time, with each segment of the RAM corresponding to a window of time (typically 6.25 ns). Only one primitive from each detector can be stored in any given segment of the RAM. The L0TP starts by finding primitives from a "reference detector", which is usually the RICH. The L0TP then checks for coincident primitives from the other systems, looking in 3 RAM segments (or slots): the same segment as the primitive from the reference detector; the previous segment; and the next segment. The other primitives are only considered to be coincident if they are within a defined time (typically ± 6.25 ns) of the primitive of the reference detector, evaluated at 100 ps precision. In case some predefined logic conditions (called Masks) are satisfied, a L0 trigger signal is produced and sent back to the detectors which transfer their data to the PC Farm. When more than one mask is satisfied, the *or* logic condition is applied. The RICH time is the reference one when a "physics" mask is satisfied. A certain fraction of events are triggered by a minimum bias condition (*Control* trigger), generated by a signal in the CHOD without any further request at L1. These events are used for trigger efficiency measurements. A downscaling factor (*D*) is used to reduce the number of events passed to the L1 trigger for some masks. L0 masks are associated to L1 trigger conditions of the remaining detectors not used at L0: LAV, STRAW, KTAG. As for the level 0, a certain number of events, called *Autopass*, are collected without looking at the response of the L1 trigger, to be used for a subsequent measurement of the L1 efficiency.

1.4 NA62 reconstructed data structure and analysis framework

Through the execution of the NA62Reconstruction software package, raw detector information are combined in time and space, using specific algorithms for each sub-detector, to build refined quantities stored in so-called *Candidates*, the basic structure handled by the final analysis. As an example, if raw information are isolated energy releases in the cells of the electromagnetic calorimeter, the *Candidates* are the possible clusters that could be assembled starting from them. In this way each *Candidate* becomes the best identification of the energy deposit of the interacting particle. Similar considerations on the spectrometer lead to a *Spectrometer Candidate*, which in this case is synonymous of a fitted track in different chambers. A set of standard tools are available to the user in the NA62Analysis package, which contains also all the software functions to access the different *Candidates*. In

practice this software permits to convert a physical selection into a series of instructions, which can be collected in a unified program called *analyzer*. Each *analyzer* can work independently or in a chain. The latter configuration is useful each time the performed analysis can be subdivided into specific parts, permitting to reuse some information also for other purposes. An example is an *analyzer* that identifies electrons using a set of sub-detectors. Among the most common algorithms are:

- Time alignment: performing part of the automatic time calibration procedure applied on data.
- Pre-Analyzers: typically applying energy and momentum fine corrections to quantities provided by the reconstruction.
- Association Tools: combining information from more than one sub-detector, taking known instrumental effects into account.
- Geometry Tools: applying acceptance cuts or vertexing algorithms.
- Filters: used to collect events which share similar characteristics.

All of these make the implementation of some standard commands (such as geometrical track propagation from a detector to another) easier and guarantee uniform selection criteria. Moreover filters are used for the creation of reduced dataset files more easily accessible by users for analysis. In this way the user can run only over the events in which the basic features of the signal are present. All the cuts applied by the tools are documented in [39]. The NA62Reconstruction and NA62Analysis packages are constantly updated and improved. In this work the standard tools are used in the applied selections but they will not be explained in depth, except where this could bring clarity.

The NA62 dataset is divided into period of acquisition to separate periods in which the experimental conditions remains stable. During 2017, the periods are indicated as A-B-C-D, while in 2018 the periods are A-B-C-D-E-F-G-H.

1.5 NA62 MC package

The NA62 MC package is based on GEANT4 [1] for the detector description, TURTLE [6] for the beam optics upstream the first detector (CEDAR), and a set of custom decay generators (mostly for kaons) that include form factors and radiative corrections.

In a MC simulation, the interactions of the generated particles with a specific detector are stored as raw data like a real data event. Raw data files are processed with the NA62Reconstruction package to produce the *Candidates* of the various sub-detectors. In this way there is a unique analysis software both for real and simulated data. The trigger decision is stored as a reconstructed quantity, later accessible by the user to asses the effect of the trigger on its selection. It is emulated with specific algorithms specific for each detector. The processing of MC and real data differs only for the filtering step, which is absent in the former. Since it is not feasible to produce MC samples reproducing each Run configuration, two data taking conditions are used for the detector simulation in each year of data taking: for the 2017 period, Run 8094 is used, while Run 9001 is used for the 2018. The generated quantities can be accessed while studying MC events, for example:

- the position and the four-momentum of a particle;
- the "parent" particle

MC files are divided into different samples, according to the simulated process (e.g. $K_{3\pi}$ events are all collected in the same files).

1.5.1 Overlaid MC samples

Overlaid MC samples are particular samples in which multiple MC raw events, coming from different single-decay-mode MC samples, are overlapped to the original MC event, with a random time offset within a certain configurable time window. This is meant to reproduce the pile up effect that is present in real data. This procedure improves the similarity of the MC and real data treatment at reconstruction level and has been shown to improve the data-MC agreement at analysis level. The event merging procedure is done by random sampling the intensity beam distribution, extracted from 2017 data, and computing the average number of pileup events. Three different particles can be overlapped to the original ones: a proton propagating in the detector, a charged pion and a charged kaon. The fraction of event, coming from the list is chosen according the measured ratio on data: 24%, 70%, 6% respectively. Both the pion and the kaon are free to decay or simply propagate in the detector. In the case of the kaon events, the dominant six decays are simulated, according to their BR : $K_{\mu 3}$, $K_{e 3}$, $K_{2\pi}$, $K_{3\pi}$, $K_{3\pi 0}$, $K_{\mu 2}$. In events containing π^0 both its 2γ and Dalitz decay channels are simulated.

It's clear that when overlaid MC samples are used for data-MC comparison, a different procedure for MC normalization is needed: this is described in detail in Appendix A.1.

Chapter 2

Measurement of the π^0 Transition Form Factor

2.1 The π^0 Dalitz decay

The π meson is the lightest strongly interacting particle in nature. It exists in three different state of charge: π^+ , π^0 and π^- . Predicted for the first time by H.Yukawa in 1935 [44] as mediator of the strong nuclear force, the charged pion was experimentally detected in 1947 by C.Powell [34] studying the composition of the cosmic rays through photographic emulsions, while the π^0 was detected later in the 1950 [5].

The π^0 decay is dominated by the electromagnetic process $\pi^0 \rightarrow \gamma\gamma$ which represents $(98.823 \pm 0.034)\%$ of the total π^0 decay width according to [45]. The remaining $\sim 1.2\%$ is dominated by the so called Dalitz decay mode, investigated for the first time by R.H Dalitz [15]. This proceeds, as the dominant decay mode, via a photon pair, but having one of the two photons produced off-shell and then converting into a e^+e^- pair as shown in Figure 2.1.

The black blob in Figure 2.1 represents the electromagnetic Transition Form Factor which encodes the whole information about the structure of the decaying particle and the interactions with other particles. In this case, it contains information on the electromagnetic structure of the π^0 . The interest to measure the TFF slope is related to the importance it has in the theoretical computations for important SM test processes like the anomalous magnetic moment of the muon, $(g - 2)_\mu$, in

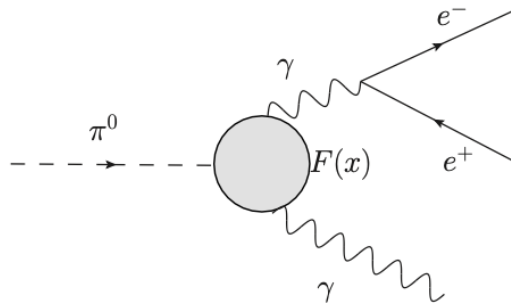


Figure 2.1: π^0 Dalitz decay diagram. The blob represents the transition form factor, defined as $F(x)$.

which it accounts for the second largest uncertainty contribution ($\sim 25\%$) through the hadronic light by light scattering, and the calculations for the rare decays, such as $BR(\pi^0 \rightarrow e^+e^-)$ [45]. The uncertainty on the Dalitz BR is the main source of uncertainty in the determination of the BR for other decays, where the Dalitz process is used as normalization channel as, for instance, the $BR(K^+ \rightarrow \pi^+\pi^0 e^+e^-)$ process [3]). More Theoretical details can be found in [41].

As discussed in [33], in electromagnetic interactions involving hadrons, the decay width or the cross section of the analyzed process can be factored in its point-like (pl) part times a form factor term $|F|^2$:

$$\frac{d\Gamma}{dq^2} = \frac{d\Gamma}{dq^2} \Big|_{\text{pl}} |F|^2 \quad (2.1)$$

The general theory on the form factor includes the case in which both photons are emitted off-shell with four-momenta q_1 , q_2 , described by the two-variable form factor

$$F = F(q_1^2, q_2^2) \quad (2.2)$$

The normalization for (2.2) can be taken in the $q_i^2 \rightarrow 0$ limit, which corresponds to the $\pi^0 \rightarrow \gamma\gamma$ decay [10]:

$$\begin{aligned} \Gamma(\pi^0 \rightarrow \gamma\gamma) &= \left(\frac{M_{\pi^0}}{4\pi} \right)^3 \left(\frac{\alpha}{F_\pi} \right)^2 = \frac{1}{64\pi} M_{\pi^0}^3 |F(0,0)|^2 \\ \Rightarrow F(0,0) &= \frac{\alpha}{\pi F_\pi} \end{aligned} \quad (2.3)$$

where $F_\pi = 92.2$ MeV is the pion decay constant. The π^0 Dalitz decay is a special case of (2.2) with one of the $q_i^2 = 0$ and the other, simply called q^2 , expressed as a function of the detectable particles four-momenta p_{e^+} , p_{e^-} :

$$q^2 = (p_{e^+} + p_{e^-})^2 \quad (2.4)$$

Useful Lorentz-invariant variables to describe the process are:

$$x = \frac{q^2}{M_{\pi^0}^2}, \quad y = \frac{2 \cdot p_{\pi^0} \cdot (p_{e^+} - p_{e^-})}{M_{\pi^0}^2(1-x)} \quad (2.5)$$

where p_{π^0} is the four momentum of the neutral pion and M_{π^0} its mass. The range of these variables is constrained by the minimum and maximum value of the q^2 quantity, the squared four-momentum of the off-shell photon:

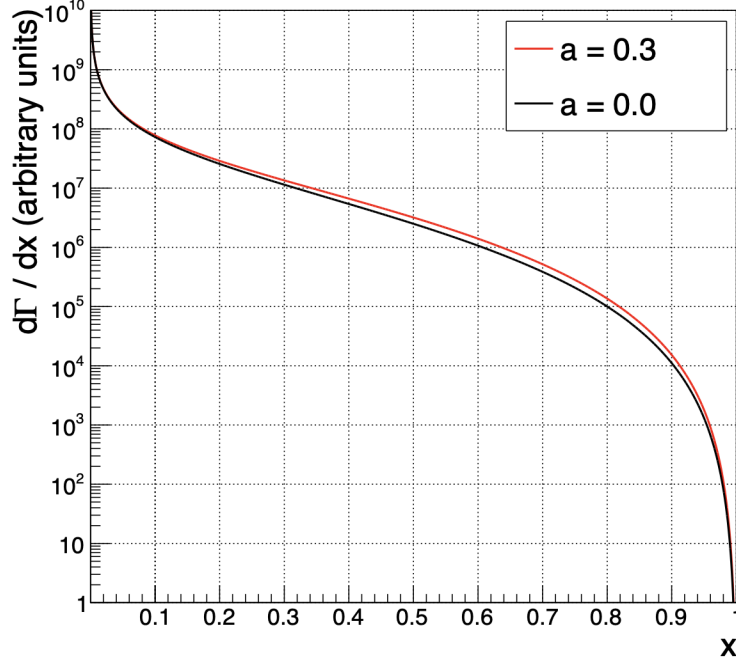
$$r^2 \leq x \leq 1, \quad -\beta \leq y \leq \beta \quad (2.6)$$

having defined

$$r = \frac{2m_e}{M_{\pi^0}}, \quad \beta = \sqrt{1 - \frac{r^2}{x}} \quad (2.7)$$

Thus using (2.5), the form factor in this process can be expressed either as a function of q^2 or x . At the leading order (LO), using relation (2.1) and $F = F(q^2) = F(x)$, the normalized differential decay width [10] is

$$\frac{1}{\Gamma(\pi^0 \rightarrow \gamma\gamma)} \frac{d^2\Gamma_D^{LO}}{dx dy} = \frac{\alpha}{4\pi} \frac{(1-x)^2}{x} \left(1 + y^2 + \frac{r^2}{x} \right) |F(x)|^2 \quad (2.8)$$


 Figure 2.2: TFF slope effect on the x distribution [36].

and integrating over the y variable

$$\frac{1}{\Gamma(\pi^0 \rightarrow \gamma\gamma)} \frac{d^2\Gamma_D^{LO}}{dx} = \frac{\alpha}{4\pi} \frac{8\beta}{3} \frac{(1-x)^3}{x} \left(1 + \frac{r^2}{2x}\right) |F(x)|^2 \quad (2.9)$$

One of the advantages in the study of π_D^0 decays is that, in the allowed kinematic region, $F(x)$ is expected to vary slowly and a linear approximation can be used:

$$F(x) = 1 + ax \quad (2.10)$$

Predicted values of a are computed in [22] using a *Vector Mesons Dominance* (VMD) approach according to which $F(x)$ is dominated by the contributions due to two resonances ρ and ω , resulting in a value of

$$a_{\text{VMD}} \sim 0.03 \quad (2.11)$$

The most significant effect of the TFF slope value is at large value of x distribution, as can be seen in Figure 2.2. In (2.9) radiative corrections must be included to obtain the Next-to-Leading-Order (NLO) decay width. These are parametrized by a $\delta(x, y)$ function defined as:

$$\frac{d^2\Gamma}{dxdy} = \frac{d^2\Gamma^{LO}}{dxdy} + \frac{d^2\Gamma^{NLO}}{dxdy}, \quad \delta(x, y) = \frac{d^2\Gamma^{NLO}}{dxdy} \bigg/ \frac{d^2\Gamma^{LO}}{dxdy} \quad (2.12)$$

Three different processes gives contributions to δ :

$$\delta = \delta_{\text{virt}} + \delta_{\text{brem}} + \delta_{1\gamma\text{IR}} \quad (2.13)$$

shown respectively in Figure 2.3 (a),(b), 2.3 (c),(d) and 2.3 (e),(f),(g). The δ_{virt} and δ_{brem} contributions were first studied by Lautrup and Smith [35] in the soft-photon approximation and extended by Mikaelian and Smith [38] by hard-photon

corrections to the whole range of the Bremsstrahlung photon energy. The most recent work on this topic is by T. Husek et al. in [28], who introduced the one-loop one-photon irreducible contribution ($\delta_{1\gamma\text{IR}}$), considered negligible by previous authors, showing that this contribution accounts for a variation of the x distribution, especially at large values, with discrepancies of 10% already at $x = 0.5$.

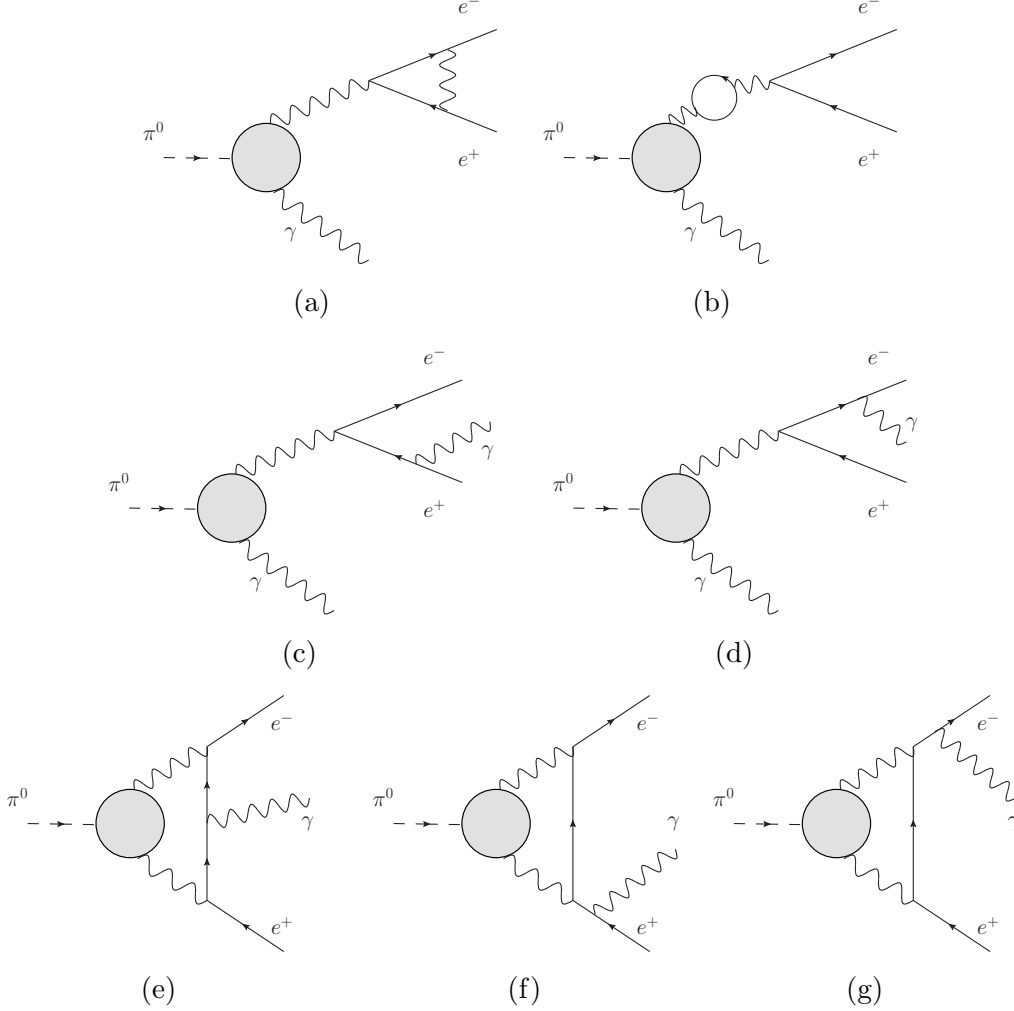


Figure 2.3: Feynman diagrams of radiative corrections to the π^0 decay width [28]. Virtual correction δ_{virt} (a-b), Bremsstrahlung δ_{brem} (c,d) and one-loop one-photon irreducible contribution $\delta_{1\gamma\text{IR}}$ (e-f-g).

The δ_{virt} contribution leads to infrared divergences, stemming from the QED vertex correction shown in Figure 2.3 (a). Such divergences are canceled by the δ_{brem} contribution, that is a 4-body decay, and for this reason described by five independent kinematic variables instead of two as for a three-body decay. The general approach is to integrate over the additional kinematic variables to get the value of $\delta_{brem}(x, y)$. One of the variables used to describe the Bremsstrahlung is the invariant mass of the di-photon pair defined as

$$x_\gamma = \frac{(p_{\gamma 1} + p_{\gamma 2})^2}{M_{\pi^0}^2} \quad (2.14)$$

which is limited in the kinematic range

$$\frac{\lambda^2}{M_{\pi^0}^2} \leq x_\gamma \leq 1 - x - \sqrt{4x - y^2\beta^{-2}(1-x)^2} \quad (2.15)$$

The λ appearing in (2.15) is the infrared cutoff used to regularise the divergent part of the contribution. Integration over the whole range of x_γ includes also events in which δ_{brem} is responsible for the emission of hard radiative photons, which are in principle detectable by the experiment. Thus, any analysis relying on cuts on the photon quantities, such as the minimum energy or distance of the clusters in the electromagnetic calorimeter, requires a proper simulation taking into account this kind of effect, in order to avoid differences between data and MC in the evaluation of the detector acceptance.

2.1.1 Dalitz MC generator

The Dalitz MC generator implemented in this analysis makes use of the model described in [28]. We describe here the fundamental conclusions of such a work. The integration of the matrix element

$$\delta_{brem}(x, y) = \int |\mathcal{M}_{brem}|^2 dx_\gamma \quad (2.16)$$

can be split into

$$\delta_{brem} = \int_0^{\min(x_\gamma^{cut}, x_\gamma^{max})} \delta_{brem}(x, y, x_\gamma) dx_\gamma + \int_{\min(x_\gamma^{cut}, x_\gamma^{max})}^{x_\gamma^{max}} \delta_{brem}(x, y, x_\gamma) dx_\gamma \quad (2.17)$$

Thus, the ratio

$$R(x_\gamma^{cut}) = \frac{\int_{x_{min}}^{x_{max}} dx \int_{y_{min}}^{y_{max}} dy \int_0^{\min(x_\gamma^{cut}, x_\gamma^{max})} [\Gamma^{LO}(x, y) + \Gamma^{NLO}(x, y, x_\gamma dx_\gamma)] dx_\gamma}{\int_{x_{min}}^{x_{max}} dx \int_{y_{min}}^{y_{max}} dy \int_{\min(x_\gamma^{cut}, x_\gamma^{max})}^{x_\gamma^{max}} [\Gamma^{LO}(x, y) \delta_{brem}(x, y, x_\gamma) dx_\gamma]} \quad (2.18)$$

can be used in the form

$$P(4 - \text{body}) = \frac{1}{R + 1} \quad (2.19)$$

to compute the probability that the process occurs as a four-body or a three-body decay, considering that equation (2.18) represents the number of soft radiative photons with respect to harder photons, when the two photons have an invariant mass greater than the cut-off value x_γ^{cut} . In the present simulation, a value of $x_\gamma^{cut} = 0.01$ is used, corresponding to a value of $R = 17.397$, and finally $P = 5\%$. The value of R for x_{cut}^γ was computed by [28] and the authors of [36], who performed this analysis with a similar scheme, considered it good to reproduce the process of interest.

The simulation of a 4-body Dalitz decay with an additional (low-energy) detectable photon ($e^+e^-\gamma(\gamma)$) proceeds as follows:

- Generation of all the four-momenta of the particles in the π^0 reference system with a constant decay amplitude
- Selection of only those events with $x > r^2$ and $x_\gamma > x_\gamma^{cut}$

- Computation of the invariant quantities

$$x = \frac{(p_{e^+} + p_{e^-})^2}{M_{\pi^0}^2}, a_i = \frac{p_{e^+} \cdot p_{\gamma_i}}{M_{\pi^0}^2} \quad (2.20)$$

$$x_\gamma = \frac{(p_{\gamma_1} + p_{\gamma_2})^2}{M_{\pi^0}^2}, b_i = \frac{p_{e^-} \cdot p_{\gamma_i}}{M_{\pi^0}^2} \quad (2.21)$$

where $i = 1, 2$ are the indices of the two photons

- Computation of the matrix element for the Bremsstrahlung process that is free of divergences for $x > x_\gamma^{cut}$
- Acceptance-rejection of the generated decay according to the matrix element distribution
- Boost to the laboratory frame

The simulation of a 3-body Dalitz decay (with possibly additional undetected low-energy photon) instead proceeds by generating events distributed according to Γ^{NLO} where the radiative contribution $\delta(x, y)$ is replaced by $\delta(x, y, x_\gamma)$ and integrated up to x_γ^{cut} .

2.2 Experimental status

The values assumed by q^2 , which can be positive or negative, determine the kind of process under study, which can be time-like or space-like. A measurement of the form factor using Dalitz decays belongs to the time-like region, since the photon converts into a e^+e^- pair and so q^2 is limited below by $2m_e$. Measurements of the TFF from π_D^0 decays have the advantage to be model-independent because they naturally occur at small q^2 values, but are limited by the statistics at large x where the effect of the slope is larger (see Figure 2.2). First experiments adopting this technique were very limited in the statistics and didn't include radiative corrections. The first experiment implementing them was the Geneva-Saclay experiment [18] in 1978, which reached the first remarkable result of establishing the positive sign of the slope. The most recent measurement of the TFF slope in the time-like region is the one performed by the NA62 experiment [36] analyzing the 2007 data, during the first phase of its experimental program. The analysis, based on a sample of 1.1 million π_D^0 decays, represents so far the most precise measurement (Figure 2.4), with a relative statistical uncertainty of $\sim 14\%$. It confirmed the positive sign of the slope parameter, with a 6σ significance, and was compatible with the *VMD* model expectation.

Experiments measuring the form factor in the space-like region exploit processes such as π^0 production cross-section in electron scattering events, where the measurement is done as a function of $Q^2 = -q^2$. Example of such a process is $e^+e^- \rightarrow e^+e^-\gamma^*\gamma^* \rightarrow e^+e^-\pi^0$, which can be performed in a "tagged" or "not-tagged" way. In both cases are admitted only measurements of $|F(q^2)|$ far from small values of q^2 , which are the ones less sensitive to the slope. Here the need to extrapolate from high Q^2 (that is $q^2 \rightarrow 0$), using specific models. This kind of approach was followed for example by the CELLO collaboration [4] in which a *VMD* fit was applied to the space-like

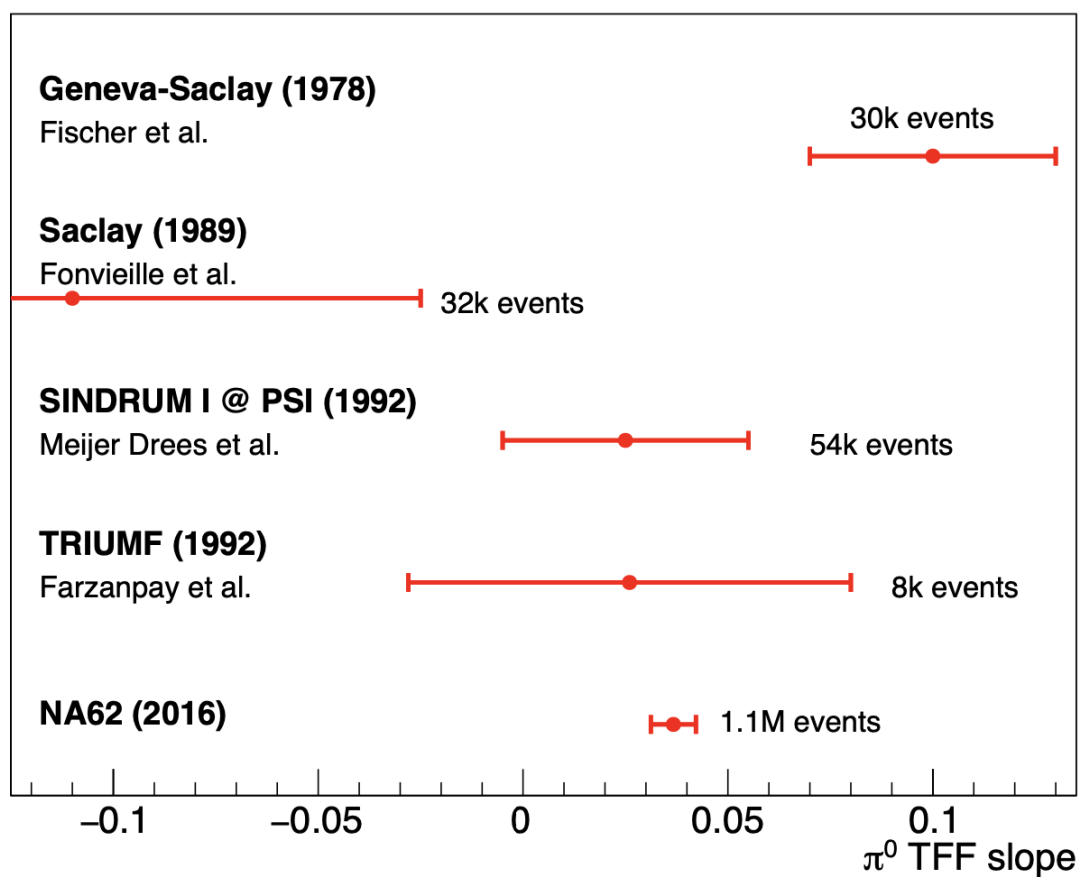


Figure 2.4: Summary of the TFF measurements in the time-like region. For each experiment is shown the number of collected π_D^0 decays [18, 19, 37, 17, 36]

data in the range $0.5 \text{ GeV}^2 < Q^2 < 2.7 \text{ GeV}^2$.

The world average for the slope parameter reported in [45] is

$$a = 3.35 \pm 0.31 \quad (2.22)$$

Other experiments, such as the Saclay collaboration [19] and TRIUMF [17] used different processes, like the charge-exchange process $\pi^p \rightarrow \pi^0 n$ and $\gamma p \rightarrow \pi^0 p$, reaching an average of

$$a = 0.034 \pm 0.009 \quad (2.23)$$

in agreement with theoretical prediction but not with a clear confirmation of a positive slope.

2.3 Analysis strategy

We adopted a similar analysis strategy to that of [18] and [36]. As in [36], we select π_D^0 decays from $K_{2\pi}$. During the Run1 period, which took place in the years 2017-2018, the NA62 experiment collected $\sim 7 \times 10^{12}$ kaon decays which corresponds to $\sim 1.4 \times 10^{12} \pi^0$. The signature of $K_{2\pi}$ decays with no neutrinos in the final state, is very clean and permits stringent constraints to be applied in the kinematic cuts. In section 2.4, we describe in detail the event selection criteria. Here we just mention that, given the detector acceptance, we expected to reconstruct $\approx 8 \times 10^6 \pi_D^0$ decays, leading to an improvement of the statistical uncertainty by a factor ~ 3 with respect to the previous measurement [36].

The signal selection is applied both on data and MC $K^+ \rightarrow \pi^+ \pi_D^0$ events. In this way two binned x distributions are obtained:

- the data distribution, expected to behave according to (2.12) with the unknown $a = a_{\text{true}}$;
- the MC distribution, whose values are known to be the result of a simulation based on (2.12) with a_{MC} .

The comparison of the two distributions reveals if the simulation is in agreement with the data or in other terms, assuming all the others contributions to (2.12) to be accurately simulated, if the value of a_{true} is in agreement with a_{MC} .

The agreement between the two distributions can be quantified with a χ^2 test, considering a x distribution divided in r bins. Calling n_i (m_i) the population of the i^{th} bin of the data (MC) distribution and

$$N = \sum_{\text{bins}} n_i \left(M = \sum_{\text{bins}} m_i \right), \quad (2.24)$$

n_i is expected to be distributed according to a Poisson distribution

$$\frac{e^{-Np_i} (Np_i)^{n_i}}{n_i!} \quad (2.25)$$

where p_i denotes the bin occupation probability. If the data and MC distributions originate from the same density function (2.12), then p_i is the same for both and then also

$$m_i \sim \frac{e^{-Mp_i} (Mp_i)^{m_i}}{m_i!} \quad (2.26)$$

Minimizing the likelihood

$$\frac{e^{-Np_i}(Np_i)^{n_i}}{n_i!} \frac{e^{-Mp_i}(Mp_i)^{m_i}}{m_i!} \quad (2.27)$$

with respect to p_i , one finds

$$\hat{p}_i = \frac{n_i + m_i}{N + M}. \quad (2.28)$$

Then, the form of the χ^2 is:

$$\chi^2 = \sum_{i=1}^r \frac{(n_i - N\hat{p}_i)^2}{N\hat{p}_i} + \sum_{i=1}^r \frac{(m_i - M\hat{p}_i)^2}{M\hat{p}_i} \quad (2.29)$$

In this sense, the best procedure would be to simulate various MC samples, each time with a different a_{MC} value and evaluating the χ^2 for each, the measurement corresponding to the generated slope which results in the minimum χ^2 . This approach is prohibitively time-consuming since requires both the generation and the reconstruction of many MC samples with the results that only few a_{MC} values can be tested. For example, the generation of 10^6 of MC events together with their reconstruction takes ~ 1 month.

An alternative is to reweigh each bin of the final x MC distribution with weights:

$$w = \frac{(1 + a_h \cdot x_{\text{true}})^2}{(1 + 0.032 \cdot x_{\text{true}})^2} \quad (2.30)$$

for different hypotheses a_h . This can be explained considering (2.12): if the population of the i^{th} bin of the MC distribution, simulated with a slope a , is m_i , and the population of the same bin simulated with a_h is m_i^h , the ratio between the two populations is

$$\frac{m_i^h}{m_i} = \frac{d\Gamma^h}{d\Gamma} = \frac{(1 + a_h x)^2}{(1 + ax)^2}. \quad (2.31)$$

Then, the population one would have obtained with a simulation with a_h is directly related to the one with a reference slope. In our case the reference slope is $a_{\text{MC}} = 0.032$, the world average before the measurement of [36].

The reweighing operation needs also a change in the χ^2 definition. The n_i are still expected to be distributed according to (2.25). The distribution of the sum of the weights in the i^{th} bin ($\equiv w_i$) is expected to be distributed according to a Gaussian distribution [21]

$$\frac{1}{\sqrt{2\pi s_i}} e^{-\frac{(w_i - W\hat{p}_i)^2}{2s_i^2}} \quad (2.32)$$

having indicated

$$W = \sum_{\text{bins}}^r w_i \quad (2.33)$$

and s_i the sum of the squares of the weights of events in the i^{th} bin. The $\hat{p}_i$ appearing in (2.32) is the estimator of the occupancy probability and the same which appears in (2.25), if the homogeneity hypothesis is true. In this case it is obtained by minimizing the likelihood

$$\frac{e^{-Np_i}(Np_i)^{n_i}}{n_i!} \frac{1}{\sqrt{2\pi s_i}} e^{-\frac{(w_i - W\hat{p}_i)^2}{2s_i^2}} \quad (2.34)$$

which gives

$$\hat{p}_i = \frac{Ww_i - Ns_i^2 + \sqrt{(Ww_i - Ns_i^2)^2 + 4W^2s_i^2n_i}}{2W^2} \quad (2.35)$$

The χ^2 definition becomes:

$$\chi^2 = \sum_{i=1}^r \frac{(n_i - N\hat{p}_i)^2}{N\hat{p}_i} + \sum_{i=1}^r \frac{(w_i - W\hat{p}_i)^2}{s_i^2} \quad (2.36)$$

Since w_i is a function of a_h , the quantities $\hat{p}_i$ and W are all functions of a_h and then

$$\chi^2 = \chi^2(a_h) \quad (2.37)$$

Minimizing χ^2 with respect to a_h gives the result of the TTF slope measurement. This procedure has a weakness: reweighing the final distribution implies that the only difference between two samples is in the number of "events" with a given x value that a certain slope produces with respect to another, without considering any changes in the acceptance. This is not true *a priori*. For example, consider the case of a slope a that produces certain values of x corresponding to the situation where the e^+e^- are emitted in the beam-pipe: the e^+e^- will be always out of the detector acceptance, producing a final empty x distribution. Instead, if a slope a' which produces values of x , or better e^+e^- tracks, in the detector acceptance, the final x distribution would have no empty bins. The passage between the former and the latter situation cannot be obtained only through a reweighing operation. To address this issue, which is relevant in case of empty bins in the reference distribution, one must assume the detector acceptance to depend only on x ; it is important that the MC reference distributions has no empty bins, to guarantee that the detector is not blind to any x value, thus justifying the reweighing procedure. The use of equipopulated bins avoids this problem and also permits to have in each bin a distribution of weights that is approximately Gaussian.

2.4 Event selection

The final state of the $K_{2\pi D}$ events is composed by three charged particles plus a photon, so we looked for events in which at least three tracks appear in the spectrometer, sharing the same origin (the vertex), and an energy cluster appears in the electromagnetic calorimeter, compatible with a photon signature, i.e. not assigned to any track of the event (as explained in detail in the following). While the momenta of the three charged particles is fully described by the spectrometer, the three-momentum of the photon was reconstructed by considering it propagating from the three-track vertex to the LKr cluster on the front face of the LKr calorimeter. Having the possibility to reconstruct all the momentum of the particles in the event, kinematic constraints were used to finally perform particle identification and also reduce the background contamination.

2.4.1 Data sample

Almost the full Run1 dataset was used for this analysis. It corresponds to the data collected in the years 2017-2018, with different beam intensities ($\approx 60 - 70\%$ the

nominal one). We considered only events passing a "*Restricted three track vertex*" filter, that excludes all those events without a three track vertex satisfying loose quality cuts, and triggered by the logic condition:

$$\text{Multitrack-electron: L0: RICH} * \text{QX} * \text{LKr20} + \text{L1: KTAG} \rightarrow \text{STRAWexo} \quad (2.38)$$

In the following, the meaning of each condition is described:

- RICH: coincidence of signals in at least three RICH photomultipliers (PMTs)
- QX: coincidence of signals in opposite quadrants of the NewCHOD
- LKr20: total energy deposit of at least 20 GeV in the LKr
- KTAG: a minimum of five out of eight KTAG sectors in coincidence and in time with the L0 trigger (to reject accidental L0 triggers)
- STRAWexo: requirement of at least a negative track in the spectrometer

The arrow in (2.38) indicates that the L1 algorithms are applied sequentially.

2.4.2 MC samples

We tuned our selection by measuring the background rejection after each selection cut, checking the data-MC agreement. We used MC samples simulated with both reference Runs (8094 for 2017 and 9001 for 2018). The signal sample is composed by two contributions: a standard sample, generated as described in section 2.1.1 with 100×10^6 generated events, and a "biased" sample with a cut on the generated value of $x > 0.5$, with 60×10^6 additional events¹. For what concerns the backgrounds, we used MC events from the decay channels listed in Table 2.1.

2.4.3 Vertex selection

The vertex is built using the Kalman filter approach described in [20]. Inputs to the vertex fit are the parameters $dx/dz|_i$, $dy/dz|_i$, $x|_i$, $y|_i$, $P|_i$ $i = 1, 2, 3$, being respectively the track slope along the x and y direction axis before the spectrometer magnet, the x and y positions of the track at the $z = 180$ m plane (slightly before the STRAW chamber 1), and the track momentum. The output of the algorithm is the vertex position and the χ^2 . We required the momentum of the vertex to be ≤ 78 GeV/ c and we enforced its quality by requiring a $\chi^2 \leq 25$. The Z position of the vertex (Z_{Vtx}) was required to be in the region $[105, 180]$ m (Figure 1.2). The vertex time, a weighted average of CHOD and NewCHOD times associated to the tracks, was also required to be within 5 ns of trigger time. This last cut was tuned by looking at the time difference distribution obtained from data, shown in Figure 2.5.

Among all the vertices, we selected only the ones in which all the tracks are in the spectrometer acceptance, to ensure their quality. Each track was required to have a momentum in the range $[5, 60]$ GeV/ c which is a range in which the spectrometer track calibration corrections are valid, and a $\chi^2_3 \leq 30$. Since $\pi^0 \rightarrow \gamma\gamma$ decays induce a

¹We used the biased sample only to perform the fit procedure and not for tuning the selection.

Decay channel	Number of events ($\times 10^6$)	Motivation
$K^+ \rightarrow \pi^+\pi^0, \pi^0 \rightarrow \gamma\gamma$	100	one photon converts in the spectrometer
$K^+ \rightarrow \pi^+\pi^0\pi^0, \pi^0 \rightarrow \gamma\gamma, \pi^0 \rightarrow e^+e^-\gamma$	50	the π^0 decaying into $\gamma\gamma$ is lost
$K^+ \rightarrow \pi^+\pi^0\pi^0, \pi^0 \rightarrow \gamma\gamma, \pi^0 \rightarrow \gamma\gamma$	100	the π^0 decaying into $\gamma\gamma$ is lost + one of the remaining photon converts into the spectrometer
$K^+ \rightarrow \pi^+\pi^+\pi^-$	260	misidentification of pions and in time cluster
$K^+ \rightarrow \pi^0e^+\nu, \pi^0 \rightarrow \gamma\gamma$	150	misidentification of e^+ +conversion of the photons in the spectrometer
$K^+ \rightarrow \pi^0e^+\nu, \pi^0 \rightarrow e^+e^-\gamma$	50	misidentification of e^+
$K^+ \rightarrow \pi^0\mu^+\nu, \pi^0 \rightarrow \gamma\gamma$	100	misidentification of μ^+ +conversion of the photons in the spectrometer
$K^+ \rightarrow \pi^0\mu^+\nu, \pi^0 \rightarrow e^+e^-\gamma$	20	misidentification of μ^+

Table 2.1: MC samples used for background studies, used to tune the event selection. In the last column a short explanation of the way the decay channel can possibly mimic the signal is reported.

dangerous background due to the photon conversion in the spectrometer, we reduced such a contamination by applying a cut of 20 mm on the minimum distance of the tracks at the first and second chamber of the spectrometer. As shown in Figure 2.6, this cut removes the majority of the $K_{2\pi}$ events.

2.4.4 Cluster selection

For each vertex, we try to assign a LKr cluster according to the following conditions:

- cluster distance from LKr "dead"² cells ≥ 20 mm;
- cluster energy ≥ 2 GeV, ensuring the validity of the LKr fine calibration;
- cluster distance from the LKr center within the interval [18, 107] cm, corresponding respectively to the beam pipe radius and the LAV 12 shadow;
- no association to any spectrometer track (i.e. track extrapolation at LKr is at least 50 mm away from the cluster position)

²A small set of LKr cells which are not properly working due to faulty electronics.

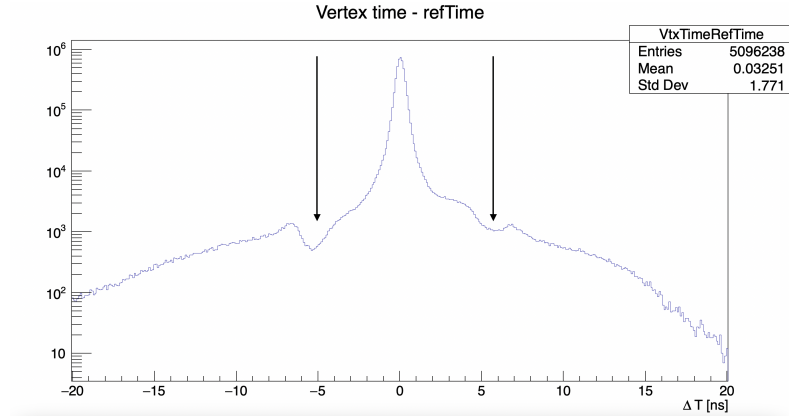


Figure 2.5: $T_{Vtx} - T_{trigger}$ distribution for the reconstructed three-track vertex time with respect to the trigger time in a single Run. The black arrows shows the ± 5 ns cut.

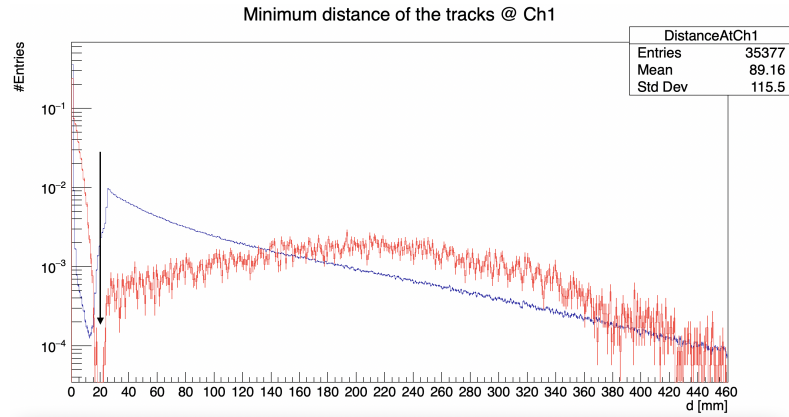


Figure 2.6: Distributions of minimum distance between the tracks at the first STRAW chamber for signal (blue) and $K_{2\pi}$ background (red) events. The black arrow indicates the value of the cut.

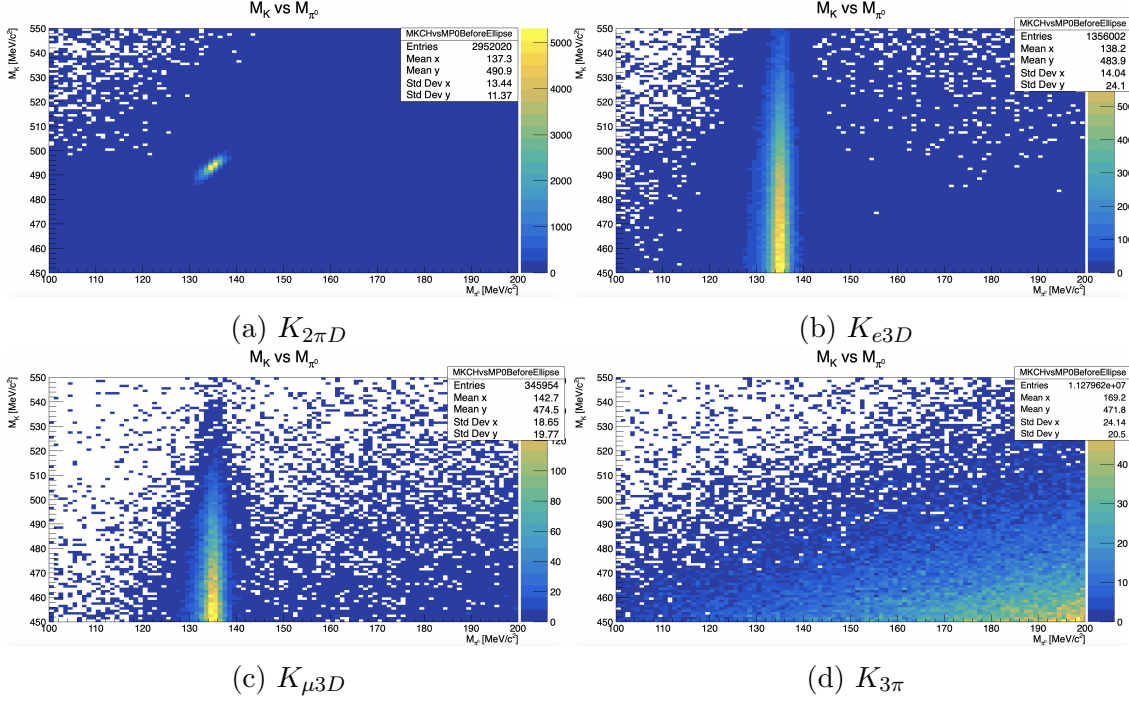


Figure 2.7: Reconstructed $M_{K^+}^{reco}$ vs $M_{\pi^0}^{reco}$ masses for different MC events.

- $|T_{cluster} - T_{Vtx}| \leq 5$ ns, timing cut between the cluster time, average of the times of the cells in the cluster, and the vertex time.

Defining $\vec{r}_{Vtx}$ and $\vec{r}_{cluster}$ the position of the vertex and of the cluster in the front face of the LKr calorimeter, we defined the direction of flight of the photon as

$$\hat{u}_\gamma = \frac{\vec{r}_{cluster} - \vec{r}_{Vtx}}{|\vec{r}_{cluster} - \vec{r}_{Vtx}|} \quad (2.39)$$

and we assigned the energy of the cluster as the magnitude of the photon momentum.

2.4.5 Kinematic cuts

A three-track vertex is defined by two positive and a negative track. Assuming that the negative track of the vertex is the electron, and given the reconstructed momentum of the photon as described in the previous section, the remaining ambiguity is related to the mass assignment to the positive tracks. The possible hypotheses are

$$hy1 : \text{Positive Track 1 : } e^+, \text{ Positive Track 2 : } \pi^+ \quad (2.40)$$

$$hy2 : \text{Positive Track 1 : } \pi^+, \text{ Positive Track 2 : } e^+ \quad (2.41)$$

Since the Dalitz decay is kinematically closed, we exploited the constraint on the π^0 and K^+ mass which must be satisfied by the reconstructed $e^+e^-\gamma$ (M_{π^0}) and $\pi^+e^+e^-\gamma$ (M_{K^+}) systems. Moreover, such kinematic constraint on the reconstructed masses contributes to reduce the backgrounds. Figure 2.7 shows the distribution of the reconstructed invariant mass of $M_{K^+}^{reco}$ vs $M_{\pi^0}^{reco}$ for signal and the most abundant background events. For a better handling of the systematics related to the invariant mass cuts, we decided to perform an elliptical cut in the $M_{K^+}^{reco}$ vs $M_{\pi^0}^{reco}$ plane. We estimated the ellipse parameters using on MC samples of the signal and using

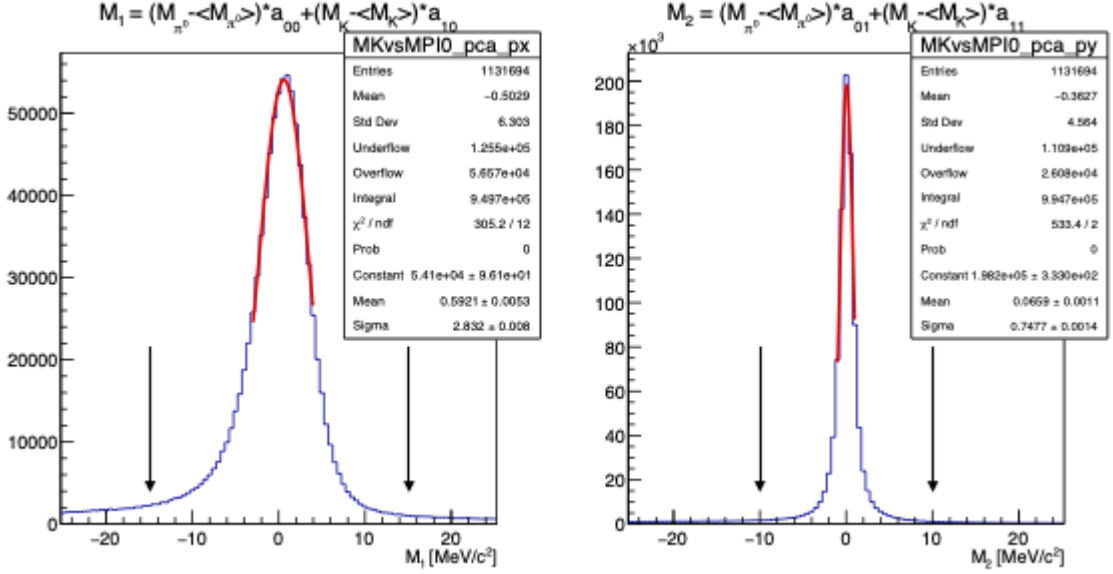


Figure 2.8: Distribution of the M_1 and M_2 distributions obtained after the rotation in the $M_{\pi^0}^{reco}$, $M_{K^+}^{reco}$ plane. The black arrow indicates the value of the applied cuts.

the Principal Component Analysis (PCA) method to identify the axes with larger variance. Defining such variables as M_1 and M_2 , they are linked to the old M_{π^0} and M_{K^+} values by the following relation:

$$M_1 = (M_{\pi^0} - \langle M_{\pi^0}^{reco} \rangle) a_{00} + (M_{K^+} - \langle M_{K^+}^{reco} \rangle) a_{10} \quad (2.42)$$

$$M_2 = (M_{\pi^0} - \langle M_{\pi^0}^{reco} \rangle) a_{01} + (M_{K^+} - \langle M_{K^+}^{reco} \rangle) a_{11} \quad (2.43)$$

where $\langle M_{\pi^0}^{reco} \rangle$, $\langle M_{K^+}^{reco} \rangle$ are the averages of the $M_{\pi^0}^{reco}$, $M_{K^+}^{reco}$ masses while

$$\begin{bmatrix} a_{00} & a_{01} \\ a_{10} & a_{11} \end{bmatrix} \quad (2.44)$$

is the rotation matrix between the old and the new set of axes. The values of a_{ij} are such that they represent a rotation of $\sim 64^\circ$ in the $M_{K^+}^{reco}$ vs $M_{\pi^0}^{reco}$ plane. Figure 2.8 shows the distributions of M_1 and M_2 . We chose a ± 15 MeV/c² and ± 10 MeV/c² cuts for the M_1 and M_2 distributions for simulated events respectively, corresponding to ≈ 5 and 14 standard deviations. While discussing the systematics uncertainties we studied the effect of varying these cuts. We selected only events in which only one of the hypotheses between (2.40) and (2.41) results in values inside such cuts. Moreover, we excluded events in which more than one vertex-cluster pair passes the selection. Figure 2.9 shows the data-MC comparisons at this step of the selection for the reconstructed M_{π^0} and M_{K^+} masses. A small shift is apparent: we considered the source of the shift as due to the LKr cluster fine energy calibration. To address this issue we developed the alternative calibration procedure described in Chapter 3, of which here we report the final result in Figure 2.10. Given the better data-MC agreement with the new calibration, we decided to adopt it for the rest of the analysis.

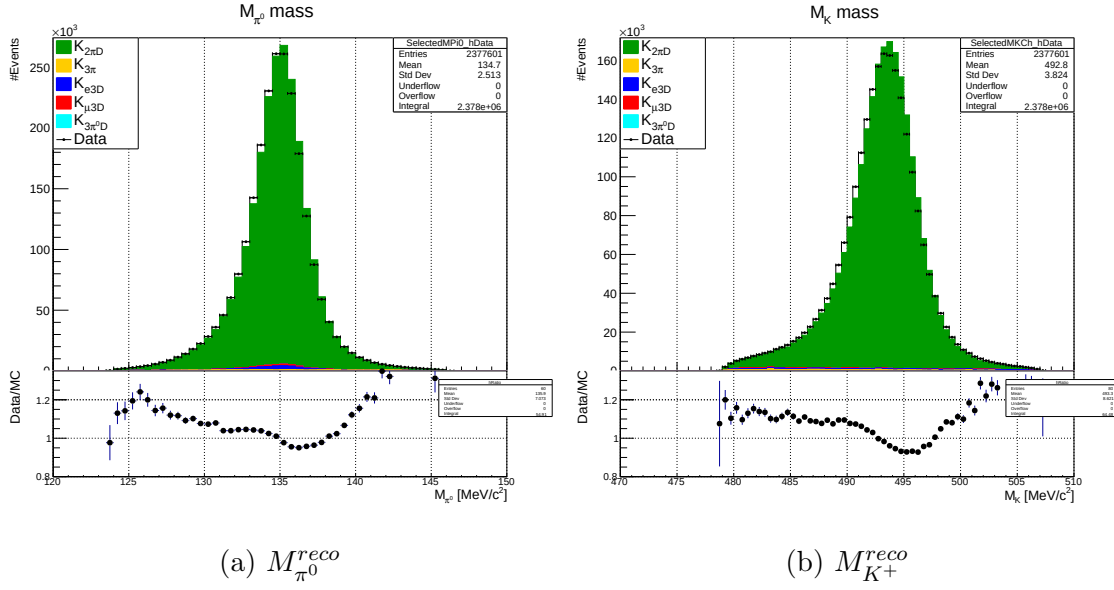


Figure 2.9: Data-MC comparison for the distribution of the reconstructed $M_{\pi^0}^{reco}$ and $M_{K^+}^{reco}$ using the LKr cluster fine energy calibration adopted as the default in the NA62 analysis framework. The shown MC samples are those reaching this step of the selection. A small shift in both distributions is visible.

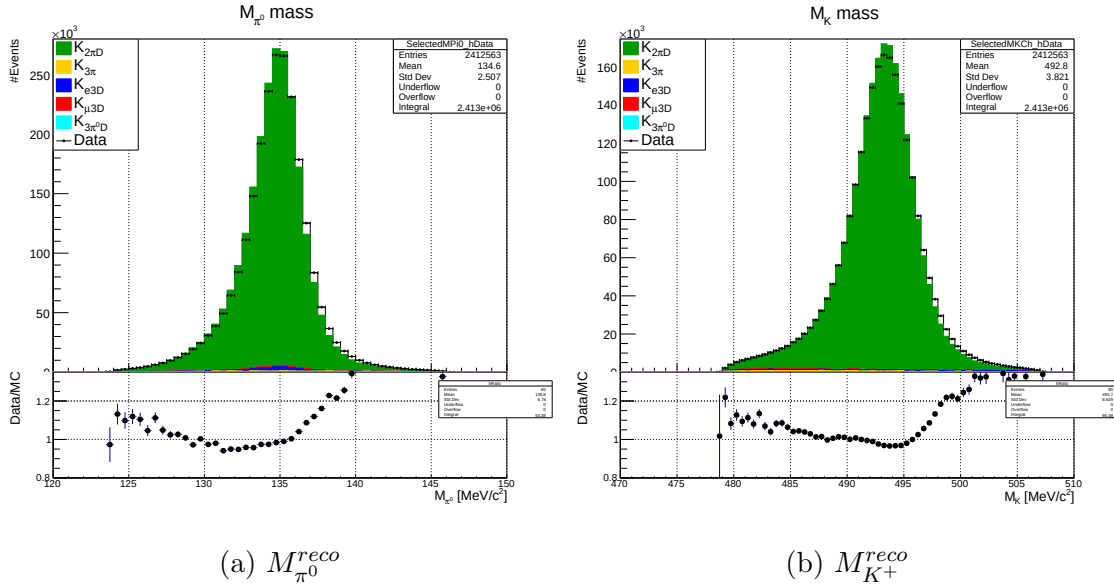


Figure 2.10: data-MC comparison for the distribution of the reconstructed $M_{\pi^0}^{reco}$ and $M_{K^+}^{reco}$ using the new LKr cluster fine calibration described in Chapter 3. The shift is seen to be reduced with respect to the case in which the standard calibration procedure is used.

2.4.6 Offline trigger conditions

A general procedure adopted also in the analysis is to implement a series of stricter cuts than those applied by the trigger mask during data collection. There are two main reasons for that:

- the *online* trigger efficiency is measured on events passing the selection criteria, so measuring it on events passing stricter cuts helps to obtain an almost constant efficiency over several data-taking periods, in this way independent from possible small shifts in the online trigger cut conditions, either due to changes in the detector performances or in the mask definitions
- to reproduce the *online* trigger efficiency on MC, we use trigger emulators, a set of software tools that emulate the trigger conditions such as the one described in (2.38). Since the simulation of the trigger decision might be imperfect at the edges of each cut composing the mask, having events passing a stricter *offline* condition made the verdict of the emulators more reliable.

We tightened the RICH condition by requiring that at least one track has four in time RICH hits, i.e. closer to the track time than 10, 6, 3 ns: the different cut values are used to take into account the different precision with which the track time was computed (with the spectrometer, NewCHOD, CHOD respectively). This requirement is stricter than the online one also because we required an "association" of track and a RICH fitted ring, i.e. a set of more than four PMTs that have been likely generated by a particle crossing the detector. If such an association works, this means that there is a track, compatible with a particle crossing the RICH volume, and so we expect the RICH trigger cut to be true, since the online condition requires only the presence of three independent hits to be true.

The QX condition is based on the tracks forming the vertex: we required that two of them are associated to hits in opposite quadrants in the NewCHOD, within 5 ns from the vertex time.

We enhanced the LKr20 condition by summing the energy of the isolated cluster and those associated to the tracks (if they are present), requiring such sum to be greater than 25 GeV.

The L1 conditions were emulated by requiring that at least a CEDAR candidate with more than 5 sectors fired is in time within 2 ns with respect to the vertex (KTAG condition), while the STRAW condition can be considered to be enforced by the quality cuts on the vertex.

The strongest impact on the selection is the QX condition, which reduces the acceptance by $\approx 35\%$. This happens because in signal events the tracks of e^+e^- are generally swept away by the spectrometer magnet due to their low momenta. The reconstructed three-momenta are shown in Figure 2.11.

2.4.7 Background suppression

As can be seen in Figure 2.11, four out of eight of the considered backgrounds survive at this stage of the selection. Among them the $K_{3\pi}$ and $K_{3\pi 0D}$ ones are the most problematic ones. Indeed from Figure 2.12 we can see that those decay channels are located mainly at large values of the x distribution, the most sensitive to the slope measurement. A cut was applied on the total momentum $\vec{P}_{tot} \equiv \vec{p}_{\pi^+} + \vec{p}_{e^+} + \vec{p}_{e^-} + \vec{p}_{\gamma}$,

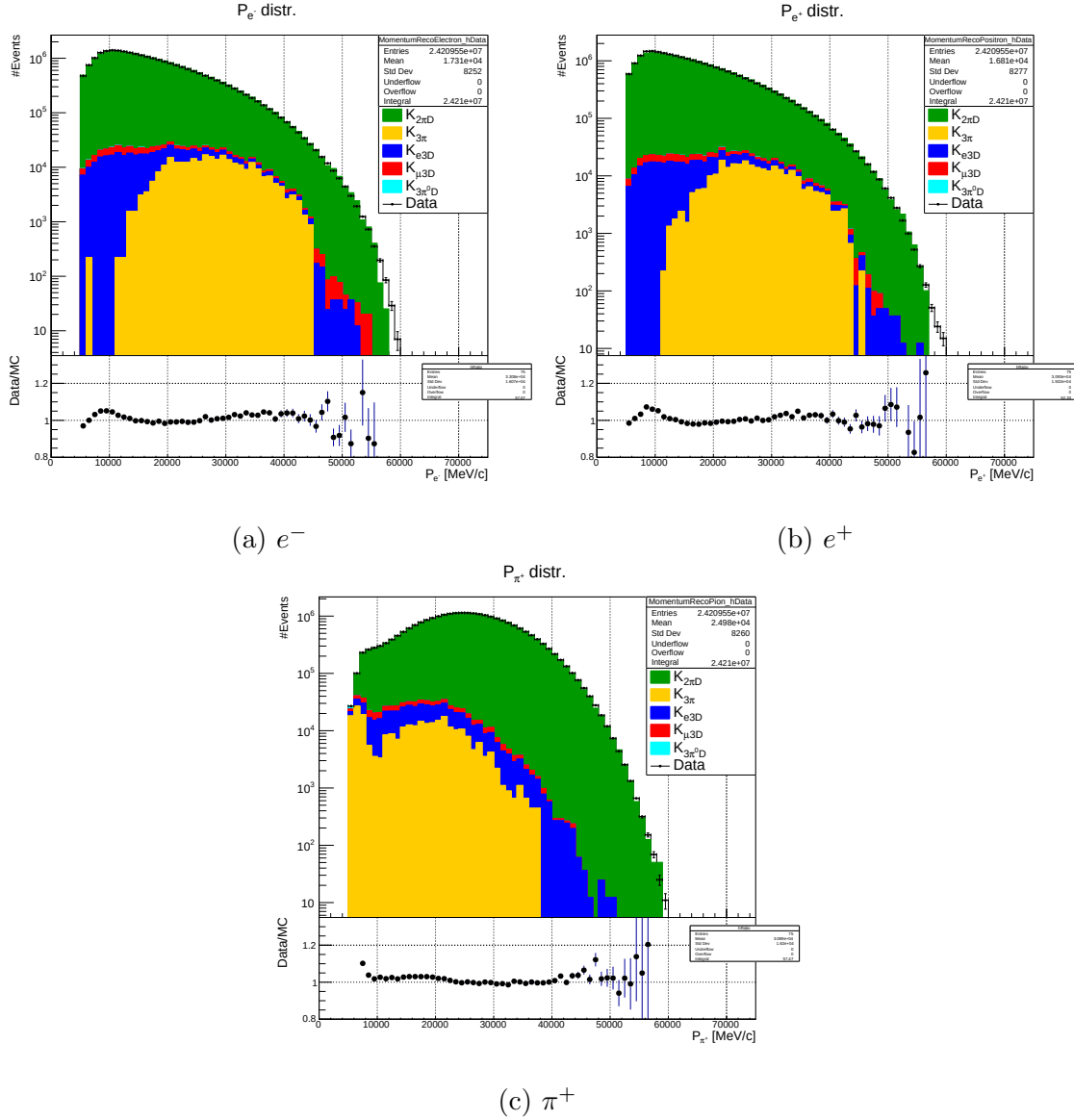


Figure 2.11: data-MC comparison for the distribution of the three-momentum for the e^+ , e^- , π^+ before the cut reproducing the online QX condition. Only the MC reaching this step of the selection are shown in the plot. Due to the low momenta of the electron and positrons, this condition reduces the acceptance of the selection.

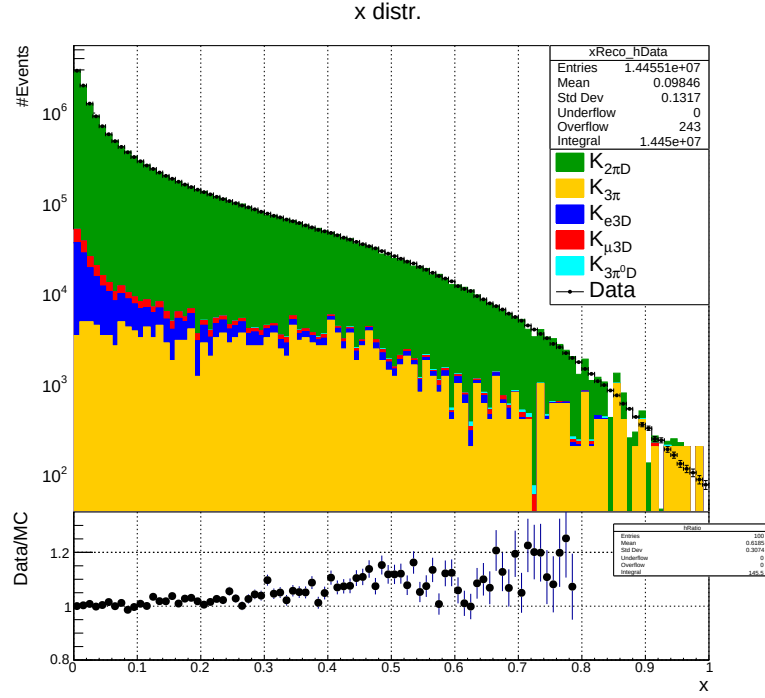


Figure 2.12: Distribution of the reconstructed x variable for data and MC after the application of all the offline trigger conditions. The $K_{3\pi}$ contamination at large value of x must be reduced to avoid a spurious slopes in the fit procedure.

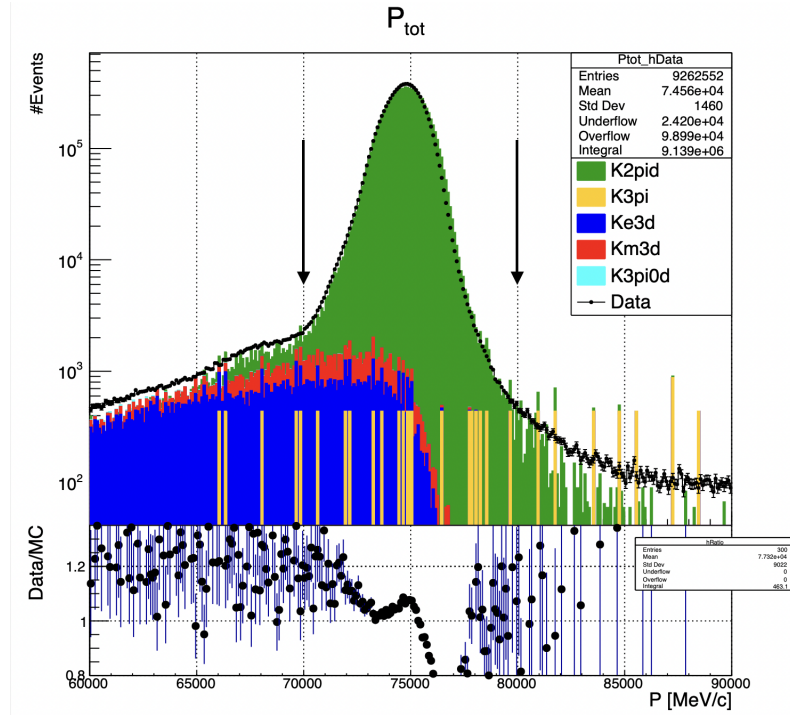


Figure 2.13: Distribution of the magnitude of the total momentum $\vec{P}_{tot}$ for data and MC. The black arrows indicates the region of the cut.

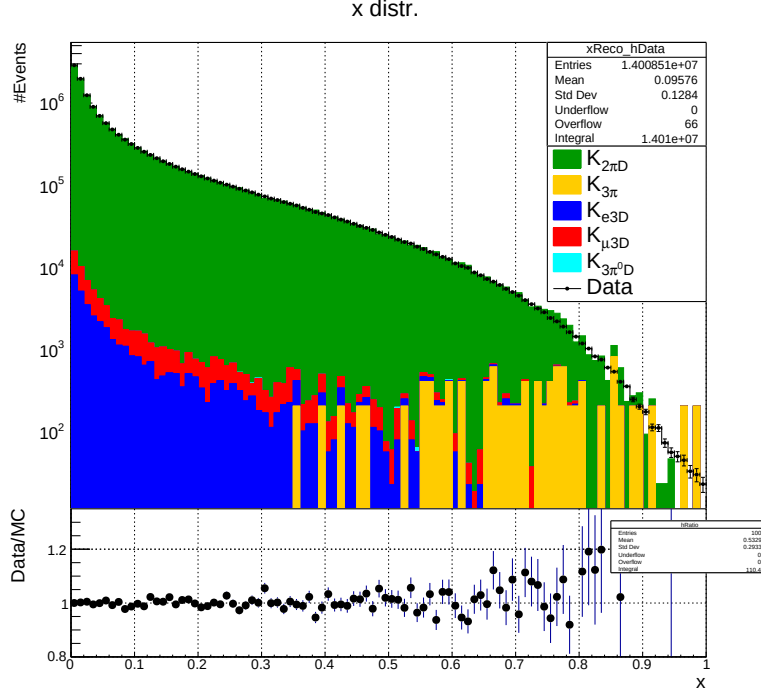


Figure 2.14: Distribution of the reconstructed x variable for data and MC samples. The plotted MC samples are those remaining at this step of the selection. The cut on P_{tot} and P_t help in reducing the $K_{3\pi}$ and $K_{3\pi^0D}$ components

requiring it to be in the $[70, 80]$ GeV/ c range (Figure 2.13) and to the transverse momentum $P_t \equiv \sqrt{P_{tot}^2 - P_l^2}$, where

$$\vec{P}_l = \frac{\vec{P}_{beam} * \vec{P}_{tot}}{|\vec{P}_{beam}|} \quad (2.45)$$

to be ≤ 50 MeV/ c . In our study we used the average beam momentum, which for the data and MC is extracted from the Gaussian fit of the $\vec{P}_{tot}$ distribution on selected $K_{3\pi}$ events. The fit is performed during the re-processing of a Run (data) or during the processing of the $K_{3\pi}$ sample (MC). The distribution after the cut is shown in Figure 2.14.

In order to remove the remaining contamination we exploited the E/p , which is the ratio of the energy of the LKr cluster associated to a spectrometer track and the momentum of the track. Since the LKr is an electromagnetic calorimeter, this ratio is peaked at 1 for tracks generated by e^+ and e^- , while it has a wide distribution from 0 to 1 for π^+ . Resolution effects smear the peak, so we classify a track as an $e^\pm$ if $E/p \geq 0.9$, and a π otherwise. We then require at least one of the following conditions to be satisfied:

$$E/p|_{\pi^+} < 0.9 * E/p|_{e^+} \geq 0.9 * E/p|_{e^-} \geq 0.9 \quad (2.46)$$

$$E/p|_{\pi^+} < 0.9 * e^+ \text{ not in LKr acceptance} * E/p|_{e^-} \geq 0.9 \quad (2.47)$$

$$E/p|_{\pi^+} < 0.9 * E/p|_{e^+} \geq 0.9 * e^- \text{ not in LKr acceptance} \quad (2.48)$$

We expect that these conditions are satisfied by the signal events (in particular the second and the third) due to the average low momenta of the tracks (see Figure

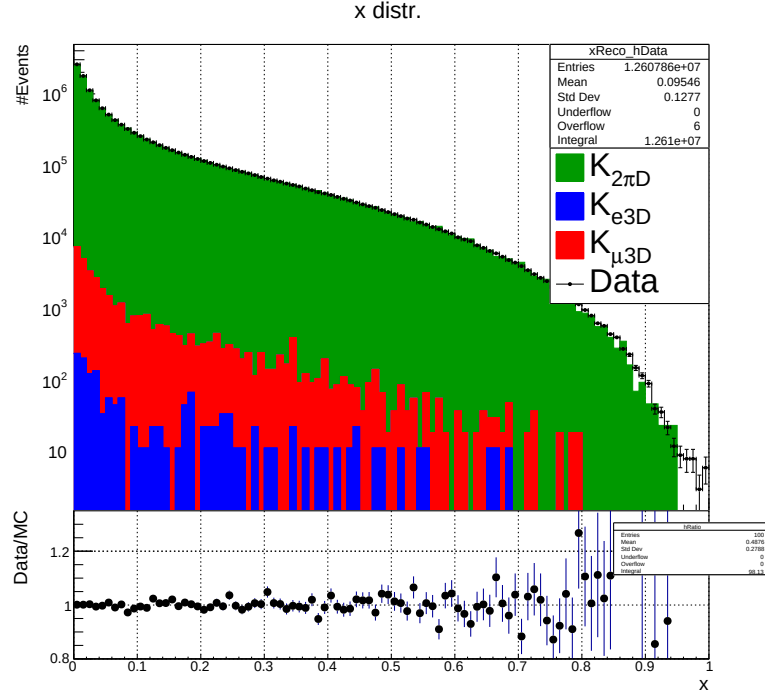


Figure 2.15: Distribution of the reconstructed x for data and remaining MC samples after the E/p cut.

2.11), with the majority of events having only one track between the e^+ and e^- reaching the LKr detector. We plot in Figure 2.15 the result of the cut with the remaining background contamination. The cut largely removed the $K_{3\pi}$ and $K_{3\pi 0D}$ backgrounds with a decrease of acceptance by 11%.

As a further possible source of background we studied also the contamination of the x distribution by mis-reconstruction of the e^+ in K_{e3D} decays. Indeed, in these decays two positrons are present, one from the main K^+ decay and the other by the π_D^0 : if the e^+ from the K^+ is reconstructed as the one of the π_D^0 , the resulting x distribution is modified. This is not directly visible in Figure 2.15 since in any case these decays are classified as a reconstructed K_{e3D} . We studied the probability of this using the K_{e3D} MC sample and we found that none of the 50×10^6 simulated events (see Table 2.1) causes such a reconstruction, leading to an upper limit of $p_{e^+ \leftrightarrow e^+} = 3.7 \times 10^{-8}$.

2.4.8 Trigger Efficiency

Given the level of purity reached by the selection, now including only Dalitz events, we performed a study of the trigger efficiency to spot eventual periods of low efficiency and remove them from the final collected sample. The L0 trigger efficiency is evaluated by studying the efficiency of its independent components: RICH, QX, LKr20. The algorithm for the RICH (NewCHOD) efficiency evaluation is the following

1. We select *Control* triggered events (based on CHOD)
2. We select events with at least one trigger primitive of the NewCHOD (RICH)

in the trigger slot

3. We consider as efficient events those in which the required primitive RICH (QX) is in the central slot or in the left or right one (section 1.3)

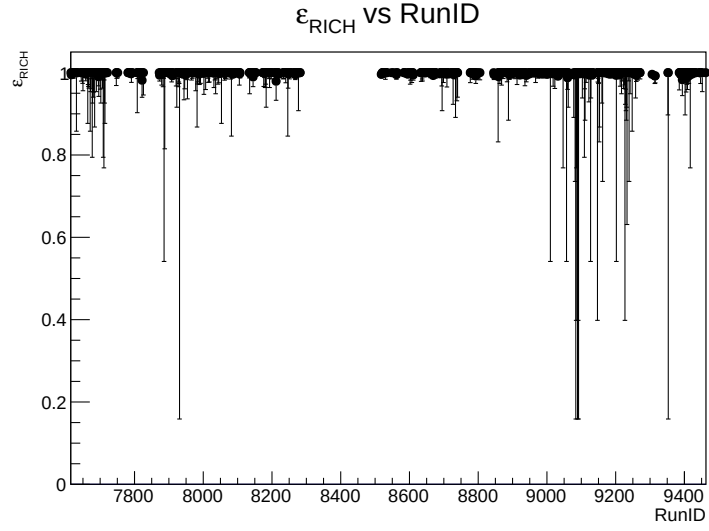
For what concerns the LKr20 primitive, the procedure is similar, the only change is that we use events satisfying the *Multitrack* condition $\text{RICH} * \text{QX}$ for the normalization sample, exploiting the large number of events passing this condition due to the lower downscaling with respect to the *Control* mask (100 instead of 400).

Also the L1 trigger efficiency is evaluated through the efficiency of each logic condition. In this case *Autopass* events were used. Considering that the KTAG, STRAWexo conditions are applied in cascade, we first checked the KTAG efficiency by computing the ratio between the number of events passing the KTAG L1 condition over all the *Autopass* events and then we computed the STRAWexo condition by computing the ratio of events passing such condition over the number of *Autopass* events which satisfy the KTAG condition.

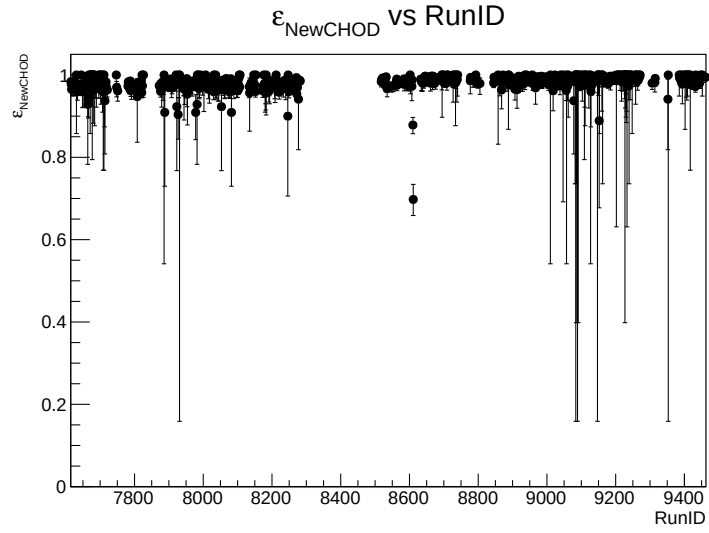
We performed our study over the whole 2017-2018 samples. The results are shown in Figure 2.16 and 2.17 for the L0 and L1 trigger level separately, as a function of the RunID. As expected, given the requirement described in section 2.4.6, the trend of the efficiency is almost flat and close to 1 for all the logic conditions. From Figure 2.16c, it can be seen that two data taking periods are characterized by a lower efficiency. This is a known problem, linked to hardware misbehaviour, therefore we decided to reject the events belonging to those periods. This caused a reduction of the data sample of $\approx 6.3\%$ events.

We also investigated in detail the efficiencies of all the data taking periods independently, and we rejected all the Runs in which such efficiencies differed by more than 3% with respect to the average efficiency of each period. The loss of events in this case is 4%.

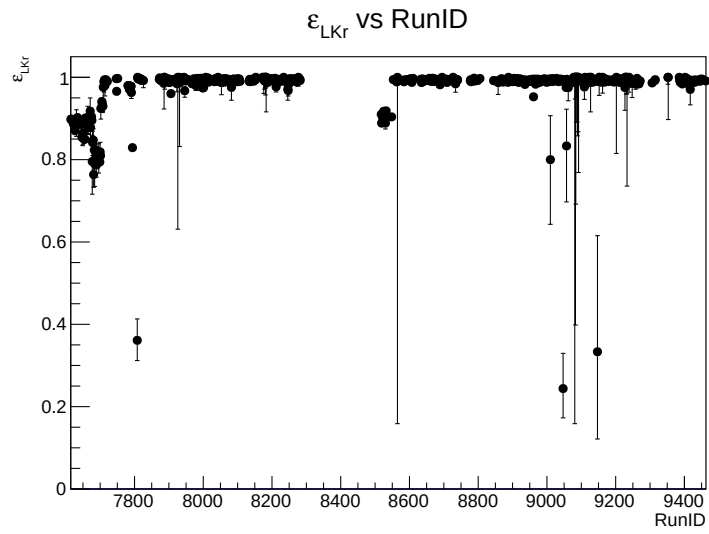
The evaluation of the L0 trigger efficiency for MC events was evaluated on $K_{2\pi D}$ events in the same way as for the data, with the only difference that we used the whole collected sample since the *Control* trigger condition is by default enabled for every simulated event. Similarly the L1 evaluation is based on the whole sample since all the events are labelled as *Autopass*. We first compared the efficiencies of the 2017 and 2018 reference MC independently with respect to all the data-taking periods of 2017 and 2018. Since our interest is on the x quantity, we report in Figures 2.18, 2.19, 2.20, 2.21, 2.22, 2.23, 2.24, 2.25, 2.26, 2.27 the efficiencies of each trigger condition with respect to the reconstructed x value. The MC simulation of each trigger condition is in agreement with the data in all the 2017 and 2018 samples. Since the number of *Control* events is limited in size, we also report in Figure 2.28 the same study but considering all the events of the 2017 and 2018 periods, and all MC events. The largest disagreement is in Figure 2.28e for small values of x . We cut those events by requiring $x > 0.01$, which in any case should not affected the measurement since the TFF slope effect is greater at large values of x (see Figure 2.2). We also show in Figures 2.29 the trigger efficiencies for the STRAWexo and LKr20 conditions as a function of the quantities on which they could have the most important impact: the $e^\pm$ and π^+ momenta for the former and the photon energy for the latter. Even in this case the data trend is well reproduced by the trigger emulators. We also found a good agreement between the data and MC x trends using the biased MC with x generated in the range $[0.5, 1]$.



(a) RICH

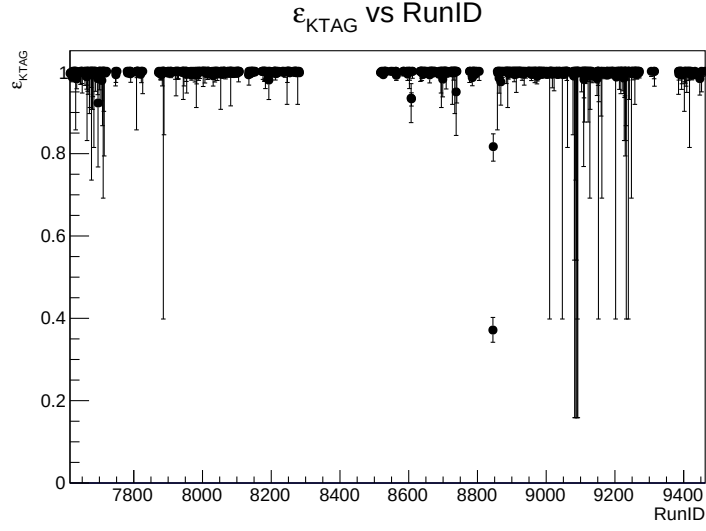


(b) QX

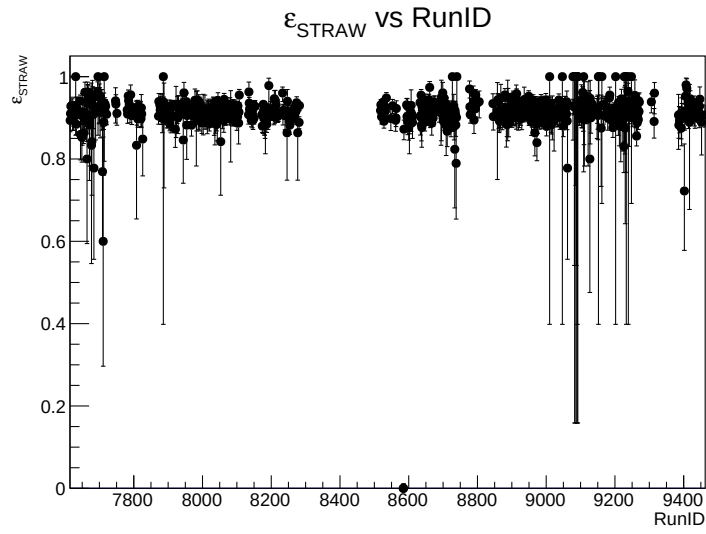


(c) LKr20

Figure 2.16: Trigger efficiency for each logic condition of the L0 trigger.



(a) KTAG



(b) STRAWexo

Figure 2.17: Trigger efficiency for each logic condition of the L1 trigger.

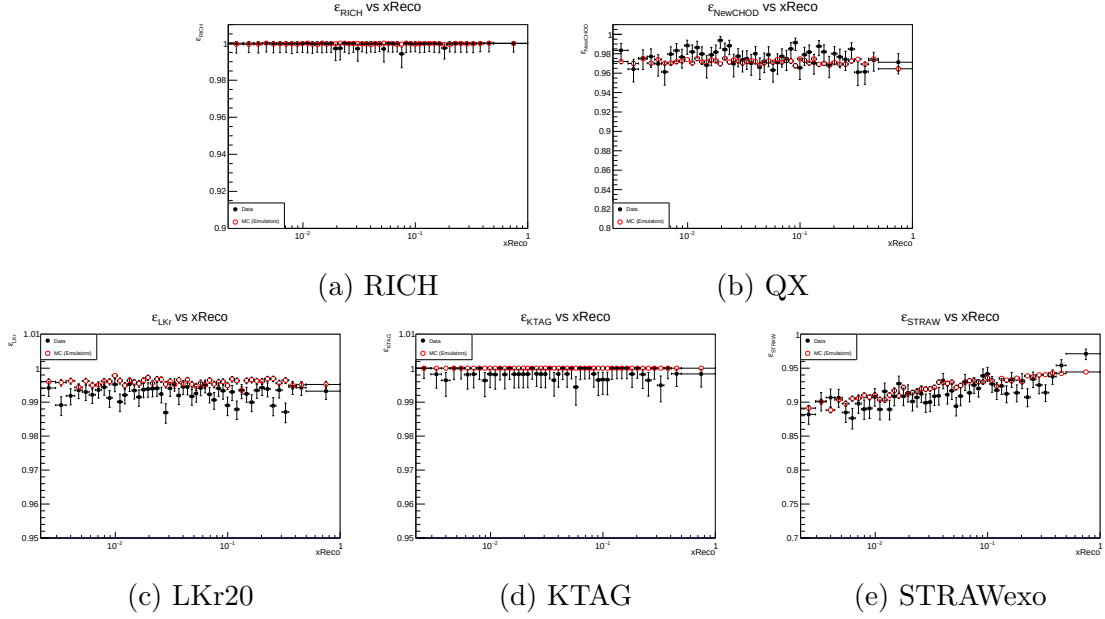


Figure 2.18: Data-MC Trigger efficiency comparison for each logic condition of the L0 and L1 trigger for the data samples of 2017A and the MC with reference conditions for 2017. The bins of the histogram are those that will be used later for the fit. For a description on the bin edges see section 2.5.1.

The data-MC agreement after the implementation of the cut is shown in Figures 2.30. The remaining discrepancies on E_γ and P_{e^-} , shown in Figure 2.30a and 2.30b respectively, are reduced by increasing the cut on the minimum value of $E_\gamma > 3$ GeV and $P_{e^-} > 6$ GeV/ c . Moreover we decided to symmetrize the track selections by increasing the cut also for P_{e^+} and P_{π^+} . The total reduction is about $\approx 25\%$. We studied the effect of these cuts in the stability checks on the final result.

In Table 2.2 we report the final numbers of events collected in all the used data samples while in Table 2.3, we report the acceptances of the selection on MC events.

Sample	Number of events
2017A	974282
2017B	1729576
2017C	210706
2018B	1174404
2018C	176690
2018D	1178742
2018E	941532
2018F	732614
2018G	1103817
2018H	468674
Total	8691037

Table 2.2: Number of collected events in each data sample after all cuts.

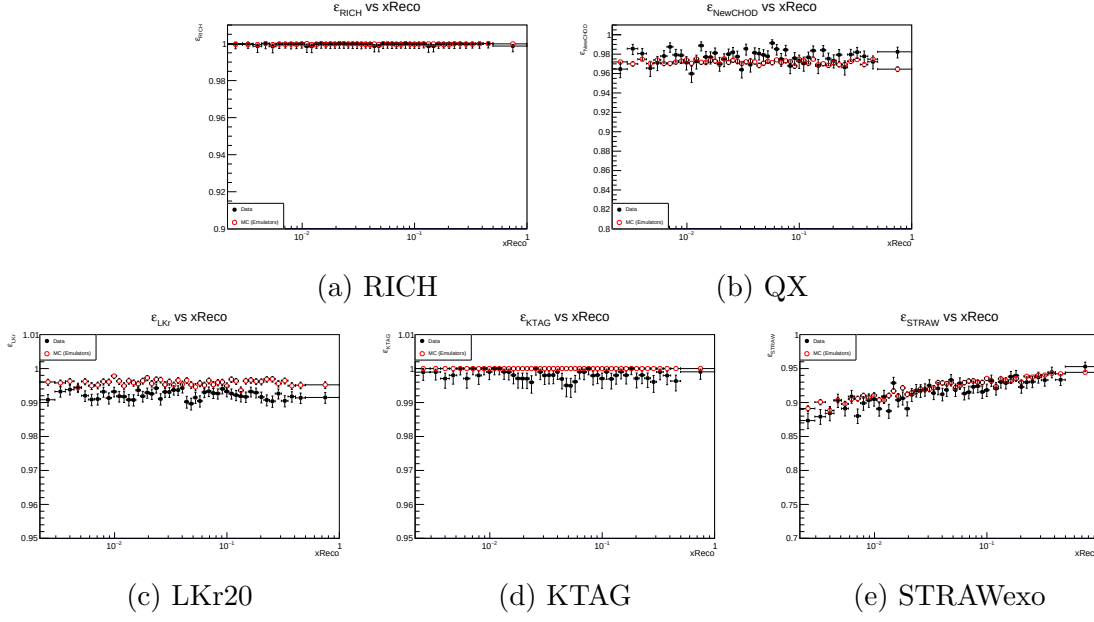


Figure 2.19: Data-MC Trigger efficiency comparison for each logic condition of the L0 and L1 trigger for the data samples of 2017B and the MC with reference conditions of 2017. The bins of the histogram are those that will be used later for the fit. For a description on the bin edges see section 2.5.1.

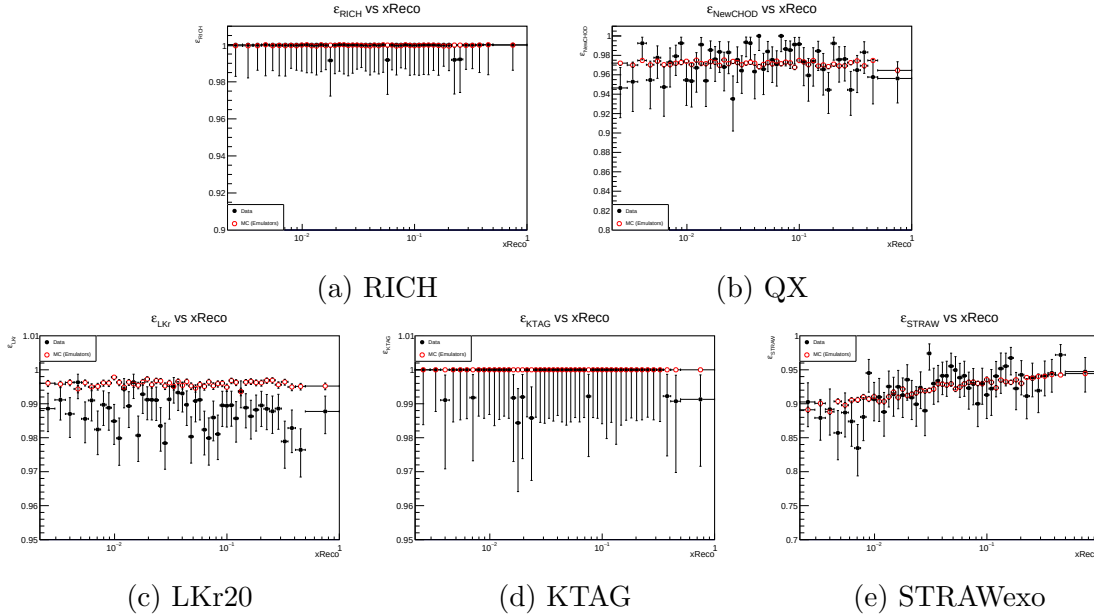


Figure 2.20: Data-MC Trigger efficiency comparison for each logic condition of the L0 and L1 trigger for the data samples of 2017C and the MC with reference conditions of 2017. The bins of the histogram are those that will be used later for the fit. For a description on the bin edges see section 2.5.1.

2.4. EVENT SELECTION

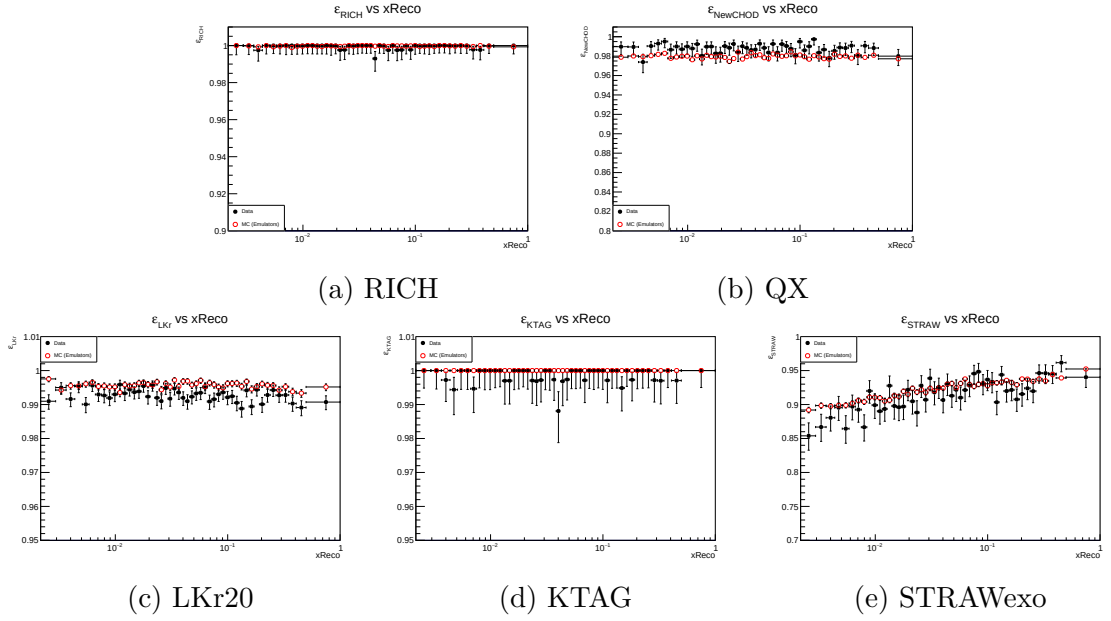


Figure 2.21: Data-MC Trigger efficiency comparison for each logic condition of the L0 and L1 trigger for the data samples of 2018B and the MC with reference conditions of 2018. The bins of the histogram are those that will be used later for the fit. For a description on the bin edges see section 2.5.1.

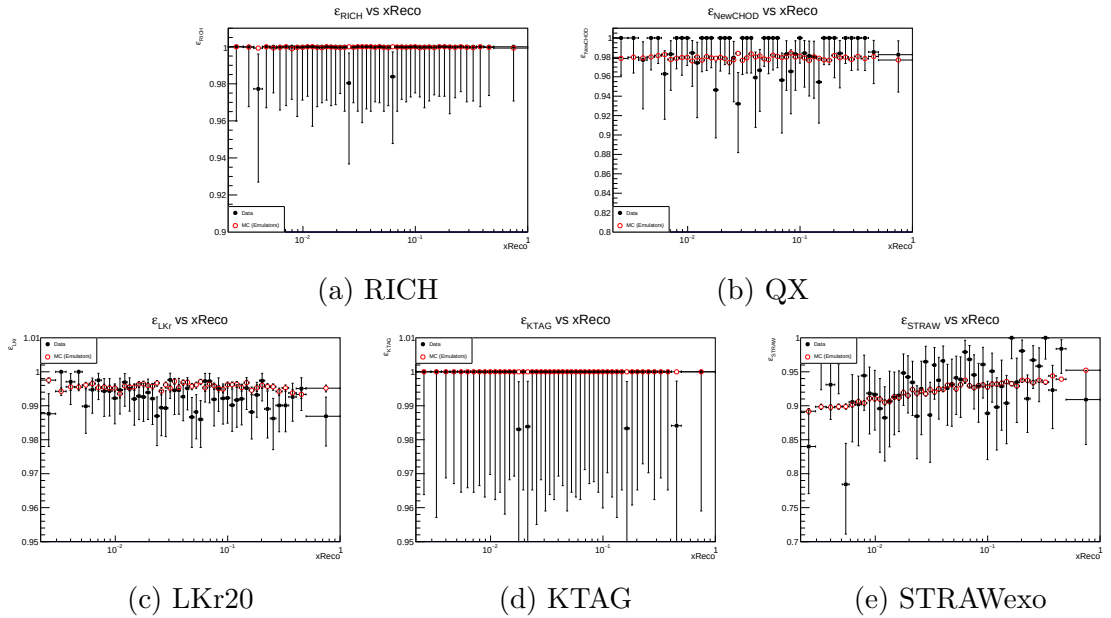


Figure 2.22: Data-MC Trigger efficiency comparison for each logic condition of the L0 and L1 trigger for the data samples of 2018C and the MC with reference conditions of 2018. The bins of the histogram are those that will be used later for the fit. For a description on the bin edges see section 2.5.1.

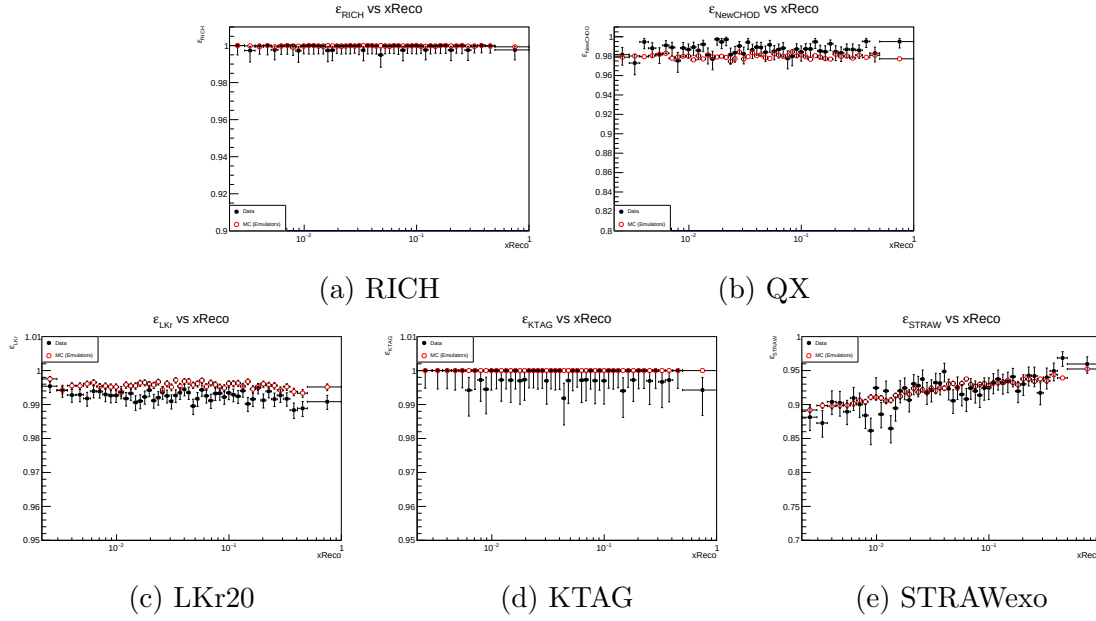


Figure 2.23: Data-MC Trigger efficiency comparison for each logic condition of the L0 and L1 trigger for the data samples of 2018D and the MC with reference conditions of 2018. The bins of the histogram are those that will be used later for the fit. For a description on the bin edges see section 2.5.1.

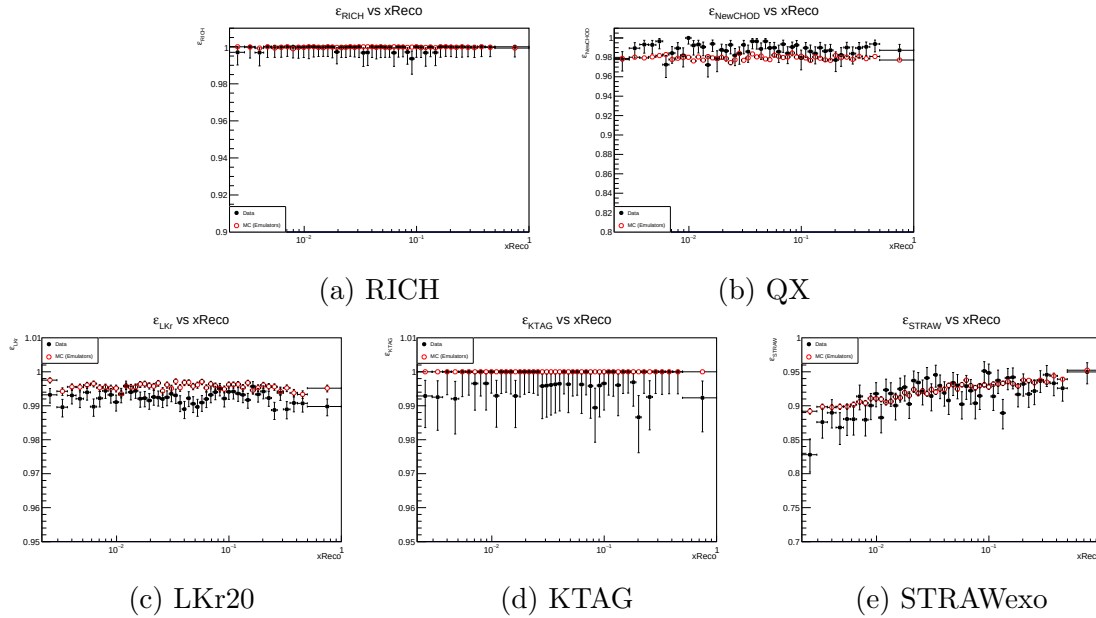


Figure 2.24: Data-MC Trigger efficiency comparison for each logic condition of the L0 and L1 trigger for the data samples of 2018E and the MC with reference conditions of 2018. The bins of the histogram are those that will be used later for the fit. For a description on the bin edges see section 2.5.1.

2.4. EVENT SELECTION

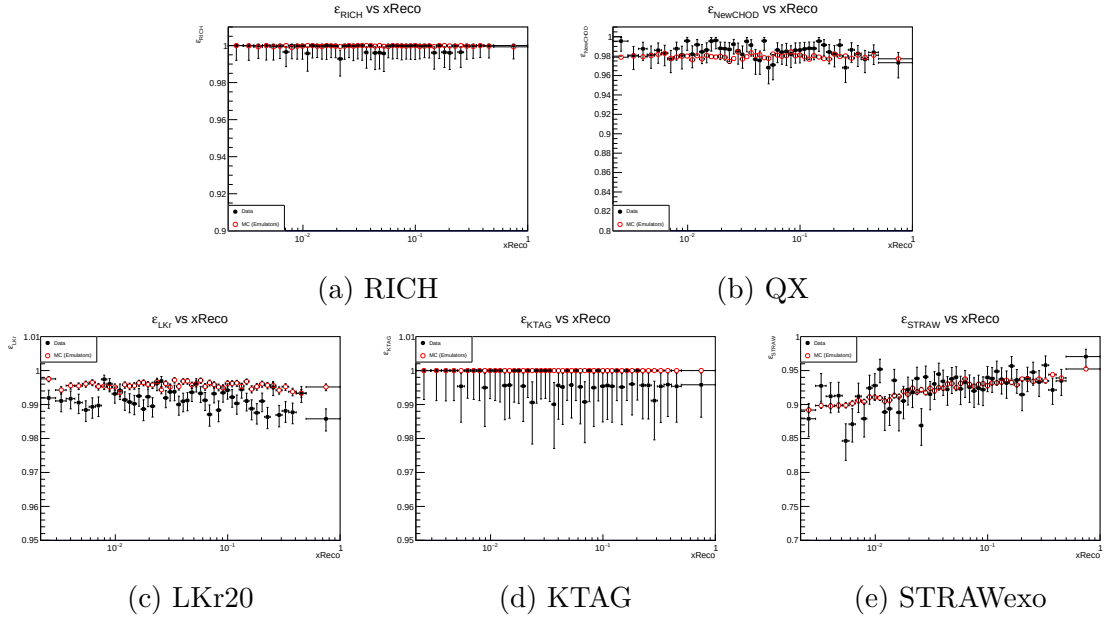


Figure 2.25: Data-MC Trigger efficiency comparison for each logic condition of the L0 and L1 trigger for the data samples of 2018F and the MC with reference conditions of 2018. The bins of the histogram are those that will be used later for the fit. For a description on the bin edges see section 2.5.1.

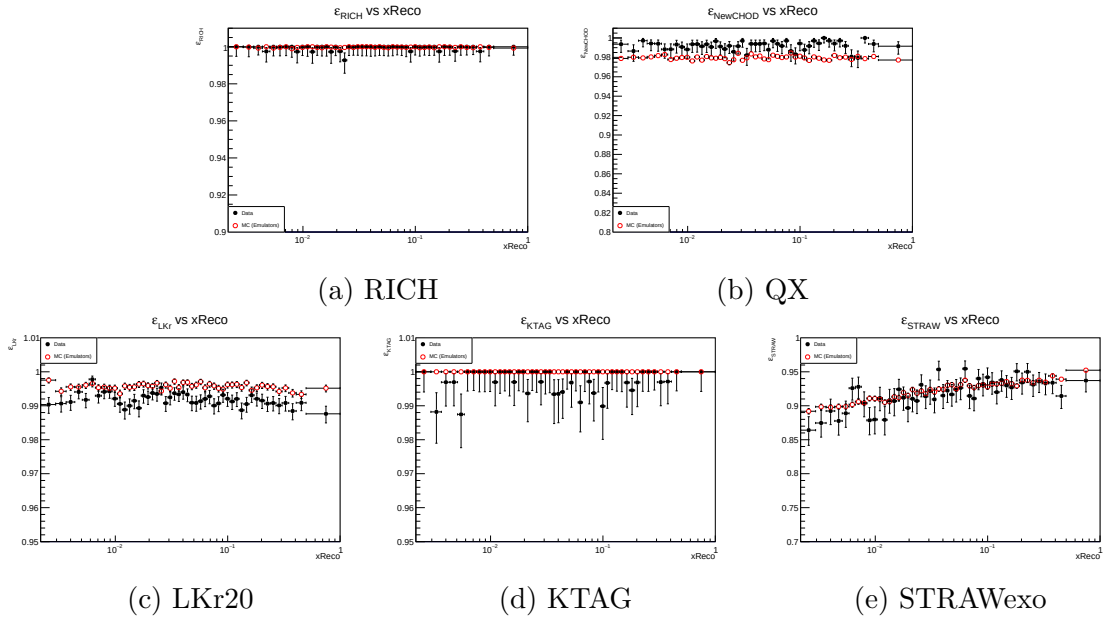


Figure 2.26: Data-MC Trigger efficiency comparison for each logic condition of the L0 and L1 trigger for the data samples of 2018G and the MC with reference conditions of 2018. The bins of the histogram are those that will be used later for the fit. For a description on the bin edges see section 2.5.1.

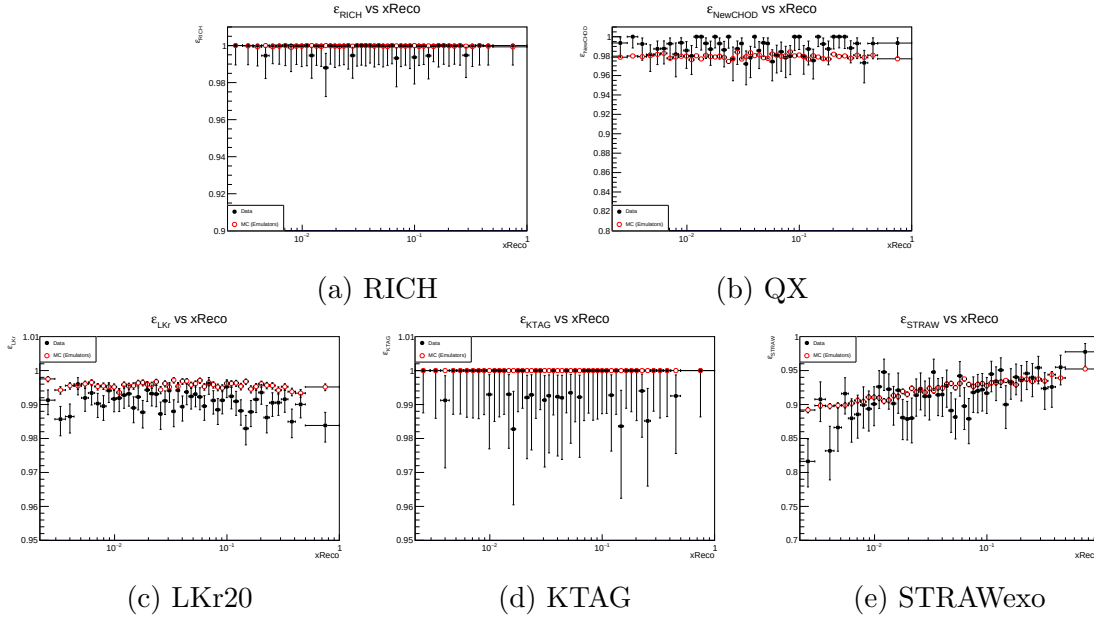


Figure 2.27: Data-MC Trigger efficiency comparison for each logic condition of the L0 and L1 trigger for the data samples of 2018H and the MC with reference conditions of 2018. The bins of the histogram are those that will be used later for the fit. For a description on the bin edges see section 2.5.1.

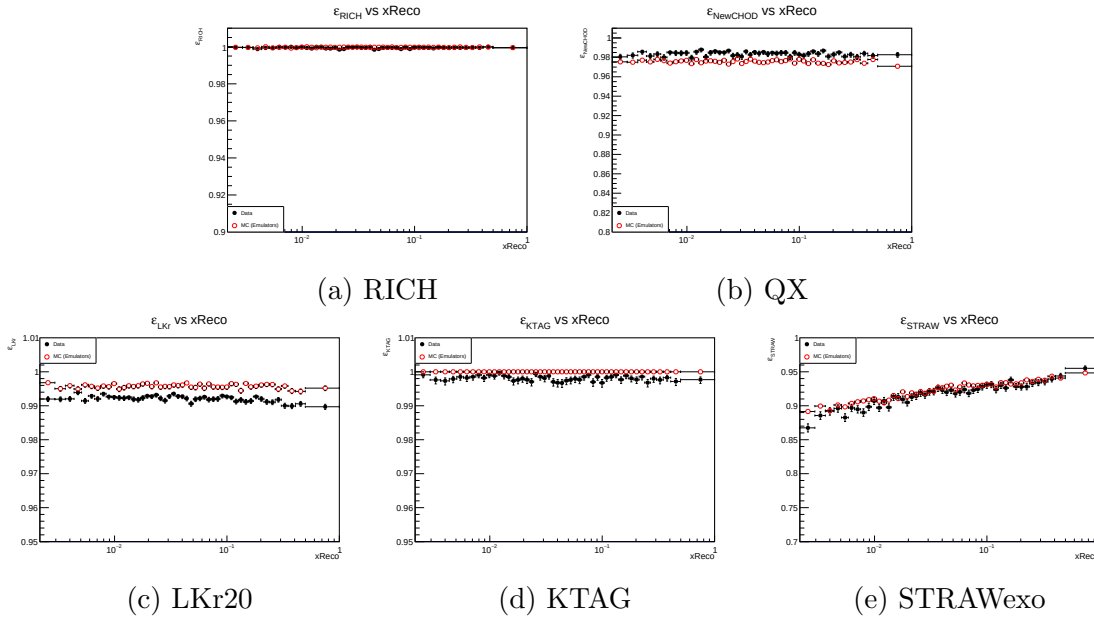


Figure 2.28: Data-MC Trigger efficiency comparison for each logic condition of the L0 and L1 trigger for the data samples of 2017+2018 and the MC with reference conditions of 2017+2018. The bins of the histogram are those that will be used later for the fit. For a description on the bin edges see section 2.5.1.

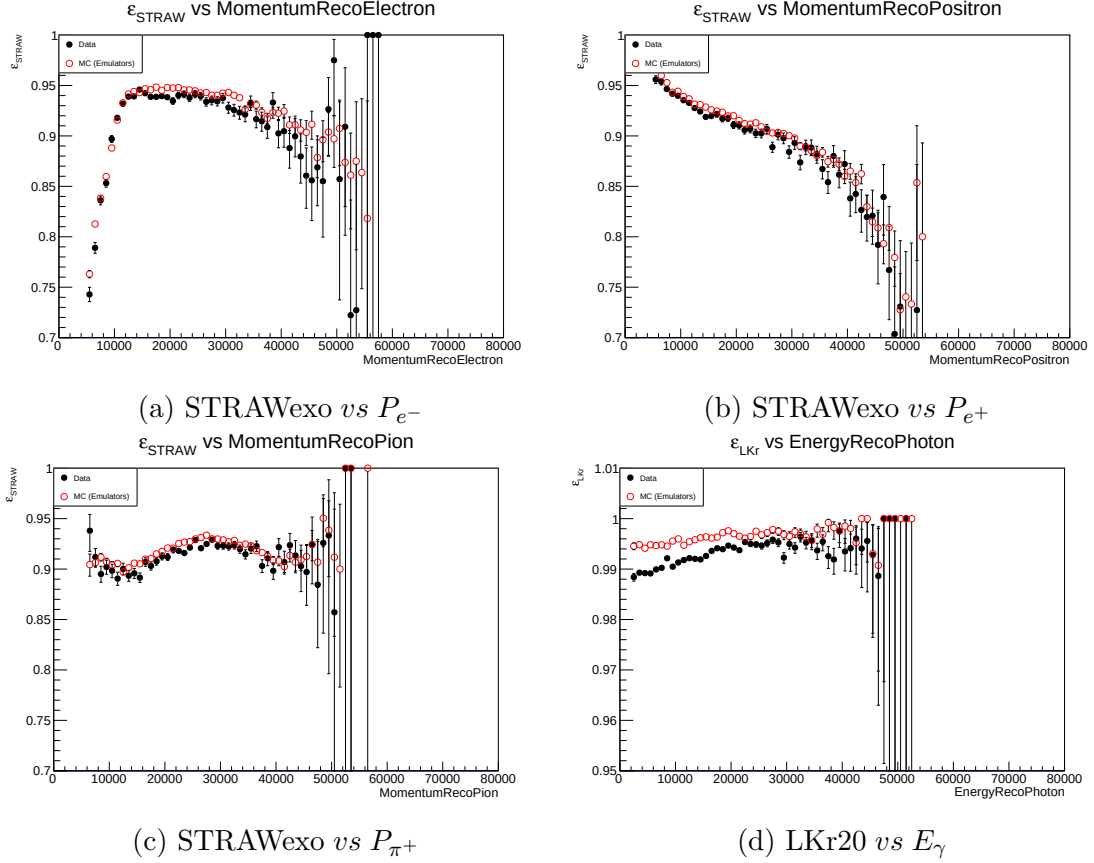


Figure 2.29: Data-MC Trigger efficiency comparison for the STRAWexo and LKr20.

Decay channel	Acceptance (%)
$K^+ \rightarrow \pi^+ \pi^0, \pi^0 \rightarrow e^+ e^- \gamma$	0.41317 ± 0.00065
$K^+ \rightarrow \pi^0 e^+ \nu, \pi^0 \rightarrow e^+ e^- \gamma$	0.000219 ± 0.000021
$K^+ \rightarrow \pi^0 \mu^+ \nu, \pi^0 \rightarrow e^+ e^- \gamma$	0.00720 ± 0.00019
N_{MC}	401843

Table 2.3: Acceptance of the MC samples at the end of the selection. The signal is highlighted in red. MC samples with negligible acceptance are not reported. The last row indicates the final number of selected all the MC events.

2.5 TFF Fit

2.5.1 Equipopulated bins

To perform the fit procedure indicated in section 2.3, equipopulated bins must be used. To compute the bin edges we fixed the number of bins and we computed the average number of elements per bin as

$$n = \frac{N}{n_{\text{Bin}}} \quad (2.49)$$

where N is the size of the data sample. Then, after ordering in ascending order the reconstructed x values, each time the count reaches a multiple of n , we fix an edge with

$$x_{\text{edge}} = \frac{x_n + x_{n+1}}{2} \quad (2.50)$$

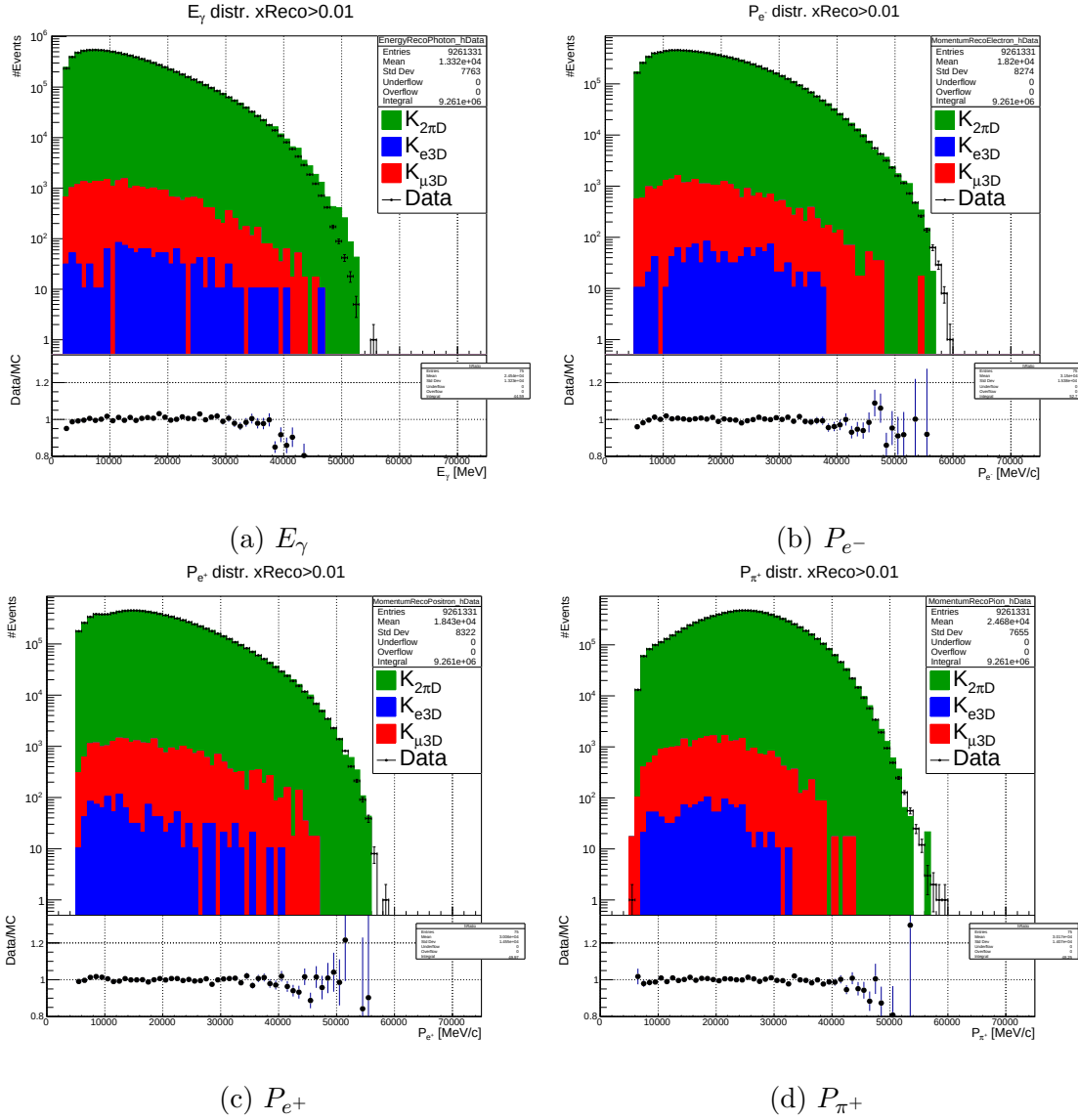


Figure 2.30: Data-MC comparison for the 2017+2018 data sample and the remaining MC samples after the cut on x . While the distribution of P_{π^+} and P_{e^+} are well described by the MC, the E_γ and P_{e^-} distribution exhibit a little discrepancy at small values. The E_γ discrepancy could be due to the efficiency of the LKr fine correction, which is only defined above 2 GeV (see Figure 3.5). The P_{e^-} discrepancy could be due to a remaining difference between the trigger and the MC trigger emulation.

2.5.2 First Measurement

The minimization of the χ^2 defined in equation (2.36) was done using

- the TMinuit class of ROOT ([7], [8]), that implements MINUIT package for the numerical minimization
- the χ^2 test routine available in the ROOT TH1 class, that implements an option for comparison between an unweighted histogram and a weighted one³. Using this option the χ^2 defined in (2.36) is used.

Technically, our algorithm filled a histogram with reconstructed x values from the data sample only once, each entry with a weight equal to one. Then, in the function minimized numerically by MINUIT, during each iteration, we filled a second histogram with reconstructed x values from MC events, each one weighted with (2.30), each time testing the value of a_h performing the χ^2 test between the data histogram and the MC one. Since there are residual backgrounds in the final samples (Table 2.3), before performing the χ^2 test, we reweighted also each MC sample with the corresponding acceptance and branching ratio. The result of the procedure with the 2017-2018 data samples is:

$$a \pm \delta a_{stat} = (3.54 \pm 0.62) \times 10^{-2} \quad (2.51)$$

where the statistical uncertainty is composed by

$$\delta a_{MC} = 0.61 \times 10^{-2} \quad (2.52)$$

$$\delta a_{Data} = 0.13 \times 10^{-2} \quad (2.53)$$

as a consequence of the available statistics of the MC sample.

2.5.3 Inclusion of the biased signal sample

With the described approach we reached a $\approx 17\%$ relative statistical uncertainty mostly due to the lack of statistics in the MC sample. To improve our precision, we included the biased signal sample described in section 2.4.2, in which $K_{2\pi D}$ events are generated in the region $x > 0.5$. The radiative corrections are the same as for the standard MC sample. The use of this sample requires some care in the MC scaling between samples, since the standard MC and the biased one only overlap in the region $x > 0.5$. We decided to cut the standard MC events by considering only those with a $x_{true} < 0.5$ and complete the x distribution with the biased sample. We also changed the branching ratio used for the standard sample as

$$BR(\pi_D^0) \rightarrow BR(\pi_D^0)[1 - BR(\pi_D^0(x > 0.5))] \quad (2.54)$$

To evaluate $BR(\pi_D^0(x > 0.5))$ we normalized to 1 the generated distribution of x_{true} for the standard $K_{2\pi D}$ MC sample, and we took the integral from 0.5 to 1. This empirical solution was preferred with respect to the direct integration of (2.12),

³The routine is described here: <https://root.cern.ch/doc/master/classTH1.html#a6c281eebc0c0a848e7a0d620425090a5>

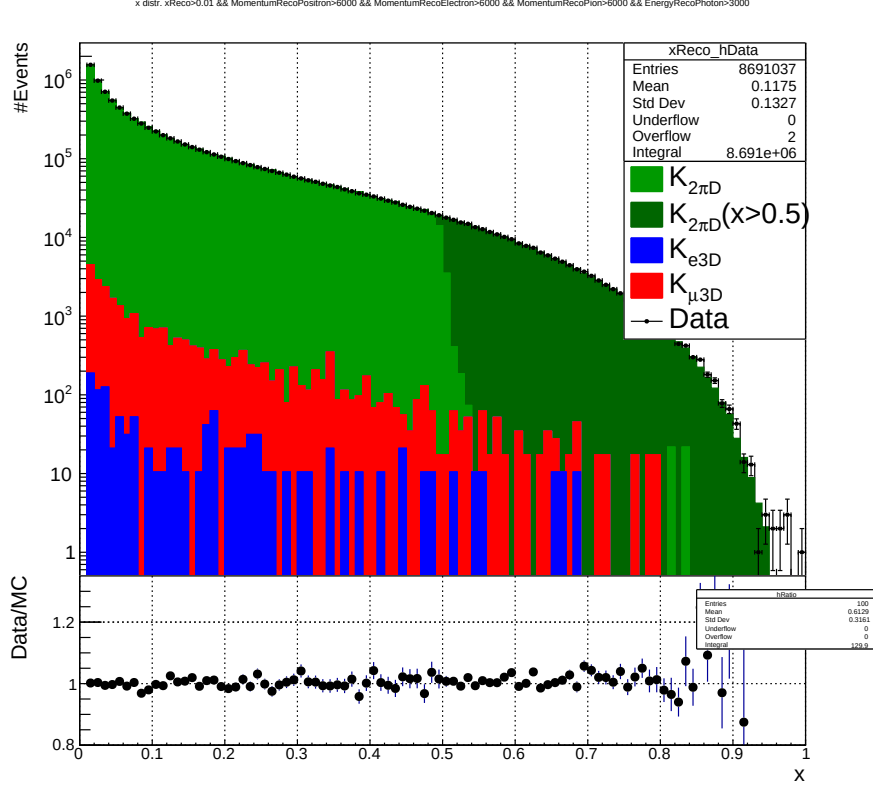


Figure 2.31: Data/MC comparison for distribution of the reconstructed x value. The data represent all the events of the 2017+2018 samples, while the MC are those remaining after applying all the cuts of the analysis. The scaling procedure described in the text has been applied for the standard $K_{2\pi D}$ and biased $K_{2\pi D}$ samples. Note that there is no discontinuity of the two samples at $x = 0.5$. The fact that some "standard" events fall in the region with $x > 0.5$ is due to resolution effects.

which presents difficulties linked to the evaluation of the radiative corrections, and it is based on the fact that

$$BR(\pi_D^0) \propto \int_{x^2}^1 \frac{d\Gamma}{dx} \propto N_{events} \text{ in the } x_{true} \text{ distribution} \quad (2.55)$$

This value depends from the slope parameter itself and for $a_{MC} = 0.032$ it gives

$$BR(\pi_D^0(x > 0.5)) = 0.00329 \quad (2.56)$$

We plot in Figure 2.31 the data-MC agreement of the reconstructed x distribution to show the result of our scaling. In Figure 2.32 we show how with the inclusion of the biased MC sample, the data-MC agreement for reference quantities improves with respect to 2.30. By including these samples (one with reference Run 8094 and another with reference Run 9001), the number of collected MC events increased to 2.2×10^6 .

2.5.4 Improved Measurement and final result

To take into account the biased MC sample, we implemented a procedure for the correction of the $BR(\pi_D^0)$ and of $N_{K_{2\pi D}, x<0.5}$, $N_{K_{2\pi D}, x>0.5}$, respectively the number

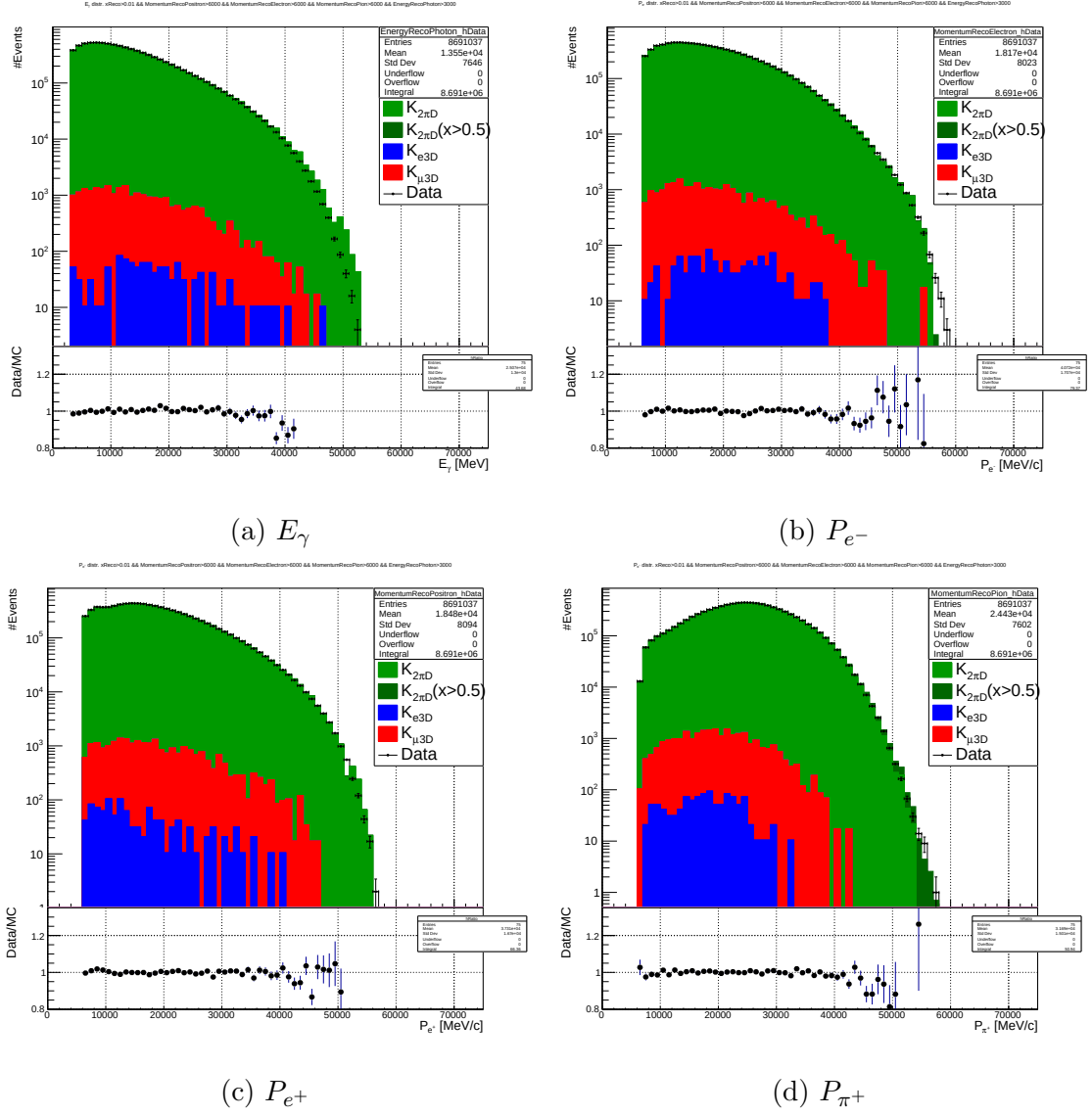


Figure 2.32: Data/MC comparison for the 2017+2018 data sample and the remaining MC samples after the cut on x and the inclusion of the biased signal sample. All the analysis cuts are applied.

of generated events in the $x < 0.5$ region in the $K_{2\pi D}$ standard MC, and the number of generated biased $K_{2\pi D}$ events. Indeed these quantities are affected by the modification of the tested slope. During the minimization procedure, we

1. filled a distribution with all the x_{true} values from the $K_{2\pi D}$ (standard) MC sample
2. filled a weighted histogram with all the x_{true} values from the $K_{2\pi D}$ (standard) MC sample but with each entry weighted using (2.30), with a_h the slope being tested.

The $N_{K_{2\pi D}, x < 0.5}$ new value is then given by the integral of the weighted distribution. We did the same with the biased sample to obtain the new value of $N_{K_{2\pi D}, x > 0.5}$. We then used the following relation

$$BR(\pi_D^0; a_h) = BR(\pi_D^0; a = 0.032) \frac{\int_{r^2}^1 \frac{d\Gamma(a_h)}{dx}}{\int_{r^2}^1 \frac{d\Gamma(a=0.032)}{dx}} = BR(\pi_D^0)_{PDG} * \frac{N_w}{N_{std}} \quad (2.57)$$

where N_w and N_{std} are the integrals over the whole x range of the weighted histogram and the reference one from the standard $K_{2\pi D}$ MC sample, while we computed the new $BR(\pi_D^0(x > 0.5))$ by normalizing to 1 the weighted histogram, and integrating from 0.5 to 1. The value we obtained with this procedure that is our final result for the TFF slope measurement is

$$a \pm \delta a_{stat} = (3.56 \pm 0.29) \times 10^{-2} \quad (2.58)$$

This value has a relative statistical uncertainty of 8.1%, the smallest statistical uncertainty obtained in the TFF measurements. Considering that

$$\delta a_{stat}^2 = \delta a_{MC}^2 + \delta a_{Data}^2 \quad (2.59)$$

we disentangled the δ_{MC} contribution from (2.58) by performing the measurement procedure after removing half of the MC sample. The two uncertainties δa_{stat}^2 and δa_{stat}^{*2} are then related by

$$\begin{cases} \delta a_{stat}^2 = \delta a_{MC}^2 + \delta a_{Data}^2 \\ \delta a_{stat}^{*2} = 2\delta a_{MC}^2 + \delta a_{Data}^2 \end{cases} \quad (2.60)$$

finally giving

$$\delta a_{MC}^2 = \delta a_{stat}^{*2} - \delta a_{stat}^2 \rightarrow \delta a_{MC} = 0.21 \quad (2.61)$$

and as a consequence

$$\delta a_{Data} = 0.2 \quad (2.62)$$

With an analog method we found the the contribution to the a_{MC} uncertainty is dominated by the standard $K_{2\pi D}$ sample.

2.6 Systematics

2.6.1 Trigger efficiency

As shown in section 2.4.8, the trigger efficiency of the data is reproduced quite well in MC using the trigger emulators. We assessed the effect of the possible imperfections

of the trigger emulation separately for each trigger logic condition in the trigger chain (2.38). We fitted with a second order polynomial

$$\varepsilon_i(x_{reco}) = p0 \cdot (1 + p1 \cdot x_{reco})^2, \quad i = \text{RICH, QX, LKr, KTAG, STRAW} \quad (2.63)$$

all the trends in Figure 2.28 of the trigger efficiencies as a function of x which might induce a spurious slope in the measurement, now including the biased sample with $x > 0.5$ for the MC efficiency, which helps in reducing the large statistic uncertainty at large values of x . We show the results in Figure 2.33 for all the trigger conditions. All the trigger conditions but the KTAG, exhibit different slopes in data and in MC.

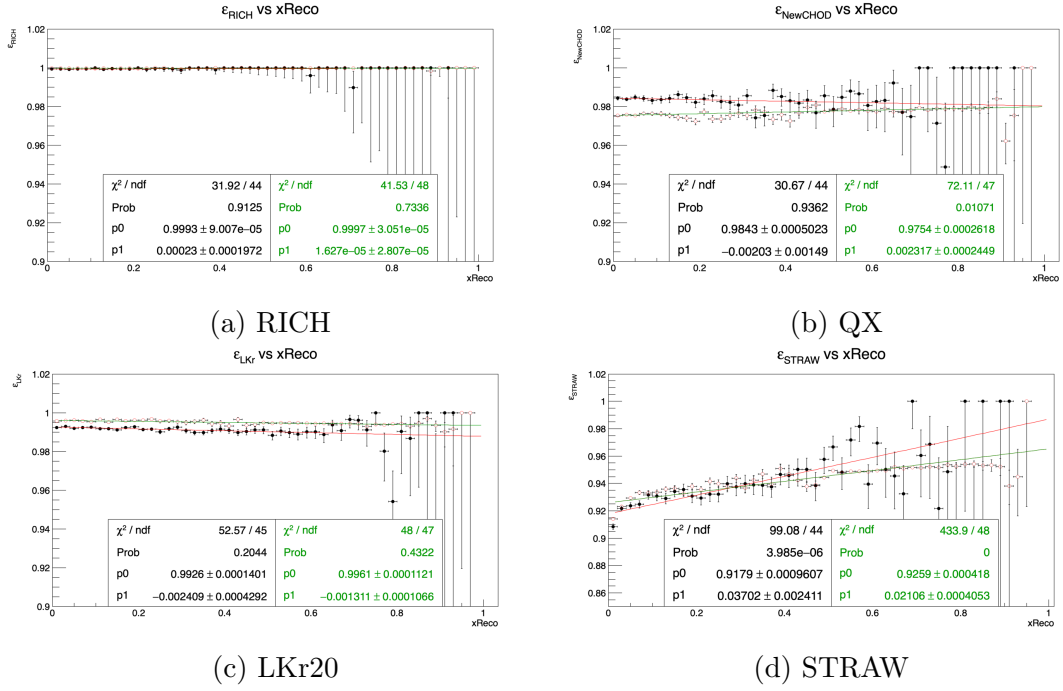


Figure 2.33: Data (black full dots) and MC (red empty points) trigger efficiency comparison for all the logic conditions of the trigger mask (2.38). We fitted with the function (2.63), representing an additional fake slope on the top of the real TFF one. The result on the MC sample is reported by the green function and the green box while the red line is the fit on the data.

To quantify this effect we developed an alternative trigger emulation based on the data trigger efficiency fit. We removed the global trigger emulator response from the selection of the MC events, replacing it with the logic *and* condition between the individual conditions. In this way we can keep the answer of all but one the primitive emulator, whose effect on the slope can be replaced by a re-weighting with the fitted $\varepsilon_i(x_{reco})$ from data. We report in Table 2.4 the result with the change in the TFF measurement. To be conservative we take the largest deviation of the slope, given by the STRAWexo trigger emulation

$$\Delta a = 1.25 \times 10^{-2} \quad (2.64)$$

where Δa is always the the difference between the reference value (2.58) and the observed maximum variation (in this case the one due to the STRAWexo emulation).

Logic condition	$(a_i \pm \delta a_i) \times 10^2$	$\Delta a \times 10^2$
RICH	3.48 ± 0.29	0.08
QX	3.96 ± 0.29	-0.4
LKr20	3.65 ± 0.29	-0.09
STRAWexo	2.31 ± 0.29	1.25

Table 2.4: Summary of the study on systematic effect related to the emulation of the trigger efficiency in MC. In the second column is shown the value of the fitted parameter and its uncertainty when the corresponding detector emulation is done with the re-weight technique described in the text. In the last column is computed the difference with respect to the reference value (2.58).

2.6.2 Spectrometer track corrections

At the analysis level the reconstructed momentum of each spectrometer track is corrected by two calibration factors (α, β) ([40], section STRAW). The relation between the corrected (P) and not corrected (P_0) momentum is

$$P = P_0(1 + \beta)(1 + q\alpha P_0) \quad (2.65)$$

where q is the charge of the track. Typical values for β are $O(10^{-3})$ while α is smaller $O(10^{-9})$ $[\text{MeV}/c]^{-1}$. During the automatized processing of a Run, first the β constant is computed, which represents the momentum scale of the spectrometer. It is determined by the shift of the reconstructed $K_{3\pi}$ invariant mass with respect to the K^+ mass from the PDG. Knowing β , the α computation is performed by fitting the $K_{3\pi}$ invariant mass versus the momentum of the negative track for different α hypothesis. Each value is compared to the PDG K^+ mass to build a χ^2 which is finally fitted as a function of α . The corrections are recomputed during each re-processing round. The corrections are computed also for MC samples, for both the reference Runs. While for data they are fundamental to take into account physical changes (e.g. the variations in the spectrometer magnetic field), in MC events they are used just to correct small variations in the STRAW track reconstruction software.

To evaluate the systematic effect we varied the value of the β and α corrections within its maximum and minimum value, repeating the whole analysis. We kept MC correction on. We find that the β correction has the most important impact causing a shift on the TFF slope of

$$\Delta a = 0.01 \times 10^{-2} \quad (2.66)$$

with respect to (2.58).

2.6.3 Accidental Vertices

We evaluated the effect of accidental events by adding to our sample out-of-time events from the timing regions

$$-30 \text{ ns} < T_{\text{vertex}} - T_{\text{trigger}} < -15 \text{ ns} \quad (2.67)$$

$$15 \text{ ns} < T_{\text{vertex}} - T_{\text{trigger}} < 30 \text{ ns} \quad (2.68)$$

Since we found ≈ 3000 events in those regions, in our selected time interval of 10 ns (see section 2.4.3) we expected ≈ 300 events. The change in the slope is

$$\Delta a = -0.01 \times 10^{-2} \quad (2.69)$$

2.6.4 Accidental Clusters

We performed a similar study for the accidental LKr activity. In this case the $T_{vertex} - T_{cluster}$ timing parameter was used to allow out-of-time events which are normally rejected in the standard analysis. We estimated ≈ 6509 events in the signal region and the fit procedure with the inclusion of the accidental clusters generates a shift in the TFF slope of

$$\Delta a = -0.24 \times 10^{-2} \quad (2.70)$$

2.6.5 LKr cluster fine calibration

The LKr cluster fine calibration introduced small differences in the event selection, in particular by affecting the acceptance. This effect is generated by the fact that in general the energy correction based on π_D^0 is a little bit larger than the one based on the π^0 mass one (see Chapter 3 for a discussion of the LKr energy calibration procedures) for almost all energy values, and since the corrected energy is obtained as

$$E_{corr} = \frac{E_{uncorrected}}{R(E_{uncorrected})}, \quad (2.71)$$

where R is the calibration factor, the cut on E_{corr}^{min} increases the sample when the π^0 calibration is used. By performing the whole procedure from scratch with the π^0 calibration we found a change of the TFF slope

$$\Delta a = 0.02 \times 10^{-2} \quad (2.72)$$

2.6.6 Kinematic Particle identification

To study the effect of possible mis-reconstruction due to the kinematic particle identification described in section 2.4.5, we relaxed the cut on the maximum number of allowed vertex-cluster combinations that are included in the elliptical signal region in the $M_{K^+}^{reco}$ vs $M_{\pi^0}^{reco}$ plane. Considering that at the end of the selection one of the two combinations must be in any case selected, we opted for taking randomly one of the two. The fit performed with the addition of these events ≈ 3000 produces a shift in the TFF slope by

$$\Delta a = -0.01 \times 10^{-2} \quad (2.73)$$

2.6.7 Neglected MC sources

We considered eight background sources in our analysis (see Table 2.1) accounting for $\approx 36\%$ of the K^+ branching ratio. The remaining decays mostly include $K^+ \rightarrow \mu^+ \nu$ events $BR \approx 64\%$, which we did not process since they are single track events, strongly suppressed by our selection. Other possible background sources are the processes $K^+ \rightarrow \pi^+ e^+ e^-$, $K^+ \rightarrow \pi^+ \mu^+ \mu^-$, $K^+ \rightarrow \pi^+ \pi_D^0 \pi_D^0$, $K^+ \rightarrow \pi^+ \pi^0$, $\pi^0 \rightarrow e^+ e^- e^+ e^-$ (double Dalitz decay) which have negligible branching ratios (maximum

$O(10^{-6})$). So the systematic due to neglected MC sources could not be greater than the one of missing the $K_{\mu 3D}$ decay. To be conservative we remove all the backgrounds from the MC sample in the measurement procedure. We do not find any variation in the final result. This could be explained by the fact that the background events concentrate at small x values, which is not sensible to the slope.

2.6.8 Overlay treatment

We described in section 1.5.1 the overlaid samples, largely used in the analysis. The use of overlaid samples requires a specific treatment to accurately match MC reconstructed events to the generated ones. We describe in appendix A.1 the algorithm used to perform the association. However, at the end of the selection procedure, 42 reconstructed events didn't match any generated event. We looked in detail at those events and we found that the matching requirement fails due to a single track being not associated out of the three used. Moreover the two associated tracks were successfully associated to $K_{2\pi D}$ events. So we decide to included those events in the fit as pertaining to the $K_{2\pi D}$ sample. The fitted value of the TFF slope changed by

$$\Delta a = 0.16 \times 10^{-2} \quad (2.74)$$

2.6.9 Stability Checks

In the following sections we report the stability checks on the TFF measurement. Since the repetition of the measurement with the change of only one conditions lead to performing the procedure on correlated samples, we show in each plot the *full* statistical uncertainty, that is the uncertainty from the fit procedure δa_{fit} , and the *uncorrelated* statistical uncertainty, which defined as

$$\delta a_{un} = \sqrt{|\delta a_{fit}^2 - \delta a_{ref}^2|} \quad (2.75)$$

where δa_{ref} is the measurement uncertainty performed in the reference condition in the specific stability check. The need for the absolute value in (2.75) is because the difference in the square root can be either positive or negative according to the size of the considered samples. The (2.75) is a reasonably good estimation of uncorrelated uncertainties between two results if the variation of the condition is performed on the same dataset, causing the varied sample to be a sub-sample of the reference one or the other way around.

Number of bins

We repeated the measurement by changing the number of equipopulated bins. This study is related to the observation that the statistical uncertainty on the fit decreases when increasing the number of bins if they are too few (with respect to the resolution). As can be seen in Figure 2.34, at $N_{bins} = 50$, the central value of the fit stabilizes and the uncertainty stops to decrease.

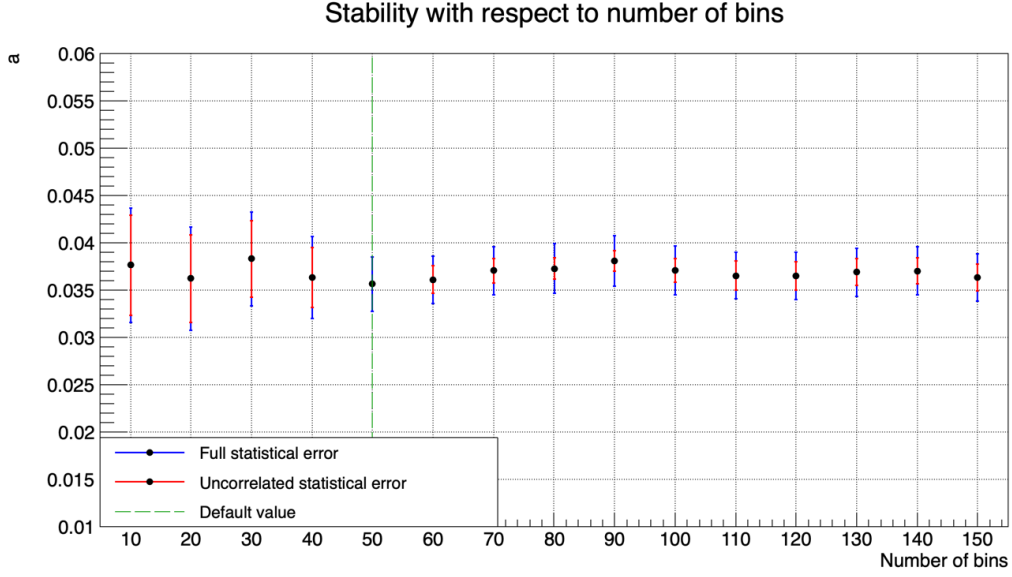


Figure 2.34: Stability of the fit result varying the number of bins used to perform the χ^2 test.

Time stability

We performed the measurement by subdividing the data sample into the different data-taking periods (Figure 2.35). Since we kept the MC sample unchanged, we used $\delta a_{ref} = 0.21$ (see 2.61).

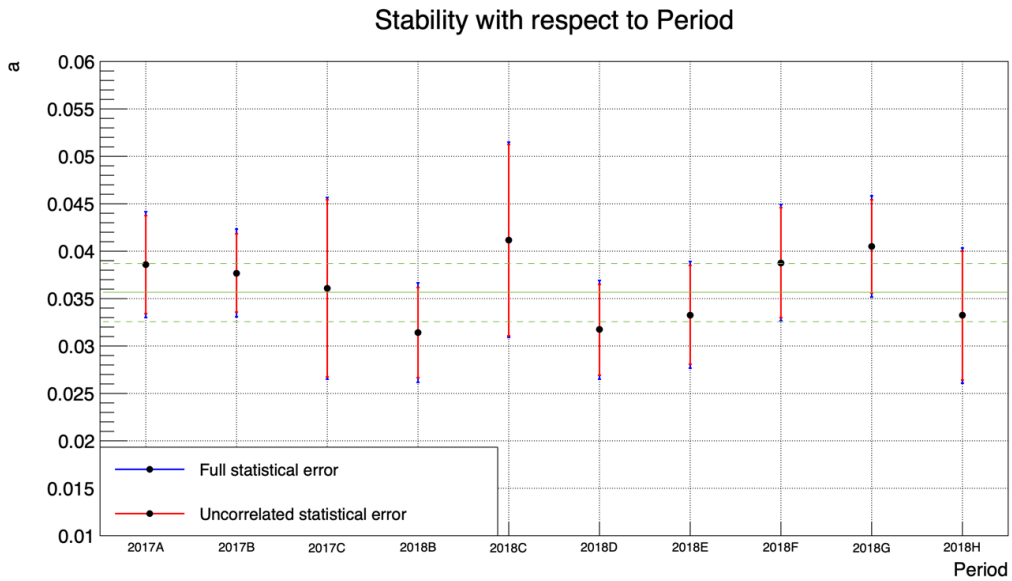


Figure 2.35: Stability of the measurement in different data-taking periods. The MC sample is kept constant in the study. In green is shown the value (2.58).

Stability with respect to selection cuts

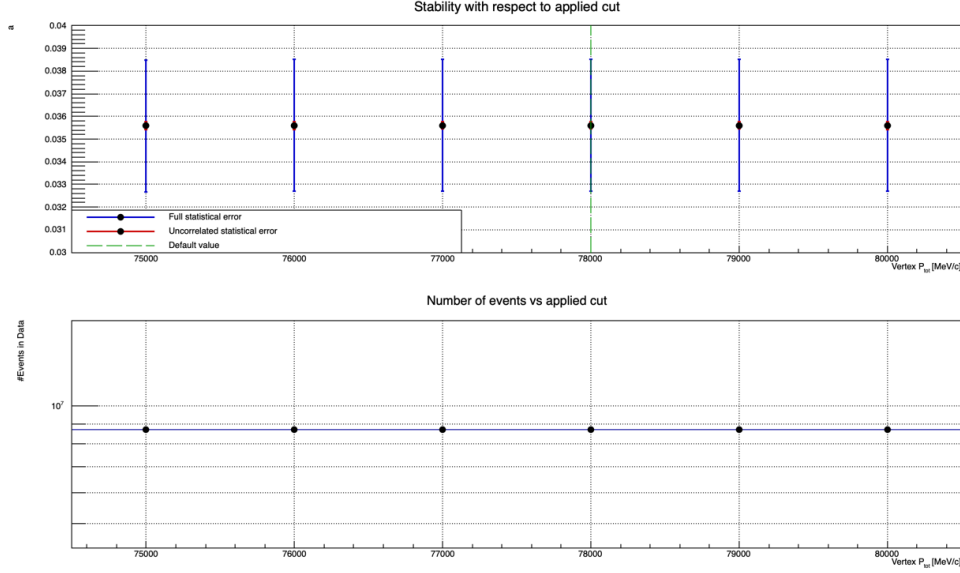


Figure 2.36: Stability with respect to the total vertex momentum. The nominal value of the cut is shown as a reference. All the measurements are statistically compatible.

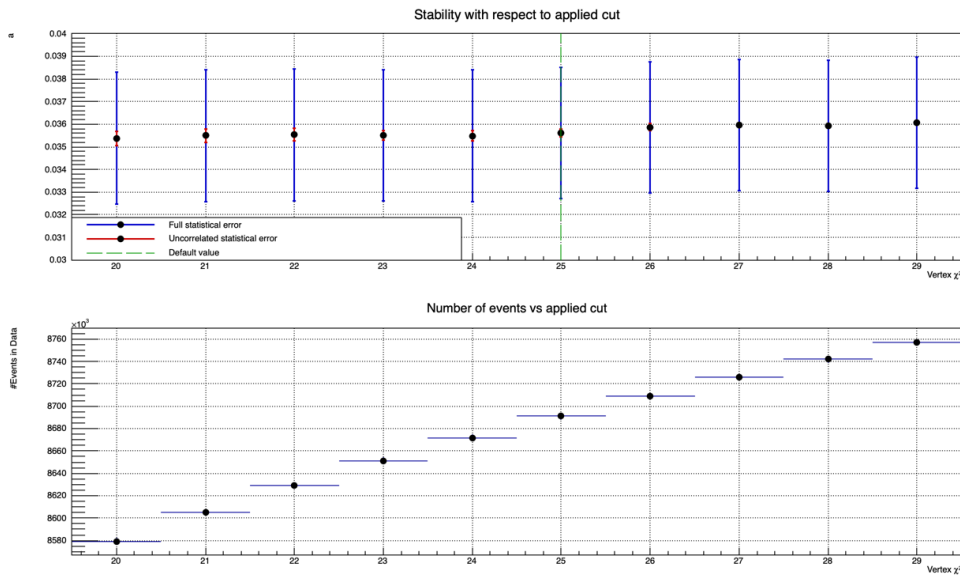


Figure 2.37: Stability with respect to the χ^2 of the vertex. The fit is stable until the reference value of $\chi^2 < 25$, which justifies our choice of the cut value.

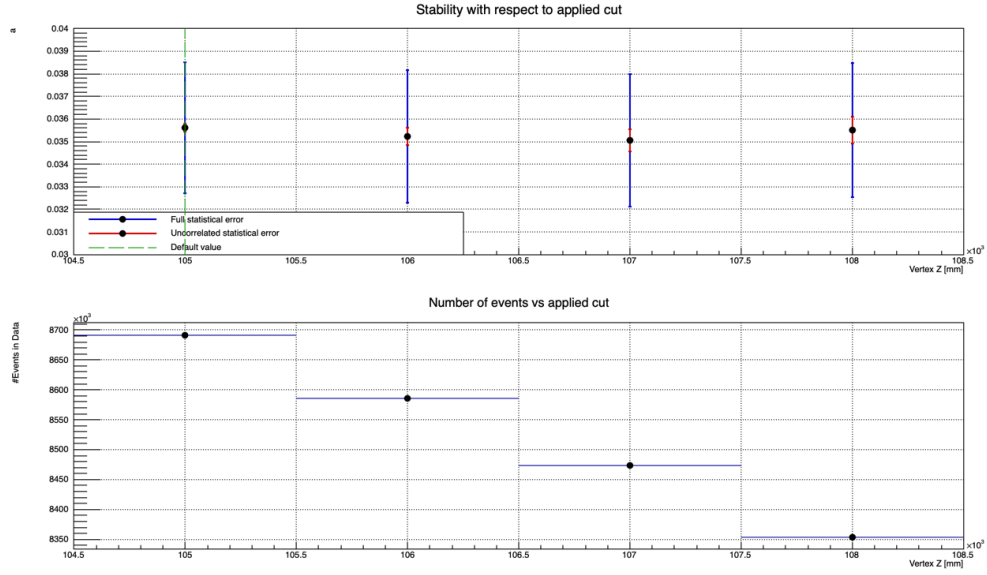


Figure 2.38: Stability check with respect to the minimum value of the Z coordinate of the reconstructed vertex. In this case we can only increase the cut with respect to the nominal value 105 mm is the minimum cut on the fiducial decay region. We did not perform the study for the maximum value of the Z coordinate since the cut is so wide that no events are present close to the cut value.

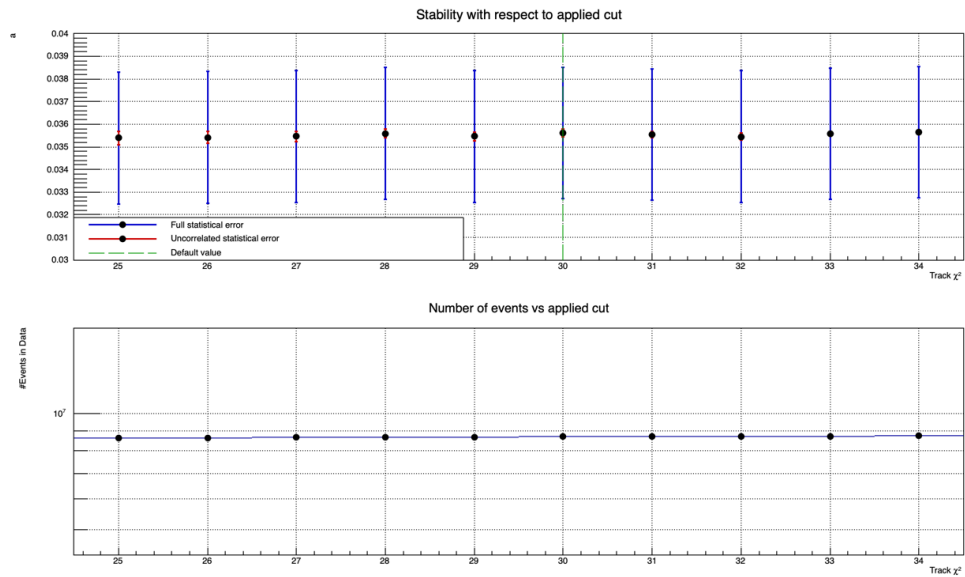


Figure 2.39: Stability of the measurement with respect to the χ^2 of each track. The cut value is the same for all the three tracks of the vertex.

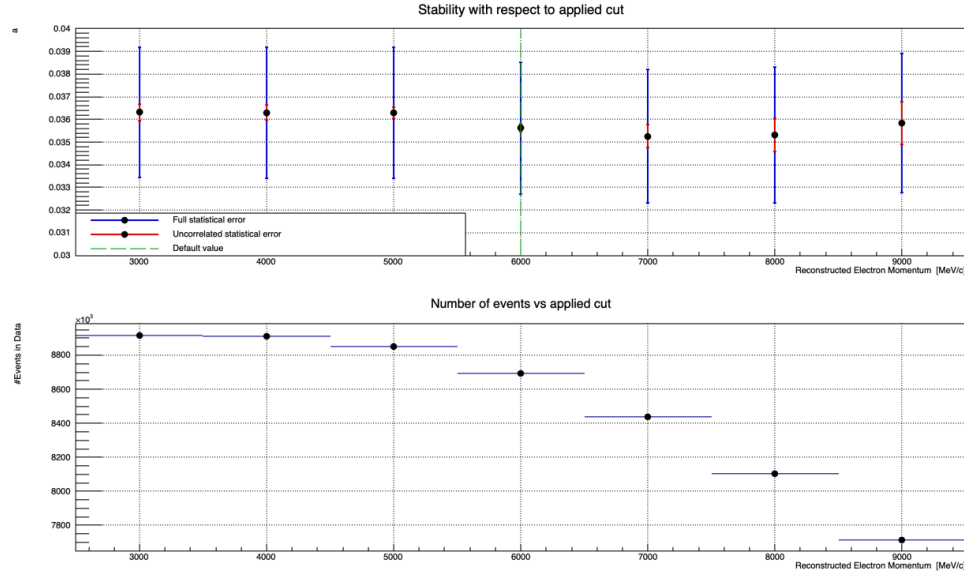


Figure 2.40: Stability with respect to the momentum of the reconstructed e^- . The instability at small values of the momenta is due to imperfection in the trigger emulation and to the validity of the track corrections.

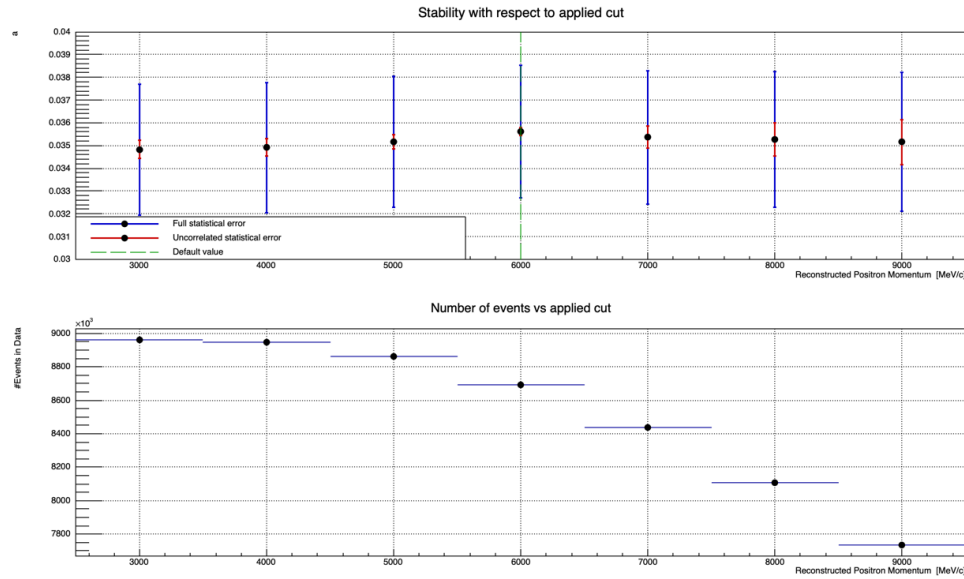


Figure 2.41: Stability with respect to the momentum of the reconstructed e^+ . The instability at small values of the momenta is due to difficulties in the trigger emulation and the track corrections.

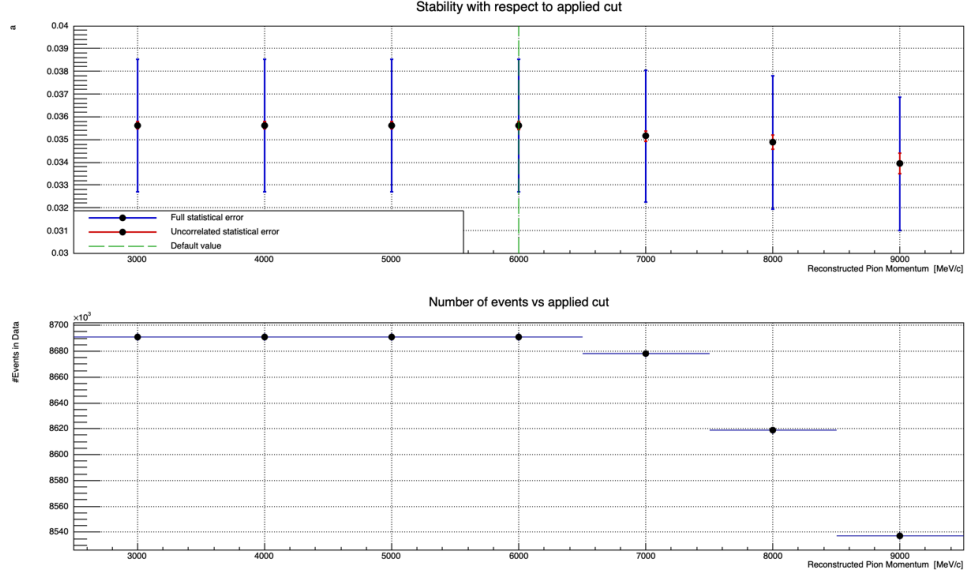


Figure 2.42: Stability with respect to the momentum of the reconstructed π^+ . The instability at large values is related to the fact that higher values of P_{π^+} correspond to low values of e^+ and e^- momenta, leading to worse reconstruction.

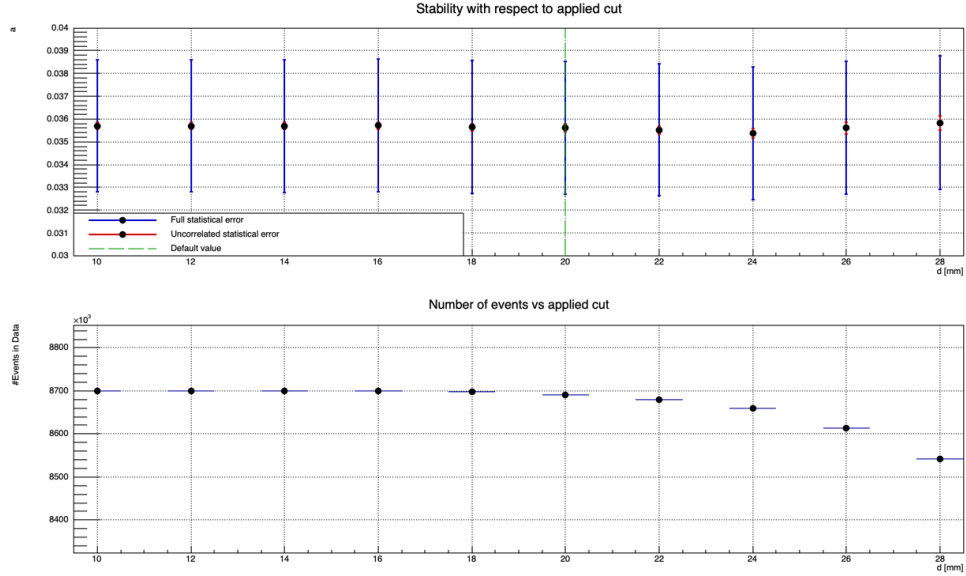


Figure 2.43: Stability with respect to the minimum separation of the tracks at the STRAW chamber 1. In case the minimum distance is the one between the e^+ and e^- tracks, this quantity is highly correlated to the x variable (related to the opening angles of the tracks). Large values of this cut correspond to large values of x , which are more sensible to e.g. to the trigger efficiency emulation. This explains the small shift at large values of the separation.

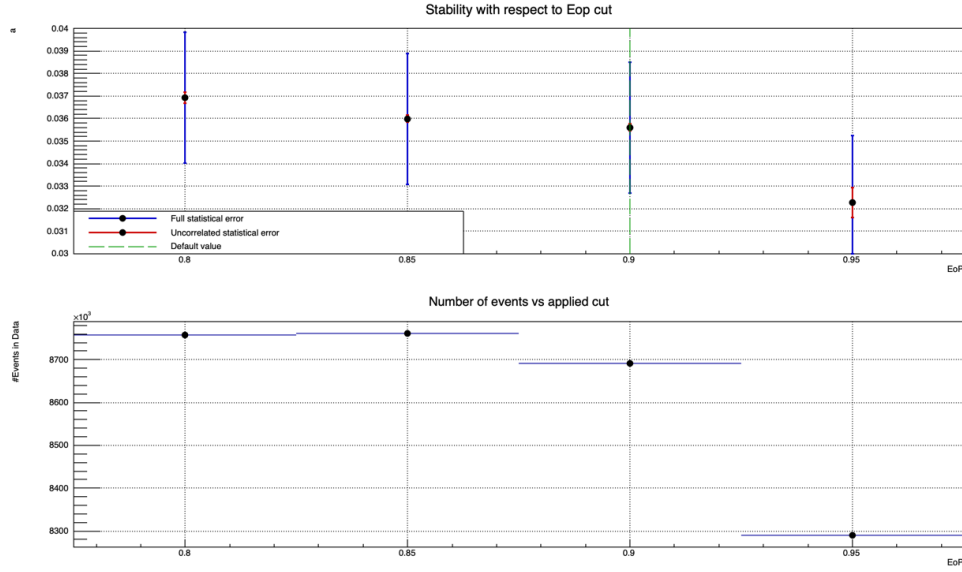


Figure 2.44: Stability of the measurement with respect to the E/p cut. The cut was changed on a wider range than usually applied in the analyses to justify our choice, which is a compromise between the expected value for the E/p for e^+ , e^- , π^+ and the resolution on this quantity.

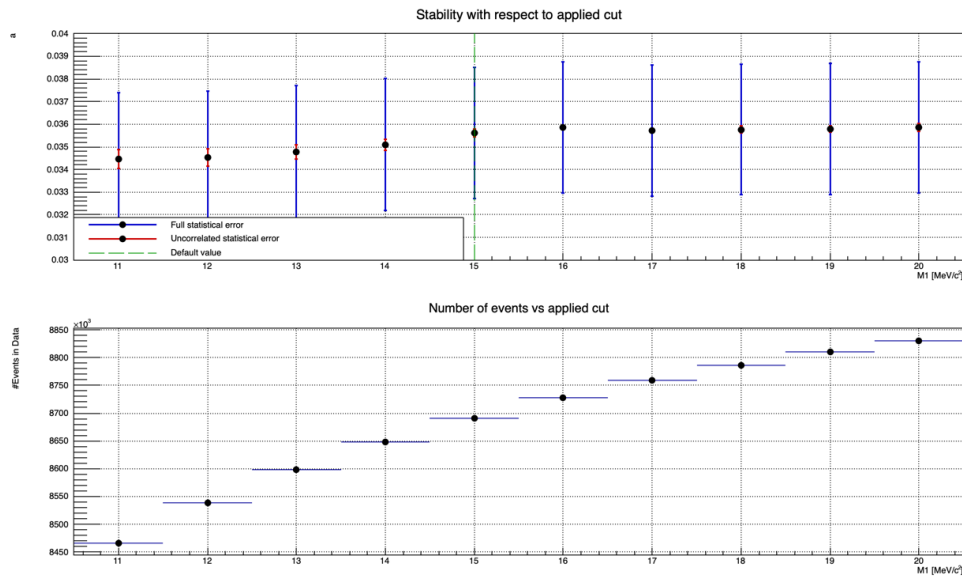


Figure 2.45: Stability with respect to the $M1$ mass defined in (2.42). Our cut is justified considering that cutting at lower values exhibit instabilities of the fit.

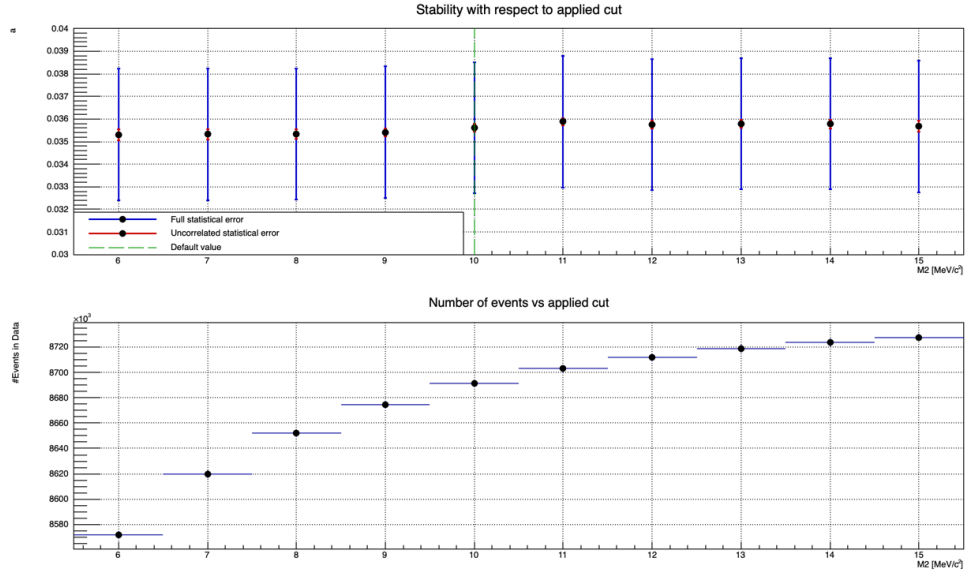


Figure 2.46: Stability with respect to the $M2$ mass defined in (2.42). The stability of the fit is better than the $M1$ case since the $M2$ distribution is more symmetric (Figure 2.8).

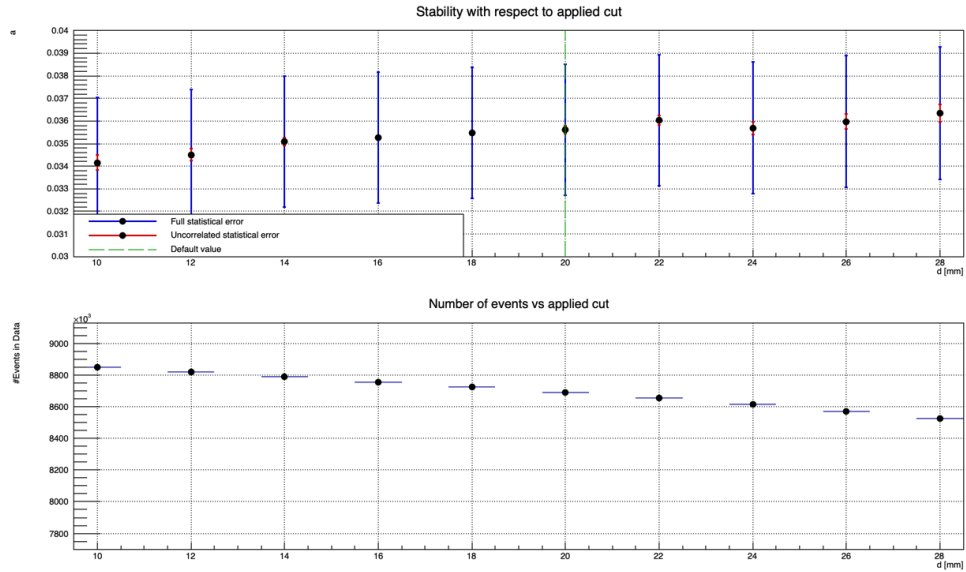


Figure 2.47: Stability with respect to the distance of the cluster to the closest "dead" cell of the LKr. Small values of the cut correspond to clusters whose energy has been reconstructed only partially, leading to mis-reconstruction of the photon energy.

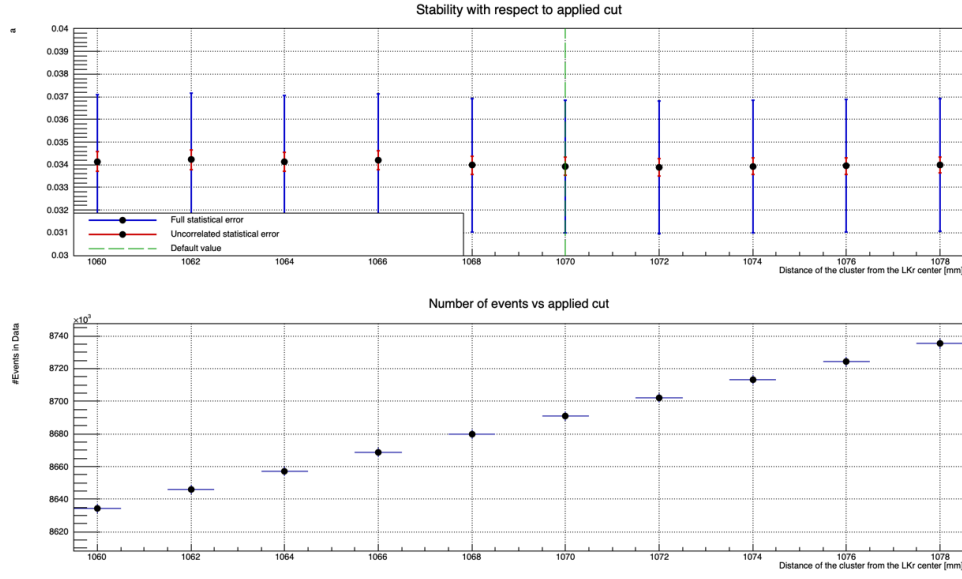


Figure 2.48: Stability with respect to the maximum distance of the cluster with respect to the center of the LKr. The 1070 mm cut corresponds to the "shadow" of the last LAV12 station. Clusters in this LKr region could originate from showers from e^+e^- conversions.

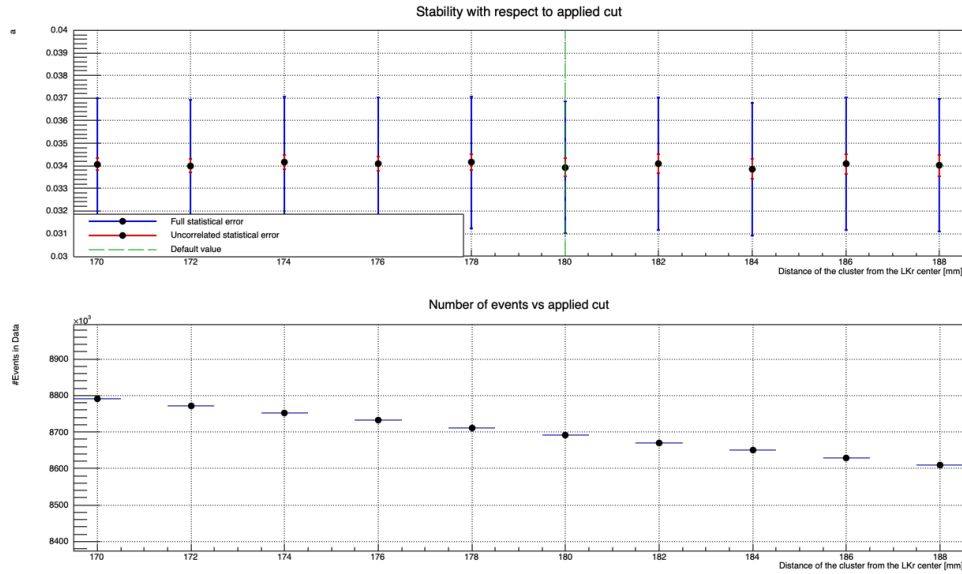


Figure 2.49: Stability with respect to the minimum distance of the cluster with respect to the center of the LKr. The 180 mm cut is done to avoid badly-reconstructed photons close to the beam pipe.

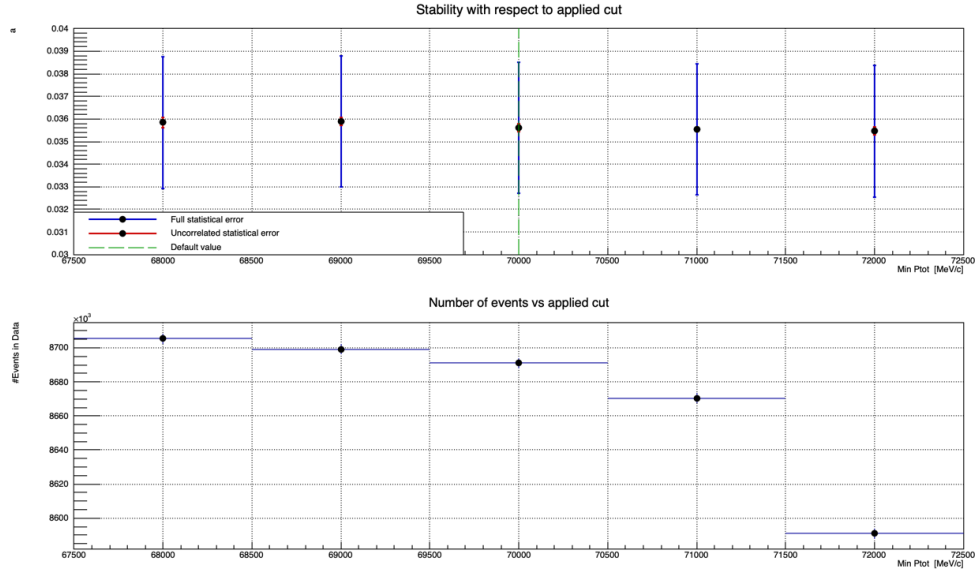


Figure 2.50: Stability with respect to the cut on the minimum total reconstructed momentum of the event. No instabilities are seen.

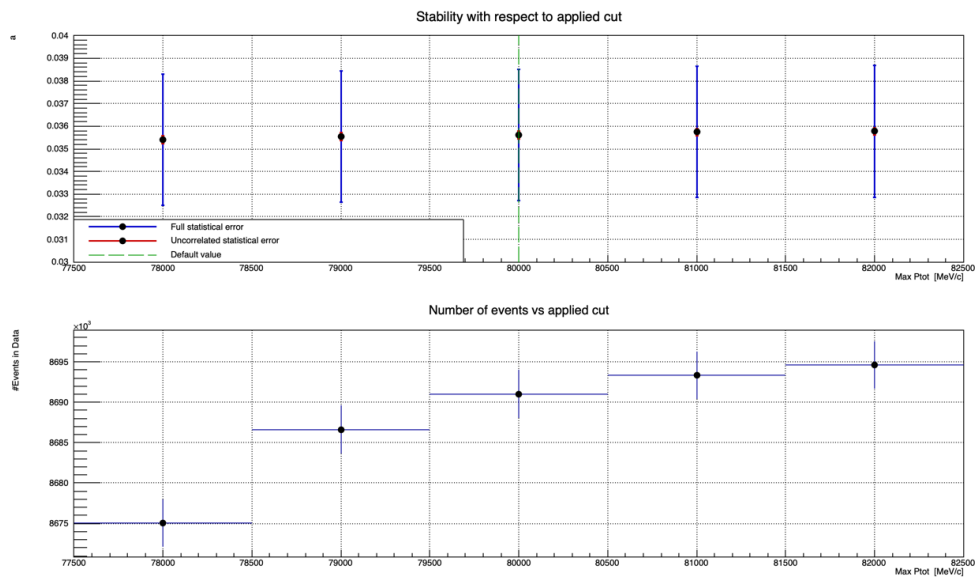


Figure 2.51: Stability with respect to the cut on the maximum total reconstructed momentum of the event. No instabilities are seen.

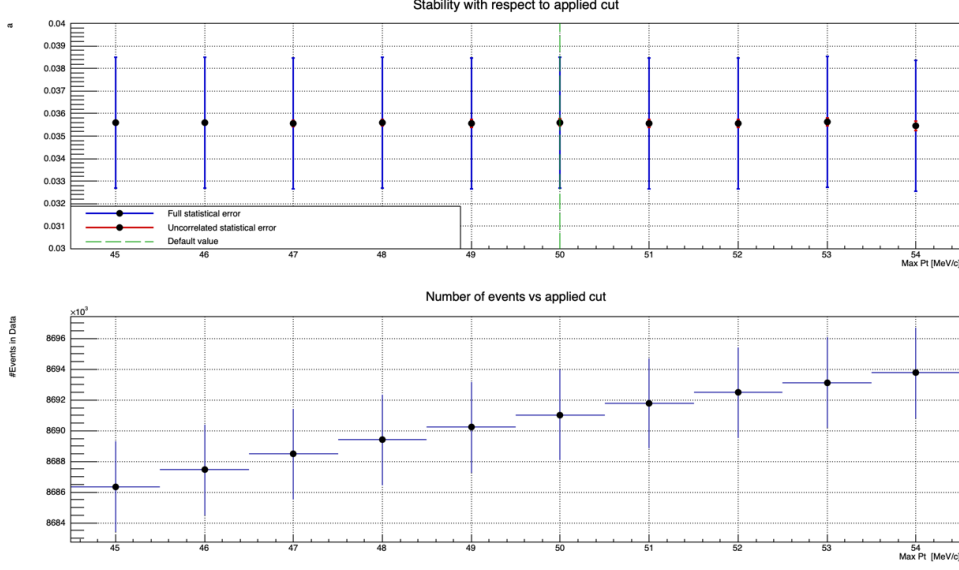


Figure 2.52: Stability with respect to the cut on the maximum total reconstructed transverse momentum of the event. No instabilities are seen.

2.7 Conclusions

A summary of all the uncertainty contributions to the TFF measured value are reported in Table 2.5. The largest contribution to the uncertainty arises from the systematic effect on the trigger efficiency emulators. Summing up all the systematic uncertainties in quadrature we got

$$a \pm \delta a_{stat} \pm \delta a_{sys} = (3.56 \pm 0.29 \pm 1.35) \times 10^{-2} \quad (2.76)$$

and finally

$$a \pm \delta a = (3.56 \pm 1.38) \times 10^{-2} \quad (2.77)$$

2.8 The $BR(\pi^0 \rightarrow e^+e^-\gamma)$ measurement

Together with the study of the TFF, the measurement of the $BR(\pi_D^0)$ decay has also been considered, exploiting also in this case the main decay channel producing π^0 , i.e. $K_{2\pi}$. The plan is to perform the measurement using $K_{2\pi}$ decays, with $\pi^0 \rightarrow \gamma\gamma$, as the normalization decay channel, to be independent from the total number of decayed K^+ , according to the relation

$$\frac{BR(\pi_D^0)}{BR(\pi_{\gamma\gamma}^0)} = \frac{BR(K_{2\pi D})}{BR(K_{2\pi\gamma\gamma})} \frac{N_S - N_{Bs}}{N_N - N_{Bn}} \frac{A_N}{A_S} \frac{\varepsilon_N}{\varepsilon_S} \frac{D_S}{D_N} \frac{D_{fs}}{D_{fn}} \quad (2.78)$$

Here the meaning of each quantity:

- N_S, N_N : number of events selected by the signal selection and normalization selection respectively from the data sample
- N_{Bs}, N_{Bn} : estimated background events within the selected N_S and N_N events

Source	$\Delta a \times 10^2$
Total statistical	0.29
Data	0.20
MC	0.21
Systematics	
RICH trigger condition emulation	0.08
QX trigger condition emulation	-0.4
LKr20 trigger condition emulation	-0.09
STRAWexo trigger condition emulation	1.25
Spectrometer track corrections	0.01
Accidental vertices	-0.01
Accidental Clusters	-0.24
LKr Fine calibration	0.02
Kinematic PID	-0.01
Neglected MC sources	-
Overlay treatment	0.16

Table 2.5: Summary of the study of systematic uncertainties.

- A_N, A_S : acceptances for the signal and the normalization sample, estimated using MC samples
- $\varepsilon_S, \varepsilon_N$: measured trigger efficiencies from data for the trigger mask used to select signal and normalization events
- D_S, D_N , downscaling factors applied to the corresponding trigger masks
- D_{fs}, D_{fn} , downscaling factors of the filter algorithm used to select the signal and normalization samples

2.9 Event filtering

For what concern the signal selection, we decided to reuse the one developed for the TFF measurement, showing a good background suppression. In the case of the normalization selection a standard selection was available, based on PID given by detectors (RICH rings and E/p by LKr) and veto conditions. However in order to avoid as far as possible any asymmetries between the Dalitz and the normalization selection, we preferred to develop a new selection making use of kinematic cuts for the PID and no veto conditions. Moreover the search for the most unbiased sample on which to tune the selection lead to the modification of the available NA62 filtered data sample. Indeed all the samples starting from which a $K^+ \rightarrow \pi^+\pi^0$ could be selected were biased by the use of veto conditions given by the LAV detector, which could lead to asymmetries with respect to the Dalitz selection on which vetoes were (and are not) applied since all the particles are detectable. The criteria of the new filter algorithm were tuned from scratch, and this required to fit not only our analysis, but also requirements of other physics analysis which, due to data storage space constraints, would have to use the new filtered stream. For this reason, the collaboration approval was required before starting the data reprocessing with the

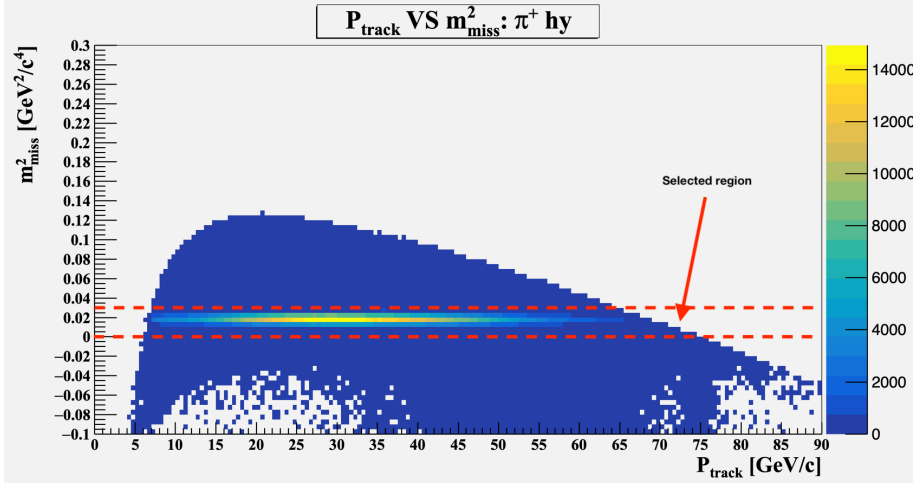


Figure 2.53: m_{miss}^2 vs P_{track} for simulated $K_{2\pi}$ samples. The region identified by the filter is highlighted.

new filtering code. The basic requirement of the new filter is that the event must have at least one track with "good" quality (acceptable χ^2 and momentum, in acceptance of all downstream detector) and two isolated, i.e. not associated to any track, LKr clusters in time with it. Moreover, $m_{miss}^2 = (p_{K^+} - p_{\pi^+})^2$ with P_{K^+} the average beam momentum should be in the $K_{2\pi}$ region $0 < m_{miss}^2 < 0.03 \text{ GeV}^2/c^4$, shown in Figure 2.53. As explained before, the need to fit also the goals of other analyses, in particular the search for rare decays, required the definition of another region to be accepted by the filter, namely $m_{miss}^2 < 0.01, m_{miss}^2 > 0.03 \text{ GeV}^2/c^4$ (called "rare" regions). Due to the available storage resources, a downscaling factor of 10 was chosen for accepting the events in the $K_{2\pi}$ region, while no downscaling was imposed in the "rare" region given the low statistics of the decay channels of interest. Unfortunately, in its first implementation, the filter algorithm was implemented as a unique stream, accepting events without any information about which decay region the event belongs to. Without this information it is impossible to know the correct downscaling factor by which the data should be re-scaled according to (2.78). A new code is now available which splits the filter stream into two different sub-streams. Since this problem was discovered only after the end of the reprocessing round, there is the need to wait for the next processing round to have the samples available (O(2) months).

Preliminary result

While waiting for a new reprocessing to be completed, the data used to get the preliminary result are from the control filtered samples, requiring only that events have the control trigger mask (i.e. a particle interacted with the CHOD detector), which is strongly downscaled because of the minimum bias requirement, plus a further downscaling of 25 applied at the filtering step. Starting from the filtered sample, the normalization selection applies only stricter cuts on the selected track, requiring also an $E/p < 0.9$ and no MUV3 clusters associated to the track in a time window of 2.5 ns. Those cuts reduce strongly the $K_{\mu 3}$ and $K_{e 3}$ backgrounds, which could mimic a $K_{2\pi}$ if the π^+ is mis-reconstructed. Indeed the E/p should act as for the $K_{2\pi D}$ selection (see section 2.4.7) while the MUV3 detector, part of

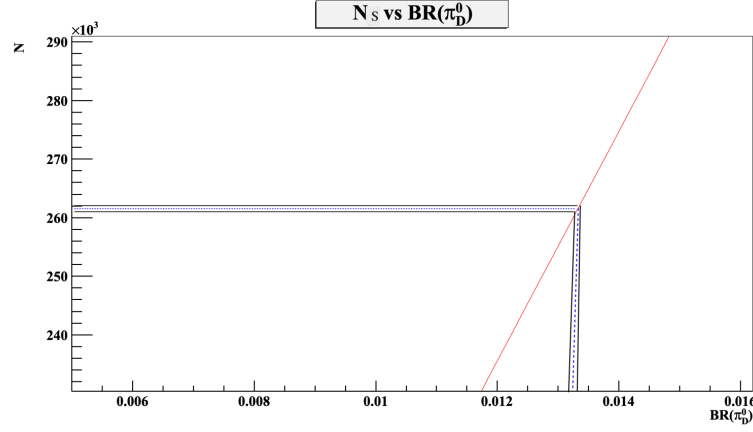


Figure 2.54: Graphic method for $BR(\pi_D^0)$ computation. The red line indicates the expected value of N_S , computed rearranging (2.78), according to the input value $BR(\pi_D^0)$. Given a measured value of N_S , the corresponding abscissa is the result of the procedure. The uncertainty is obtained by taking the abscissa values of $N_S \pm \sqrt{N_S}$.

the veto system (see section 1.2.2) identify in-time events containing a muon. The remaining background fraction is $\sim 2\%$, considering the PDG branching ratio of the background modes used in the MC. The computation of $BR(\pi_D^0)$ according to (2.78) can be performed by a comparison of the expected number of N_S and the collected one, as a function of the Dalitz BR . In Figure 2.54 is shown a first try of implementation of such procedure. The preliminary value is

$$BR(\pi_D^0) = 0.01333 \pm 0.00003_{stat}$$

The result is incompatible within the present PDG value with the quoted uncertainty which is ~ 1.2 times smaller, but besides the need for further cross-checks, it should be noted that no systematic uncertainties are included since, as mentioned, this is not the preferred way in which the measurement will be performed.

2.9.1 Prospects

Another possible approach to be used for the computation of $BR(\pi_D^0)$ is to perform a model-dependent measurement of

$$R = \frac{BR(\pi_D^0)}{BR(\pi_{\gamma\gamma}^0)} \quad (2.79)$$

including in the new world average the value of our TFF measurement, once the systematic uncertainties will be reduced. Indeed from R one has

$$BR(\pi_{\gamma\gamma}^0) \simeq \frac{1 - BR(\pi_{DD}^0)}{1 + R} \quad (2.80)$$

$$BR(\pi_{D(\gamma)}^0) \simeq \frac{R}{R + 1} [1 - BR(\pi_{DD}^0)] \quad (2.81)$$

$$(2.82)$$

where $BR(\pi_{DD}^0)$ is the branching ratio of the double Dalitz decay $BR(\pi_D^0 D) = 3.3 \times 10^{-5}$ [45]. The validity of the previous equation relies on the fact that

$$1 - \pi_{DD}^0 \simeq BR(\pi_{\gamma\gamma}^0) + BR(\pi_{D(\gamma)}^0) \quad (2.83)$$

being the branching ratios of other decay modes smaller than 10^{-6} . This was the method adopted by the authors of [29], which reduces the uncertainty on R by two order of magnitudes with the world average including the uncertainty of [36], so we expect to decrease it even more.

Chapter 3

LKr energy fine calibration based on $\pi^0 \rightarrow e^+e^-\gamma$ decay

3.1 Introduction

A precise measurement of the LKr cluster energies is crucial for many analyses. The general approach for the evaluation of the calibration factor (R) is to compare the reconstructed value of a certain quantity to its expected one. Different calibration techniques rely on different physical quantities used to obtain R . We found that, in the available calibrations, biases are present due to the assumptions on which they are built on, and we introduced a calibration based on the reconstruction of the photon of the $\pi^0 \rightarrow e^+e^-\gamma$ decays to try to solve some of them. We also considered improvements on the processing side, such as the possible reduction of the time required for the calibration step in the processing of a single Run.

In the first section we describe the general computation of the LKr fine calibration R and we show the weak points of the present calibrations. From the second section we go in detail in the description of the new technique, describing the event selection (third section), the R fitting procedure (fourth section) and the comparison between our technique and the other ones (fifth section). In the last section some feedback from analyses implementing the new calibration are shown.

3.2 Calibration computation and motivation for a π_D^0 based calibration

R is the ratio of the cluster energy returned by the (uncalibrated) reconstruction to the true cluster energy. In general, the value of R is not constant and depends on the energy. Thus the general procedure to extract the trend $R(E_{reco})$ is to run over a reconstructed data sample, computing the $2D$ distribution of R vs E_{reco} , where $R(E_{reco})$ value depends on the adopted technique. After filling the $2D$ distribution, this is sliced and averaged in energy bins to get the experimental trend of $R(E_{reco})$. In the NA62Analysis framework multiple LKr fine calibrations are already available. Generally the so called E/p or $1D - \pi^0$ mass calibrations are used in the analyses. We found that both present some weak points. The E/p calibration uses as E_{reco} the value of the energy of the cluster associated to a spectrometer track, reconstructed as a positron. Based on the assumption that positrons deposit their whole energy (E) in

the LKr, and considering that their masses are negligible compared to their momenta (p), the ratio E/p should be ≈ 1 . This assumption doesn't take into account the fact that e^+ can lose energy during their propagation from the spectrometer to the LKr, for example due to interactions with the detector material, leading to a bias towards low values of E_{reco} . To solve this problem, the $1D - \pi^0$ mass based calibration was developed. Here we summarize the main idea behind it to show where a π_D^0 based calibration can improve results. The $1D - \pi^0$ mass calibration is based on the kinematic reconstruction of the π^0 mass from $K^+ \rightarrow \pi^+ \pi^0$ decays, using the distance of the two clusters at LKr ($D_{\gamma\gamma}$), the distance between the reconstructed vertex and the LKr position ($Z_{LKr} - Z_{vtx}$) and the energies of the two clusters (E_1, E_2) through the following relation:

$$(m_{\pi^0})^2 = \frac{D_{\gamma\gamma}^2}{(Z_{LKr} - Z_{vtx})^2} (E_1 \cdot E_2) \quad (3.1)$$

Comparing the obtained value of m_{π^0} to the PDG one, it's possible to extract the value of the correction to be applied to the energies. Given the structure of equation (3.1), the computation of the correction is divided into two steps: the first one generates an approximation of the correction using "symmetric" events, where only reconstructed photons with $|E_1^{reco} - E_2^{reco}| < 1$ GeV are used. For these events, $E_{reco} = 0.5(E_1^{reco} + E_2^{reco})$ can be considered the energy of the event, and equation (3.1) gives the correction

$$R(E_{reco}) = \frac{m_{\pi^0}}{m_{\pi_{PDG}^0}} = \frac{1}{m_{\pi_{PDG}^0}} \frac{D_{\gamma\gamma}}{(Z_{LKr} - Z_{vtx})} \sqrt{E_1 \cdot E_2} \quad (3.2)$$

The second step, called "asymmetric", makes use of the $R(E_{reco})$, evaluated using the symmetric step, to correct the value of only one of E_1^{reco} or E_2^{reco} (the one in the validity range of $R(E_{reco})$). Trusting the correction for one cluster, equation (3.1) can be simplified to extract the correction for the other one. We found three main weak points of this approach:

- The selection of events exploits LKr clusters so it implicitly assumes that the value of the correction is small
- The correction of $E_{1(2)}$ relies on correctness of $E_{2(1)}$, generating oscillations of R around the correct value and compensation effects
- The asymmetric step is an iterative procedure since with each step the value of the correction improves, approximating the "true" one. Generally the "asymmetric" step is run three times in a processing round, each time requiring to fill and slice the $2D$ distribution R vs E_{reco} , so the $1D - \pi^0$ mass calibration requires 4 steps in total to be executed. Moreover, the event selection is run on reconstructed files (not filtered) increasing the running required time.

With a π_D^0 based calibration, with π^0 s taken from $K^+ \rightarrow \pi^+ \pi^0$ decays, these problems can be solved thanks to the final state of this decay channel, which has only one photon. Indeed,

- the selection of events can be made completely independent from LKr clusters, without any assumption on the strength of the correction

- evaluation of $R(E_{reco})$ is very simple and given by the ratio between the reconstructed energy of the photon (E_{reco}) and the expected one (E_{exp}), actually eliminating the problem of compensation
- the computation can be implemented in a single step and can be run on filtered (*Restricted three track vertex*, see section 2.4.1) events reducing by a factor more than 4 the processing execution time

3.3 π_D^0 selection

The selection of π_D^0 to be used for the R evaluation is a three-track selection similar to the one described in section 2.4, but exploiting information only from STRAW, KTAG, GTK. The first step is the isolation of events with at least a reconstructed vertex satisfying the following condition

- $P_{vtx} < 78 \text{ GeV}/c$
- $\chi_{vtx}^2 < 25$
- $105 \text{ m} < Z_{vtx} < 180 \text{ m}$
- all the vertex tracks in acceptance of all the chambers of the spectrometer
- all the vertex tracks with $5 \text{ GeV}/c < p < 60 \text{ GeV}/c$
- all the vertex tracks with $\chi^2 < 30$
- minimum distance of the vertex tracks at STRAW chamber 1 $\geq 20 \text{ mm}$

Each selected vertex is paired with the closest in time KTAG candidate (with ≥ 5 sectors). The KTAG candidate is used to compute the closest in time GTK candidate (within 0.5 ns).

Defining

$$P_K \equiv (\vec{p}_{beam}, \sqrt{\vec{p}_{beam}^2 + m_K^2}) \quad (3.3)$$

and $\vec{p}_{neg}, \vec{p}_{pos1}, \vec{p}_{pos2}$ the negative and the two positive reconstructed three-momenta of the tracks of each selected vertex, we compute the two possible missing masses

$$m_{miss1}^2 = (P_K - P_{\pi+1})^2, P_{\pi+1} \equiv (\vec{p}_{pos1}, E = \sqrt{\vec{p}_{pos1}^2 + m_{\pi+}^2}) \quad (3.4)$$

$$m_{miss2}^2 = (P_K - P_{\pi+2})^2, P_{\pi+2} \equiv (\vec{p}_{pos2}, E = \sqrt{\vec{p}_{pos2}^2 + m_{\pi+}^2}) \quad (3.5)$$

and we only select vertices where at least one missing mass is in the range $|m_{miss}^2 - m_{\pi^0}^2| < 0.007 \text{ MeV}/c^2$, corresponding to the requirement that at least one π^0 is present in the event. The value of the cut is motivated by the distribution of $m_{miss}^2 - m_{\pi^0}^2$ for MC events shown in Figure 3.1. After building

$$P_{\pi+}^i = (\vec{p}_{pos_i}, E = \sqrt{\vec{p}_{pos_i}^2 + m_{\pi+}^2}), i = 1, 2 \quad (3.6)$$

$$P_{e+}^j = (\vec{p}_{pos_j}, E = \sqrt{\vec{p}_{pos_j}^2 + m_{e+}^2}), j = 1, 2, j \neq i \quad (3.7)$$

$$P_{e-} = (\vec{p}_{neg}, E = \sqrt{\vec{p}_{neg}^2 + m_{e-}^2}) \quad (3.8)$$

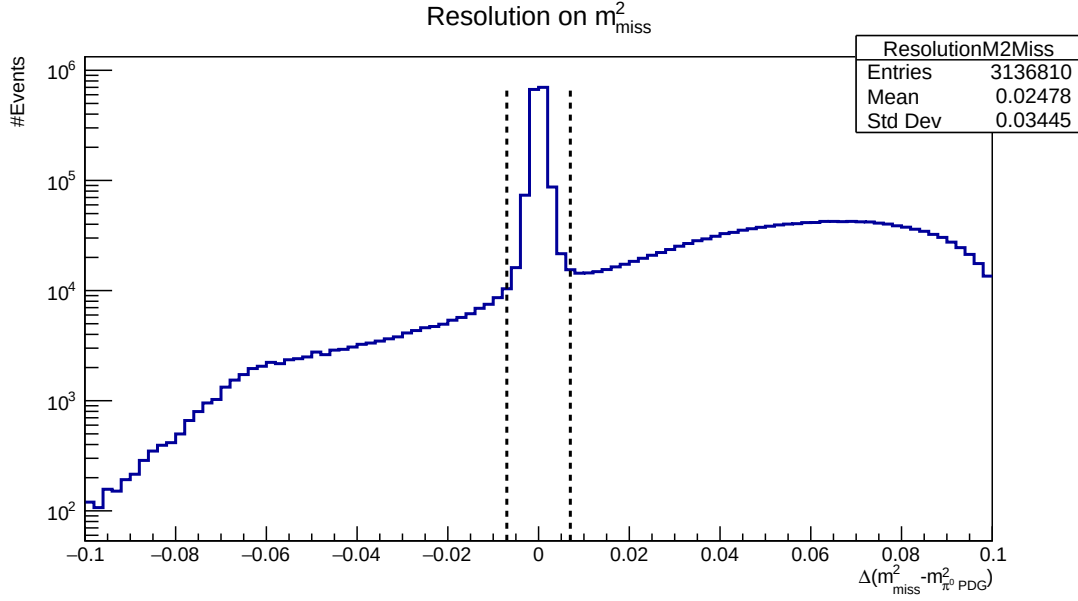


Figure 3.1: Distribution of $m_{miss}^2 - m_{\pi^0}^2$. The signal region is inside the black dashed lines.

where the subscripts 1, 2 have to be considered only if both positive tracks satisfy the previous requirement on m_{miss}^2 (otherwise the P_{π^+} and P_{e^+} are uniquely determined), the following quantities

$$(m1\gamma_{miss})_{ij}^2 = (P_{\pi^0}^j - P_{e^+}^i - P_{e^-})^2, \quad (3.9)$$

$$(m2\gamma_{miss})_{ij}^2 = (P_K - P_{\pi^+}^j - P_{e^+}^i - P_{e^-})^2, \quad (3.10)$$

with $i = 1(2)$, $j = 2(1)$ and

$$P_{\pi^0}^j = (\vec{p}_{beam} - \vec{p}_{pos_j}, \sqrt{(\vec{p}_{beam} - \vec{p}_{pos_j})^2 + m_{\pi^0}^2}) \quad (3.11)$$

are used to select events with the elliptical cut in the 2D distribution shown in Figure 3.2¹. Requiring that only one mass combination falls in the signal region, the remaining ambiguity on the positive track-particle assignment is solved and the event is selected.

For selected events, the direction of flight of the photon is taken as $\vec{p}_{beam} - \vec{p}_{\pi^+} - \vec{p}_{e^+} - \vec{p}_{e^-}$ and it is checked that the extrapolated point from the vertex to the LKr surface is within the LKr acceptance. By looping on all the available clusters in the event not associated to any track, the distance between the extrapolation and the cluster is computed, and if only one cluster is found within 50 mm from the extrapolation, the event is accepted and we get the value of E_{reco} as the value of the cluster energy. Two further cuts are applied to check that the cluster is far from dead cells (at least 20 mm) and not in the so called LAV12 shadow (distance from center of the calorimeter $180 < d < 1070$ mm). To extract the expected value of the energy (E_{exp}) we consider the following relation

$$E_{exp} = \frac{m_{\pi^0}^2 - 2P_{e^+}P_{e^-}}{2(P_{e^+} \cdot U_\gamma + P_{e^-} \cdot U_\gamma)} \quad (3.12)$$

¹The distribution in Figure 3.2 includes the quality checks on the LKr clusters, described in the following. Such cuts are now moved after the filling of such distribution histograms in order to disentangle event selection and calibration parts.

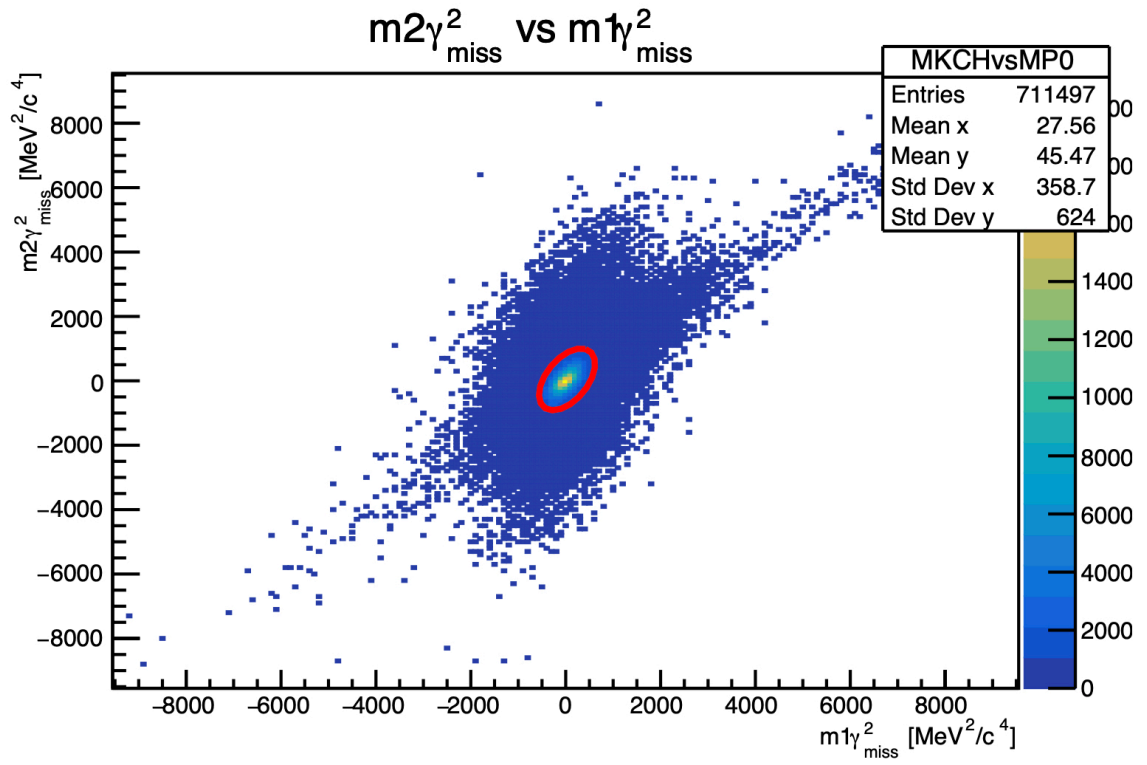


Figure 3.2: Distribution of $(m2_{\gamma_{miss}})^2$ vs $(m1_{\gamma_{miss}})^2$ on $K^+ \rightarrow \pi^+ \pi_D^0$ MC events. The signal region is highlighted in red. The overall acceptance of the selection evaluated on $K_{2\pi D}$ MC sample is $\sim 1.13\%$.

where U_γ is the normalized photon four-momentum, namely

$$U_\gamma = (1, \hat{u}_{miss}), \quad \hat{u}_{miss} = \frac{\vec{p}_{beam} - \vec{p}_{e^+} - \vec{p}_{e^-} - \vec{p}_{\pi^+}}{|\vec{p}_{beam} - \vec{p}_{e^+} - \vec{p}_{e^-} - \vec{p}_{\pi^+}|} \quad (3.13)$$

In a second improved version of the procedure, we correct the above quantity by replacing the direction given from the missing momentum with the one from the actual cluster position

$$\hat{u}_\gamma = \frac{\vec{r}_{cluster} - \vec{r}_{vertex}}{|\vec{r}_{cluster} - \vec{r}_{vertex}|} \quad (3.14)$$

This leads to a more precise computation of E_{exp} as shown in Figure 3.3.

3.4 Fitting procedure

From selected events we fill the 2D distribution R_D vs E_{reco} , with R_D (the subscript "D" stands for Dalitz), defined as

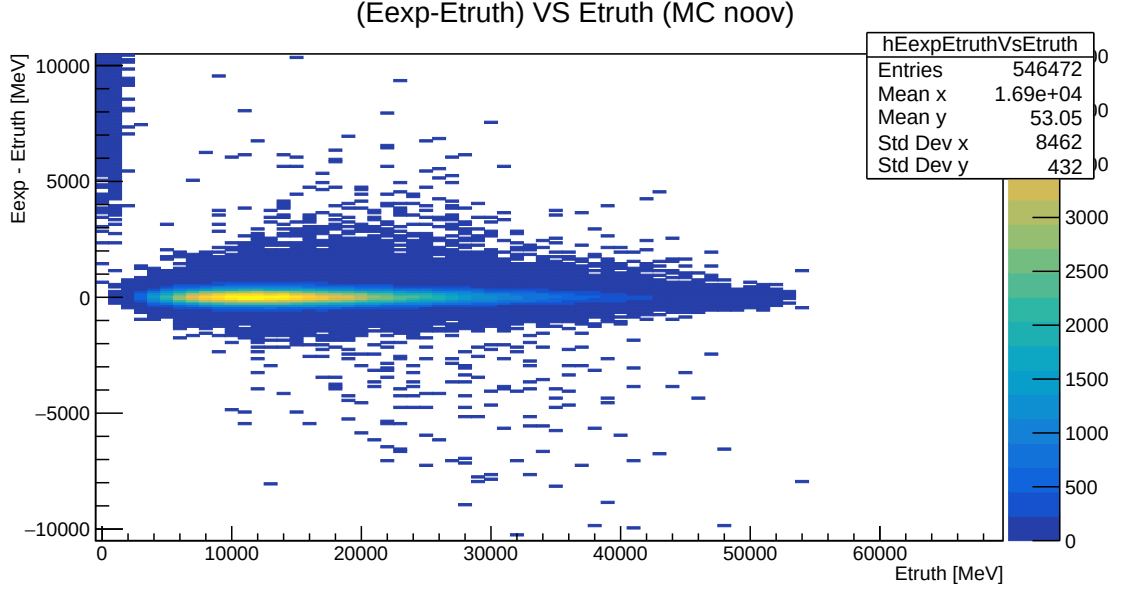
$$R_D = \frac{E_{reco}}{E_{exp}} \quad (3.15)$$

Since we expect a limited statistic in each Run, we studied at the same time the fitting procedure both on a single test Run (9001) and on the whole period 2018E to have the asymptotic approximation, assuming that the value doesn't change drastically from Run to Run in a given period. The obtained 2D distributions are shown in Figure 3.4. By slicing the 2D distribution into 1 GeV energy bins along the E_{reco} direction, and fitting each slice with a Gaussian, we obtain the plots shown in Figure 3.5, where the average and error on the average returned by the fit are plotted. In both the averaged and the single Run procedures we observed no dependence a constant trend of the correction on the reconstructed energy value, indicating that the correction is only a scaling factor.

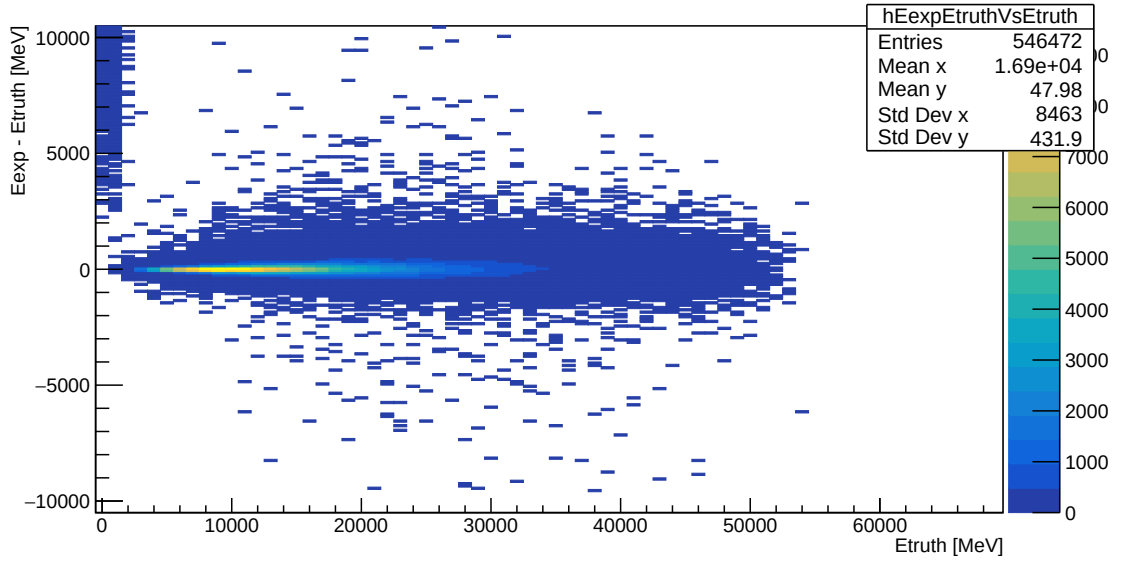
3.5 Comparison with standard calibration

Since we propose to implement the described calibration method as the default one (replacing the one based on 1D- π^0 mass) to compute the LKr fine calibration, we performed a comparison between the two methods on $K^+ \rightarrow \pi^+\pi^0, \pi^0 \rightarrow \gamma\gamma$ decays. The selected events are those from the a standard $K_{2\pi}$ selection available in the NA62 analysis framework. From selected events we took the selected track candidate, supposed to be the π^+ track, and the event time (the average of the times of the two selected clusters) and we match the track with the best-in-time GTK candidate to get the position of the decay vertex. We used the charged vertex to recompute the directions of the 3-momenta of the two photons and, from these, the di-photon masses (mass of the π^0), first using as energies those corrected with the standard method and then those corrected using the Dalitz correction. We were interested in the comparison of two fundamental parameters:

- reconstructed π^0 mass peak position
- stability with respect to the energy

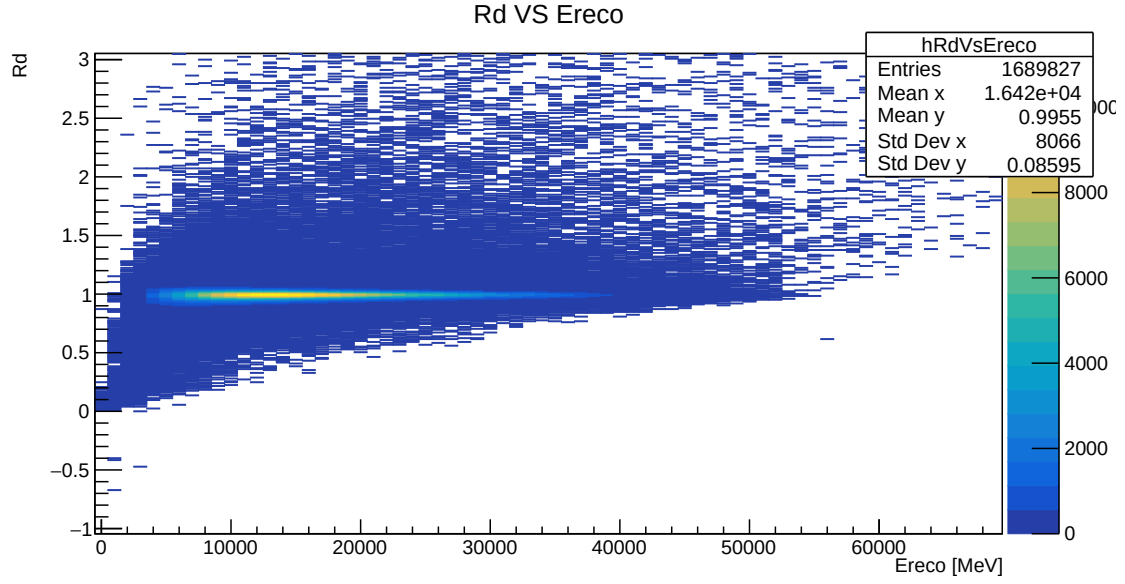


(a) $E_{exp} - E_{truth}$ vs E_{truth} with $\hat{u}_{miss}$
(Eexp-Etruth) VS Etruth (MC noov)

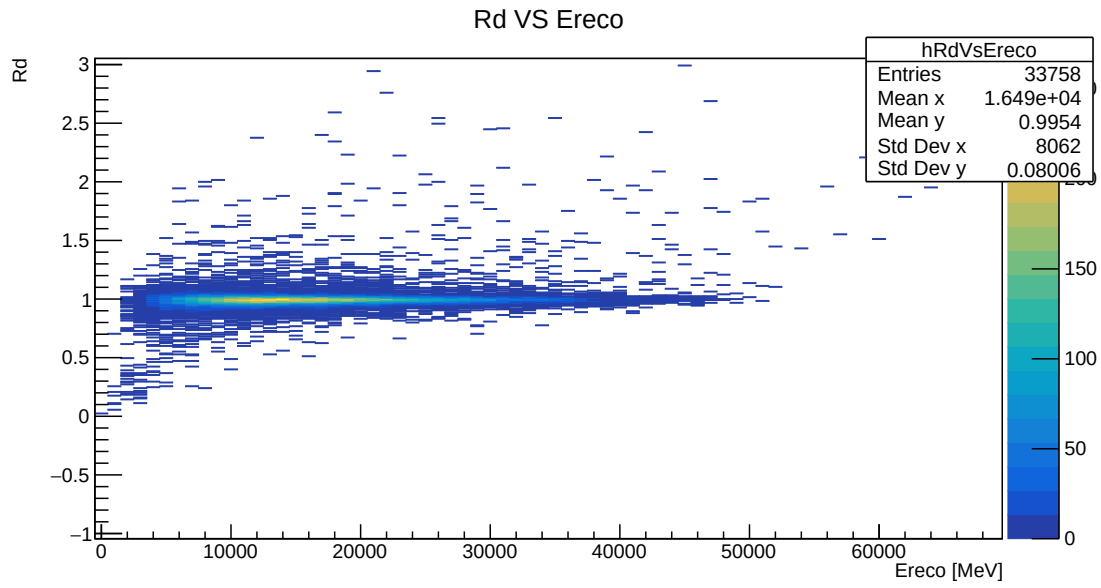


(b) $E_{exp} - E_{truth}$ vs E_{truth} with $\hat{u}_{\gamma}$

Figure 3.3: Distribution of the difference between the $E_{exp} - E_{truth}$ vs E_{truth} for MC events with E_{exp} computed using (a) the missing momentum direction and (b) the cluster position.

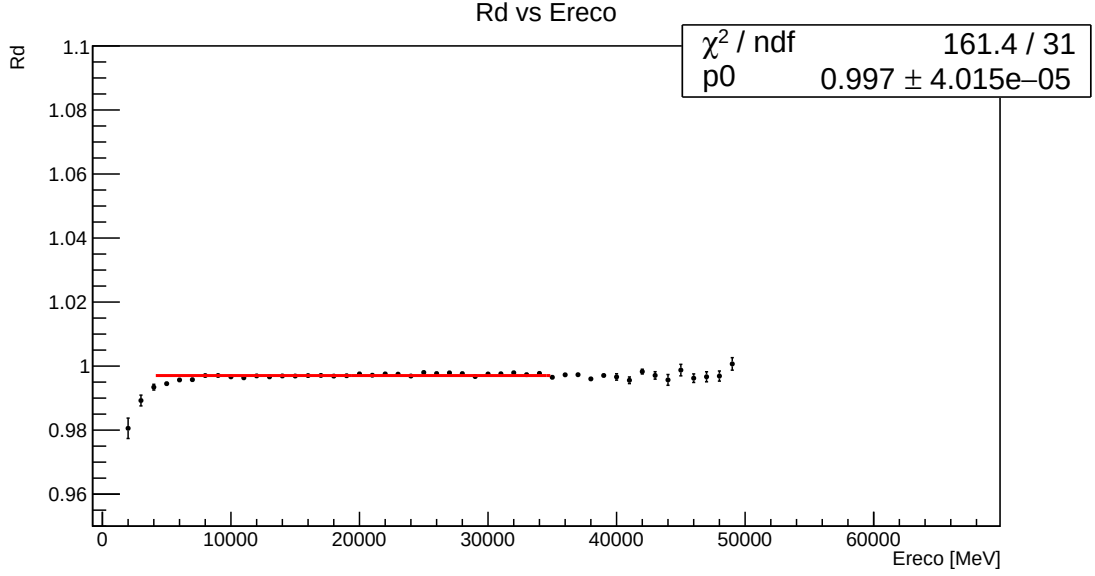


(a) 2018E

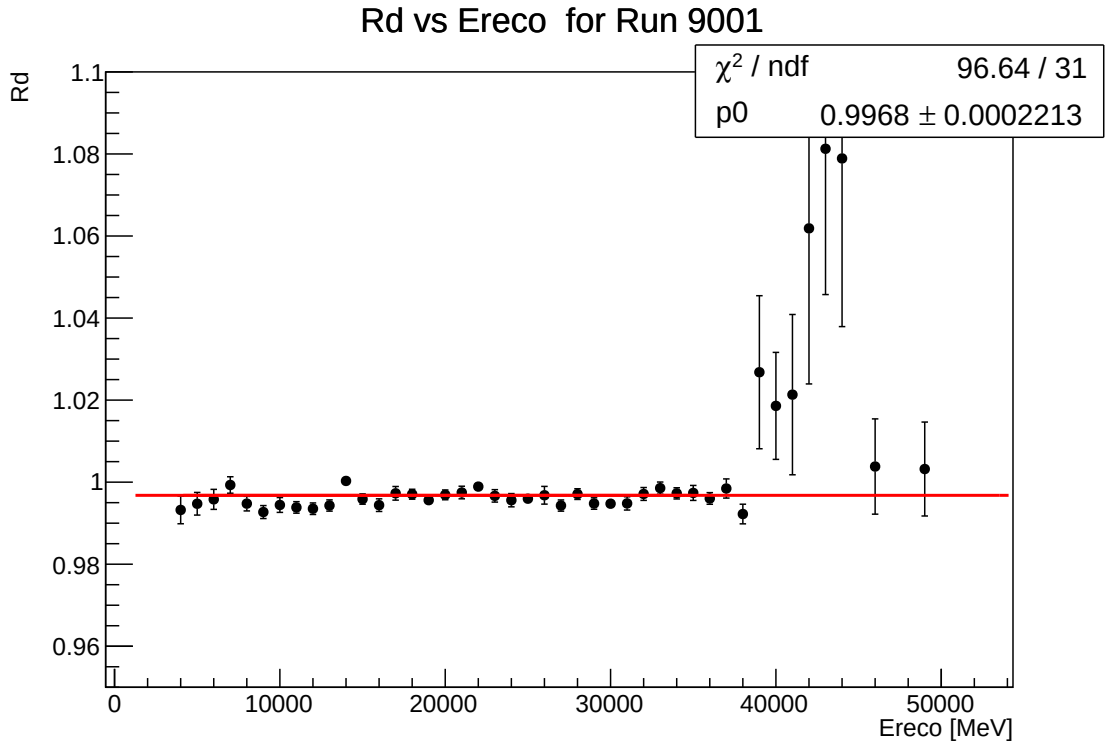


(b) Run 9001

Figure 3.4: Distribution of R_D vs E_{reco} obtained from selected events for (a) 2018E, (b) Run 9001.



(a) 2018E



(b) Run 9001

Figure 3.5: R_D vs E_{reco} obtained after the slicing+Gaussian fit procedure on (a) 2018E, (b) Run 9001. The red line is the constant function adopted as model to describe the observed trend. The value of the constant ($p0$) is shown in the top right corner.

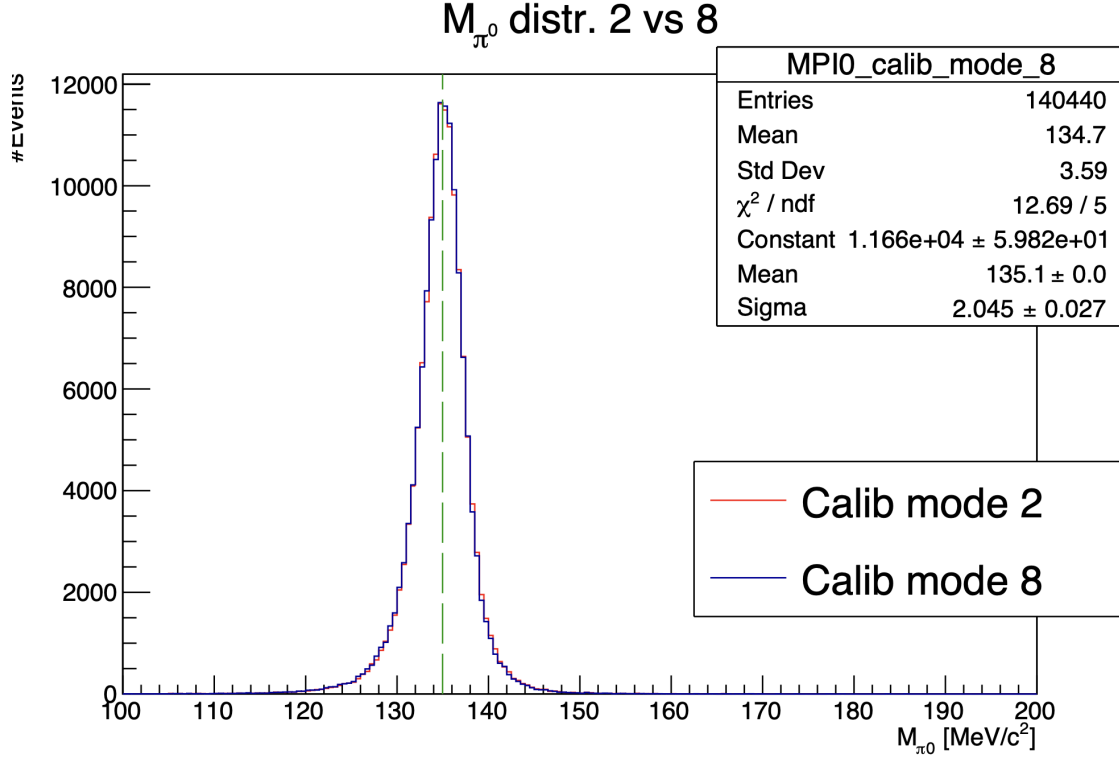


Figure 3.6: Reconstructed π^0 mass distribution with the corrected values of the di-photon mass. The red line shows the distribution for the standard 1D- π^0 calibration (indicated as "mode 2") while the blue line shows the distribution for the Dalitz calibration. The green dashed line shows the value of the PDG π^0 mass (indicated as "mode 8"). The 2 and 8 are referring to the index of the calibration in the NA62 analysis framework (1D- π^0 mass and Dalitz respectively).

The study was performed on the Run 9001. The result on the peak position is shown in Figure 3.6. We see that for both methods it lies in the same position, close to the π^0 PDG value. For a better understanding of the stability we studied the trend of the m_{π^0} mass as a function of the more(less) energetic cluster, indicated as E_{max} (E_{min}) in Figure 3.7. We see that both methods perform well when both corrected clusters are in the range $[5, 50]$ GeV, with almost no dependence on the energy. Mis-reconstruction of the π^0 mass starts to appear when both the photons exceed the validity range of the calibration, $E \leq 3$ GeV and $E \geq 50$ GeV. For example, from the E_{max} plot, we can see that, when $E_{max} \leq 12$ GeV, the mass of the π^0 start to be mis-reconstructed. This happens because when the more energetic photon has an energy of 12 GeV, than the other has lower energy by definition, and so both photons are corrected with a R in the limit of the validity range.

3.6 Feedback from analyses

In order to confirm the validity of the method we reported here the result of its application to the analysis described in chapter 2. Figure 3.8 shows the comparison of the reconstructed π^0 mass distribution from reconstructed $K^+ \rightarrow \pi^+ \pi_D^0$ decays. We remind that the selection is three-track based, with the energy of the cluster used to build the 4-momentum of the photon candidate.

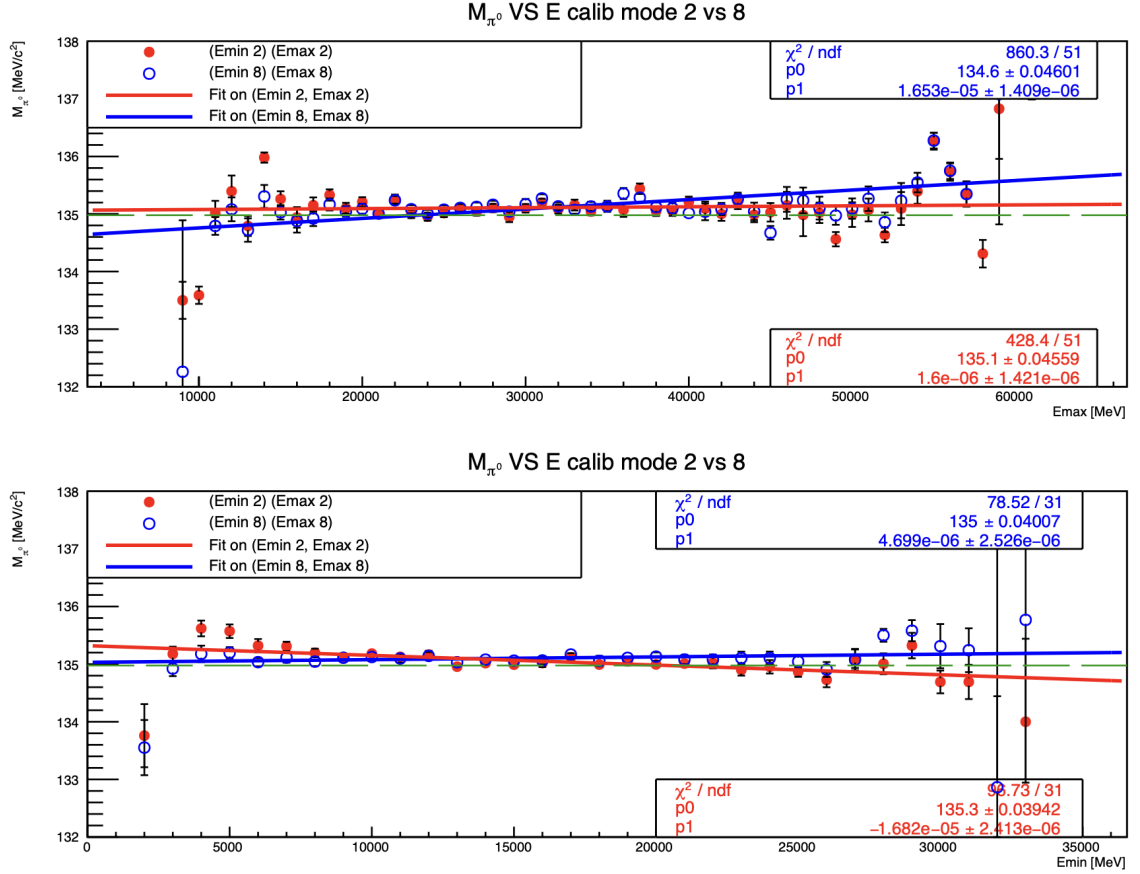


Figure 3.7: Stability plot of the reconstructed π^0 mass as a function of the energy of the two photons. The top plot shows the trend as a function of E_{max} , the energy of the most energetic photon in the pair, the bottom one shows instead the trend as a function of E_{min} . Quantities in red show the value of the masses obtained when the clusters are corrected with the standard calibration ("mode 2"), while the blue ones with the Dalitz calibration ("mode 8").

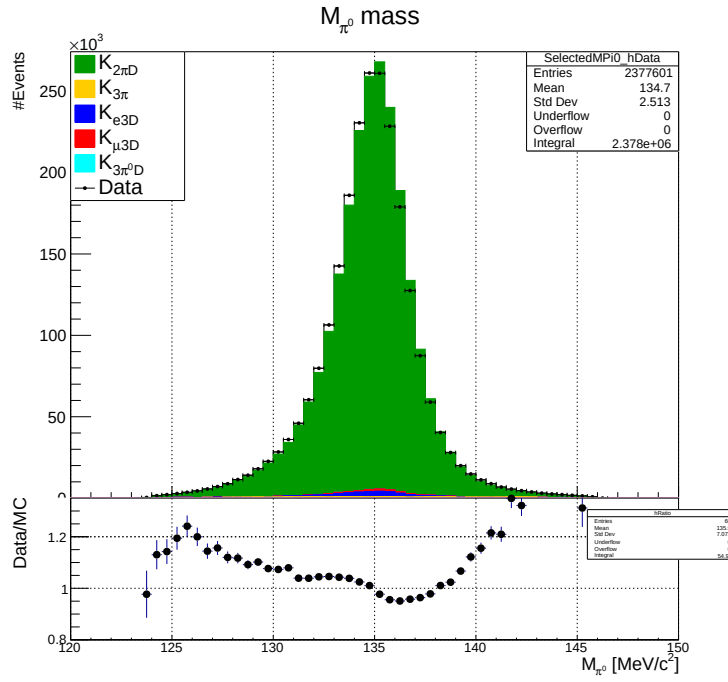
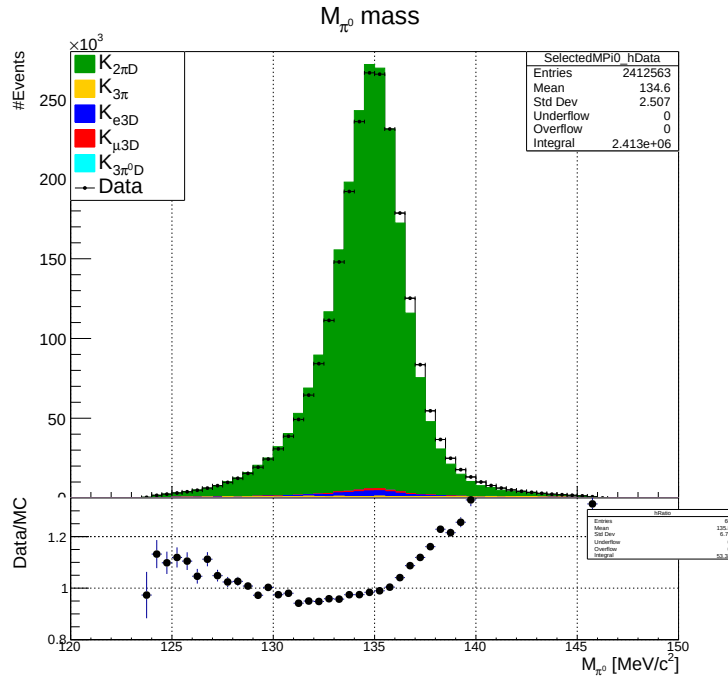
(a) Standard ($1D-\pi^0$ mass) calibration(b) π_D^0 calibration

Figure 3.8: Comparison of the reconstructed π^0 mass computed in the $K^+ \rightarrow \pi^+ \pi_D^0$ analysis. The solid distribution is the result of the application of the selection described in section 2.4 on data and MC events, for which the π_D^0 calibration was computed. In (a) the energy of the cluster is corrected with the standard calibration ($1D-\pi^0$ mass based), while in (b) it is corrected with the Dalitz calibration. We can see the reduction of the shift between data and MC in the latter case.

3.7 Status and prospects

The π_D^0 Dalitz calibration was implemented in the official data-processing system of the NA62 experiment, so it is automatically computed during the reprocessing of a Run and of MC samples. Considering the good results it provides, and the fact that it requires less than 1/4 of computation time, it is now adopted as the default calibration for the LKr clusters energy.

Considering the single photon final state of the π_D^0 decay, a possible further improvement might be the implementation of a 2D cell-by-cell LKr fine calibration (to be evaluated over multiple runs).

Chapter 4

Feasibility study of a Deep Learning algorithm for event classification by a seedless RICH circle detection

4.1 Introduction

Circles in the RICH detector are generated by charged particles that exceed the Cherenkov velocity threshold and produce photons that are reflected back to the two spots containing 976 PMTs each. At reconstruction level the two spots are overlapped to gain in spatial resolution in the x direction. All the PMTs can be arranged in an array of 1952 positions, with a 1 and 0 marking the fired/not fired ones. Due to the small number of Cherenkov photons and the PMT efficiency, in the majority of the events only few hits are available, generally arranged as arcs of the corresponding ring. Moreover some circles can overlap due to the presence of more than one charged particle in the final state of some K^+ decays (i.e. $K_{3\pi}$, $K_{2\pi D}$, etc.) or due to pileup activity. Finally, PMT noise can result in spurious hits, not belonging to any ring.

By seedless event classification problem we mean the problem of finding an algorithm able to separate events between "electron" and "not electron" events, whose decision is only based on the information of fired PMTs. The usefulness of the study relies on the fact that if such an algorithm exists, it could be implemented in any of the three trigger levels, providing a powerful high level trigger information for the classification of interesting events, today based on less sophisticated conditions or downscaling factors to reduce the incoming rates. The use of a Deep Learning (DL) algorithm to perform such a task has received little attention in the literature so far because "classical" (i.e. analytic) solutions exist to solve the problem [11]. One of the major limitations of such algorithms is the required computational power (and therefore required time). In some cases the tasks can be parallelized, but a specific hardware is required (GPU). DL offers a great alternative to the classical approach, which could better fit the timing requirements of an online trigger system. The required effort for tuning in advance the algorithm would lead to a machinery capable of online classifying an event in a very short time. In principal, the task considered appears to be within the field of application of this kind of algorithms,

which are generally used to solve complex nonlinear problems. Benefits could also be obtained at the analysis level, giving an independent way of event classification which could be used to probe the efficiency of the present algorithms.

In this Chapter we describe in detail the studies performed on these algorithms.

After a general introduction to Neural Networks in the first section, the Chapter is divided into two main sections: classifier models and regressor models, reflecting the time sequence that we used during the study.

The study was performed using the open source programming language Python and the machine learning framework Pytorch [42].

4.2 Theory behind Deep Learning

Artificial Intelligence (AI) is the study of the hardware and software implementation of algorithms that can be given to a computer to perform tasks with performances similar to those of humans. Machine Learning (ML) [26] is a sub-field of AI, defined as those set of algorithms through which a computer can "learn" directly from data, without any prior knowledge about the dataset under study. Neural Networks are a specific kind of ML algorithms, which mimic the structure of the human brain. An example of NN is shown in Figure 4.1. Computational units, called neurons, are arranged in layers (a single hidden layer is present in this case). The data flow from the input layer to the output one, passing through the hidden one. The input neurons are filled with the "features", that is the quantities that are available in the dataset. Each neuron of the hidden layer receives as input the weighted average of all the outputs from the neurons of the previous layer and a nonlinear activation function is computed on the weighted average as output. The data flow from the hidden to the output layer follows the same flow, with the hidden layer now being the input for the output one. The choice of activation function depends on the specific task the network is required to satisfy, and can influence also the learning speed of the algorithm. Deep Neural Networks differ from simple NNs only for the presence of more than one hidden layer.

There are two phases in a typical ML study:

- **Training:** this is the phase in which the algorithm learns from data. After defining the network structure and a cost function, the DNN is presented some data together with a label classifying the event (supervised training). Using an iterative procedure, some cost (or loss) function is minimized, by modifying the network parameters which are used to compute the weighted averages. Generally the training is performed by splitting the training dataset into batches, to speed-up the approach towards the minimum. When all the batches have been used, this is considered as the end of an epoch of training.
- **Testing:** in this phase the network is required to perform inference on new data with the same quantitative features of the training data. On this set the performances of the network are measured.

There is a middle-phase called validation, used to check the efficiency of the network on a set of events not used for training. This phase is generally used to tune the hyperparameters of the network (such as the number of layers or the number of neurons per layer) preventing their tuning on the test dataset which should be used

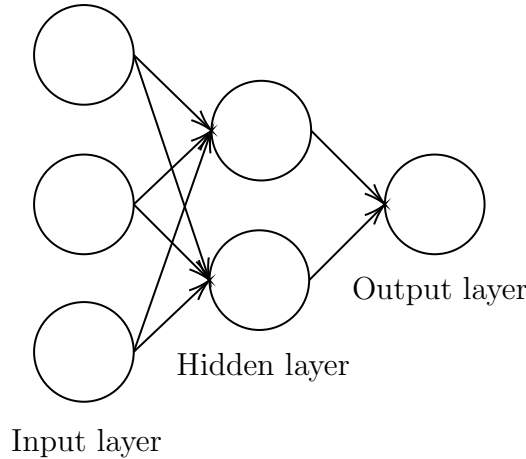


Figure 4.1: General Sketch of a DNN algorithm. This specific form is called Fully Connected Neural Network, because all the computational units of one layer are connected to all those of the following layer. The layers are also known as "dense" layers for this reason. The data flow is from left to right.

only at the end of the procedure.

Different kinds of tested network configurations will be described later in the appropriate sections.

4.3 Event generator

We used our own event generator for this study. The reason is due to the fact that the labeling of events through the "official" MC samples would be based on the RICH ring reconstructed with the offline software, which might propagate its bias also to the network. Moreover the generation of particular datasets (e.g. only one ring events, without noise etc.) is not easy with the official MC, which simulates many effects for the same event. Since our problem is purely geometrical, the results should not be influenced by this choice.

The generator does not take into account any "physics", it's just a simple ring generator that maps the positions of the hits of rings generated in a plane to the position of the nearest RICH PMTs, given the official map for these. A ring in an event is first generated with continuous hit coordinates, later forced to match a PMT center using a lookup table with the actual RICH PMT positions, according to the smallest distance. Hence the event becomes a list of 2048¹ elements with 0 or 1, specifying the not fired/fired PMTs. Examples of generated events are shown in Figure 4.2. Multiple parameters have been implemented in the generator:

- **n_events**: number of events;
- **n_rings**: number of rings in an event;
- **n_hits**: number of hits for each circle in the event;
- **cx_position**: x coordinate of the ring center;

¹For technical reasons, we used a lookup table with 2048 entries instead of 1952 as the number of PMTs, but the extra positions are never used (always 0).

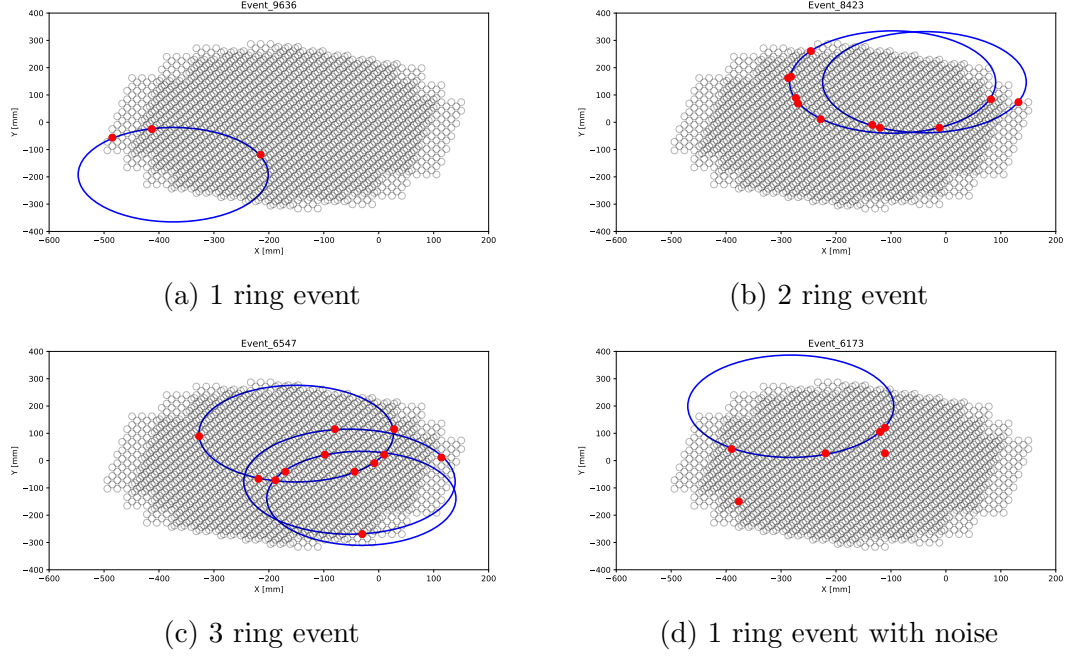


Figure 4.2: Examples of generic events generated by the implemented generator. The dark empty points represent the two overlapped 1024 PMT spots. Red points are the fired PMTs in the event. The original rings are shown to illustrate the working of the generator.

- **cy_position**: y coordinate of the ring center;
- **radius**: radius of each ring in the event;
- **r_sig_discr**: ring-radius range for classifying an event as signal;
- **force**: minimum number of signal rings in the event;
- **noise**: number of noise hits (i.e. hits not belonging to any ring);
- **n_min_hits**: number of minimum hits for a ring;
- **seed**: numerical value for the random generation (implemented to have reproducible results).

Each of the above quantities can be fixed or variable in a specific generation run. In the second case each variable is sampled from a uniform distribution with appropriate limits.

4.4 Classifier models

In this section we describe the tested classifier models. Formalizing the definition of "electron" events, we consider this to be an event in which at least a generated ring with a radius in the range $[185, 195]$ mm was present. Figure 4.3 shows why this range was chosen: basically electrons are the only particles with a fixed constant

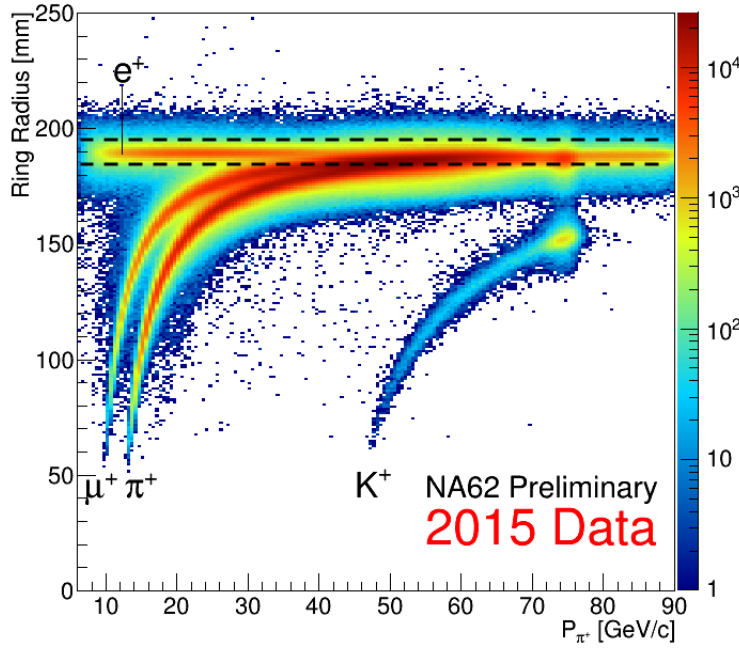


Figure 4.3: RICH ring radius vs π^+ Momentum. The area used to label events as "electron" (or signal) and "not electron" (or background) is shown.

radius independent from the momentum. Hence a detection of a "signal" ring in the event could be directly associated to a trigger decision. We define background events those in which no signal rings are present, so with all generated rings in the range [170, 185] mm. The lower limit of 170 mm was set arbitrarily. Since the main idea is to implement the algorithm in the trigger, the preprocessing on each event should be kept as simple as possible making the preprocessing + prediction phase as fast as possible to sustain the trigger rates.

We have investigated three dataset-type-classifier possibilities:

- **Neural Network** on a vector of 2048 inputs;
- **Convolutional Neural Network** on RICH images;
- **Convolutional Neural Network** on preprocessed RICH images with the Hough transform technique.

In the following each of these possibilities is described.

4.4.1 Fully connected Deep Neural Network classifier

The simplest approach is to inject into a DNN the vector of the 2048 PMTs that have fired in a certain event, requiring the network to predict the label, encoded as a 1 for the signal and 0 for the background. Hence the input layer of network consists of 2048 neurons while the output contains 2 neurons. The whole network is shown in Figure 4.4. The hidden layers are arranged in such a way that each has 1/4 of neurons of the previous one.

We formalize here the computation flow of the network, more information can be

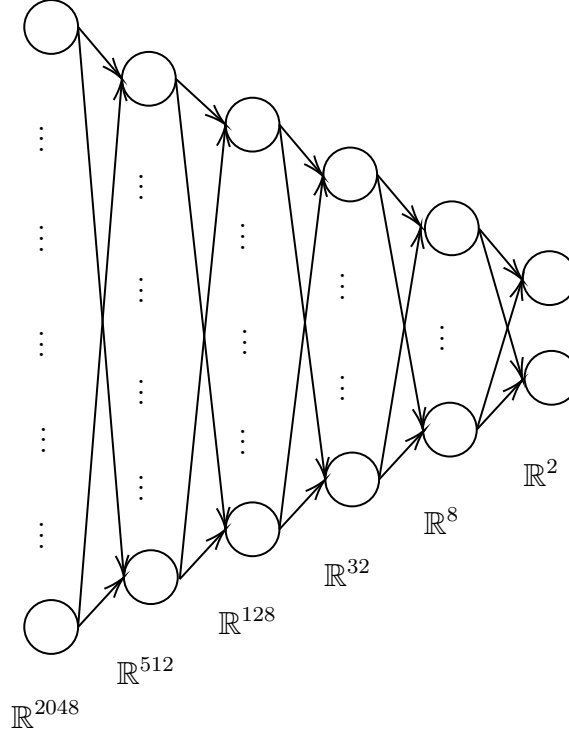


Figure 4.4: Structure of the DNN classifier. Each layer is a 1D vector with N neurons in it, indicated as $\mathbb{R}^N$. For clarity we draw only the first and the last neuron of each layer and only their connections with the following layer.

found here [24]. Defining x_i^l the value in the i th neuron in the l th layer and z_k^{l+1} the k th value in the $(l+1)$ th layer, they are related by

$$z_k^{l+1} = \sum_i^N w_{ki}^{l,l+1} x_i^l \quad (4.1)$$

being $w_{ki}^{l,l+1}$ the weight connecting the i th neuron of the l th layer to the k th neuron of the $l+1$ layer. The sum runs over the N neurons in the l th layer. The activation function for all the hidden layers is a ReLU, defined as:

$$\text{ReLU}(z_k^l) = \begin{cases} z_k^l & \text{if } z_k^l \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (4.2)$$

while for the output layer is LogSoftmax, defined as

$$\hat{y}_i^k = \text{LogSoftmax}(z_k^o) = \log \frac{\exp(z_k^o)}{\sum_{k=1}^2 \exp(z_k^o)}, o \equiv \text{output layer} \quad (4.3)$$

which represents the estimated probability that the i th instance injected in the model belongs to class k . Hence for each input instance is defined the output probability vector

$$\hat{\mathbf{y}}_i = (\hat{y}_i^0, \hat{y}_i^1, \dots, \hat{y}_i^C) \quad (4.4)$$

whose length is equal to the number of possible classes of the problem. During the training phase the Negative Log Likelihood Loss (NLLLoss) function is used to evaluate the network state of learning. Given the vector of the estimated probabilities

for each instance in the dataset $\hat{\mathbf{y}} = (\hat{\mathbf{y}}_0, \hat{\mathbf{y}}_1, \dots, \hat{\mathbf{y}}_M)$ and the target class index for each instance $\mathbf{y} = (y_0, y_1, \dots, y_M)$ (so each y_i is an integer taking a value from 0 to C), it is defined as:

$$\text{NLLLoss}(\hat{\mathbf{y}}, \mathbf{y}) = -\frac{1}{M} \sum_{i=0}^M \sum_{k=0}^C l_i^k, \quad l_i^k = \mathbf{1}\{y_i = k\} \hat{y}_i^k \quad (4.5)$$

where $\mathbf{1}$ is the characteristic function defined as

$$\mathbf{1} = \begin{cases} 1 & \text{if } y_i = k \\ 0 & \text{if } y_i \neq k \end{cases} \quad (4.6)$$

The NLLLoss is suitable for a classification problem because if the target class is different from the highest estimated class, this means that in the loss a high value is added because the network has given a low probability to the class that would have to be predicted, therefore the corresponding $\hat{y}_i^k$ is large. This function is reduced with an iterative algorithm, named Adam [31]. The parameters $w_{ij}^{k,k+1}$ are changed epoch by epoch by this algorithm to reduce the value of the cost function. The final prediction of the network is obtained by taking the neuron index of the maximum of (4.3)

$$\hat{y}_i = \text{argmax}(\text{LogSoftmax}(\mathbf{z}^o)) \quad (4.7)$$

where $\text{LogSoftmax}(\mathbf{z}^o)$ is applied elementwise. To prevent overfitting, that is when the network starts to adapt too much to the training data, losing generalization power, "Dropout" technique [27] is used in the hidden layers. It works by switching on and off randomly some neurons, forcing the network to use all the neurons at its disposal, not relying too much on a specific subset of them. "Dropout" is not applied during validation and testing phase, so the DNN can use all its neurons for prediction. For this reason generally the "Training loss", computed on the training dataset, is greater than the "validation loss", computed on the validation dataset. To avoid mixing too many effects in the analysis of the DNN performance, we proceed by first studying the performance on the simplest possible dataset and gradually increasing the difficulty.

Fixed single ring dataset, random radius

The purpose of this first test is just to check that every piece of the framework is working as expected, since rings are actually spread everywhere over the plane of the RICH PMTs. We generated 10^4 events with a single ring per event. The dataset is split into two parts:

- 8000 events for training
- 2000 events for validation

Example of events are shown in Figure 4.5. The task is implicitly simple since the information coming from the occupancy of each PMTs is sufficient to separate the signal and background classes. We can see this by the 2D histogram in Figure 4.6, in which for each event (i.e. each ring in this case) the PMT index bin is filled according to the

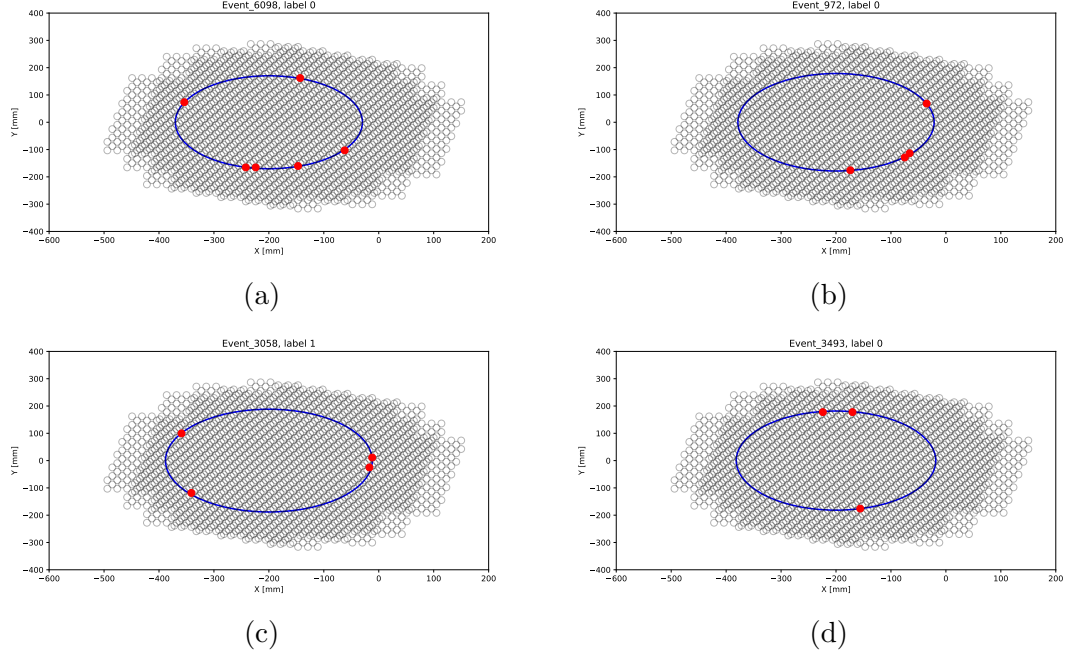


Figure 4.5: Examples for the current dataset. Rings with fixed center position and variable number of hits and radii. Figure 4.5a, 4.5b, 4.5d are labeled 0 while 4.5c is labeled 1 according to our definition of "background" and "signal" rings.

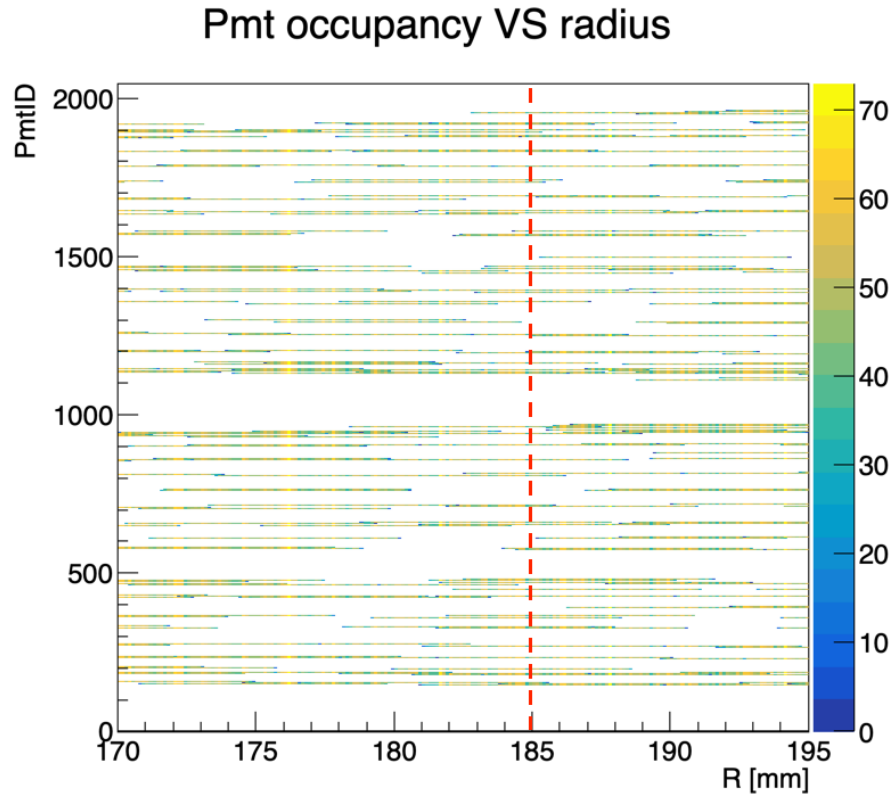


Figure 4.6: PMT occupancy vs R in the fixed center dataset. The red dashed line shows the class definition: signal (> 185 mm) and background.

radius of the ring in the event. From this distribution it's possible to get the indices of the PMTs which are filled only when the ring radius is greater than 185 mm and plot them to see that they are obviously arranged into a ring, as shown in Figure 4.7.

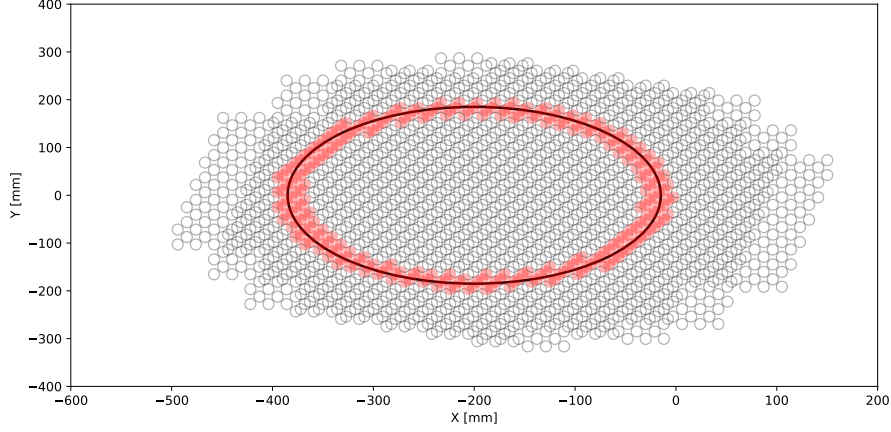


Figure 4.7: Visualization of the occupancy for the PMTs which are only populated when the radius of the generated ring is greater than 185 mm (black solid circle), which is the border between the signal and background regions.

This is a useful information that the network can learn to use to disentangle the two classes.

The result of the training is shown in Figure 4.8 (definitions for the performance indicators are in the Appendix B.1). We stop the training when the validation loss starts increasing after reaching a minimum. This prevent the overfitting problem. A value of 5 epochs is chosen as a tolerance for waiting for the validation loss to drop again to the minimum value.

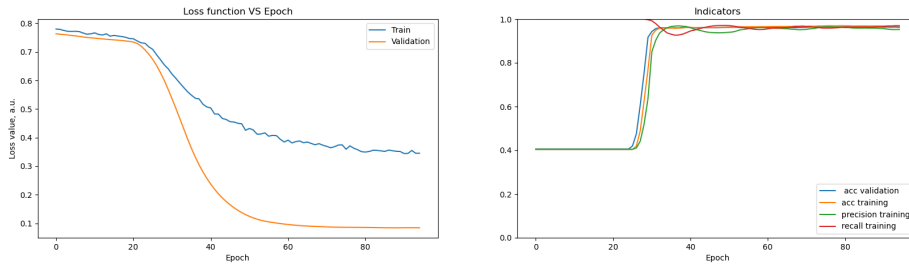


Figure 4.8: Results of the training phase on the described dataset. On the left: the training and the validation loss. The training loss is used by the network to tune the parameters while the validation is used to eventually auto-stop the training. Note that the loss is greater for the validation sample than for the training one despite the fact that the instances in the validation sample are not used for training. This is because the implementation of the "Dropout" technique. On the right: DNN performance indicators as a function of the training epoch.

The performances of the network have been evaluated on an independent sample (i.e. generated with a different seed) of 10000 events. The performances are

evaluated using the confusion matrix, whose definition is in Appendix B.1, shown in Figure 4.9.

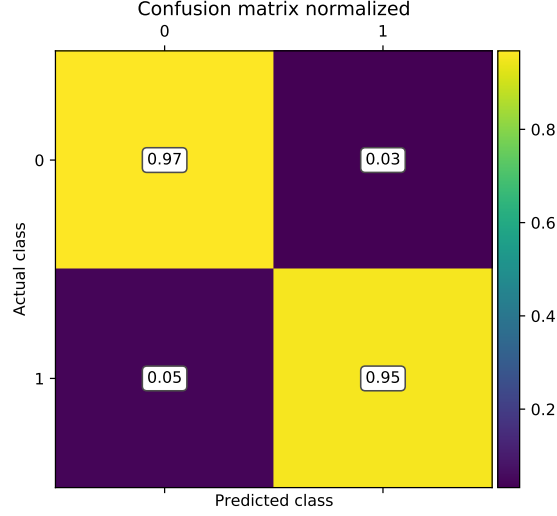


Figure 4.9: Plot version of the confusion matrix, showing the result of the prediction on the fixed center dataset. The matrix is normalized per row, according to the actual events in each classes. Since the task is very simple, the network performs very well with a correct classification of 97% of true negatives and 95% of true positives.

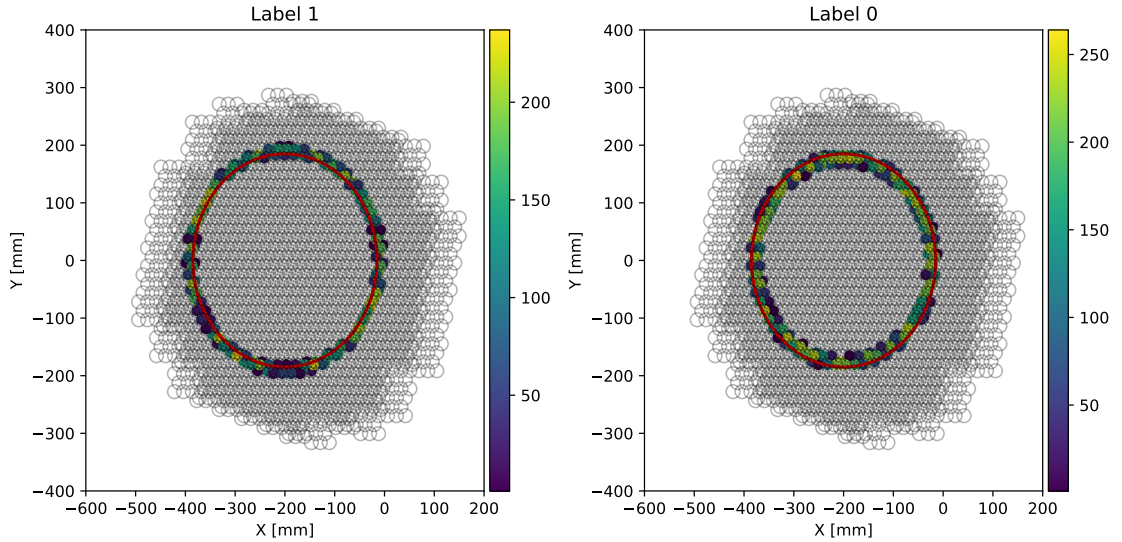


Figure 4.10: PMT occupancy plot for test events with fixed position of the center, as they have been classified by the network, after training it on events with fixed position of the center. On the left: occupancy for events classified as signal (or 1). On the right: occupancy for events classified as background (or 0). The red solid line represents a circle with radius 185 mm, showing the limit between background events and signal ones as defined in the dataset. The plot shows that when the prediction is signal the most occupied PMTs are out of the red line; the opposite with background circles.

As expected, the results are very good since the task is very simple. Just to be sure that the output is not due to random guessing, we plot the occupancy of each PMTs in events that the network classifies as signal or background. The plot is shown in Figure 4.10. It's interesting to see that the DNN is actually exploiting those channels that through the occupancy plot result populated only for a certain radius range (in particular the signal one).

Fixed X , random Y center coordinate, single ring dataset, random radius

We exploit the option for controlling the center position to generate 10^4 single ring events with the x coordinate kept fixed and a variable y . Examples of the dataset are show in Figure 4.11.

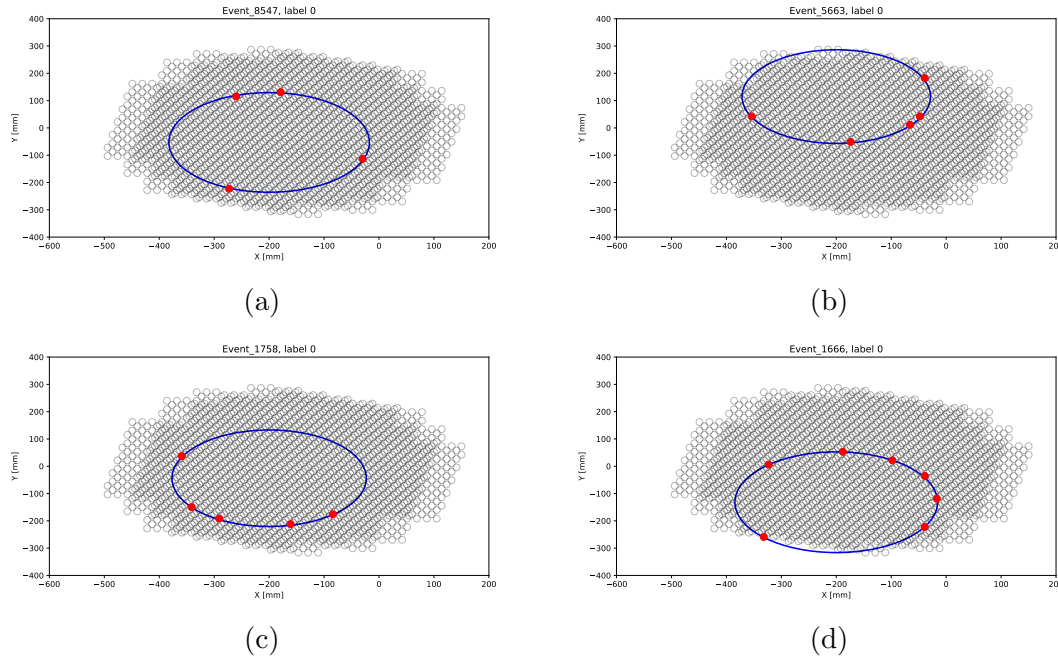


Figure 4.11: Examples of events for the fixed x dataset: rings with fixed horizontal center position and variable number of hits and radii. Figure 4.11a, 4.11b, 4.11d are labeled 0 while 4.11c is labeled 1.

To see how much the addition of the y variability mixes up the PMT information, the PMT occupancy plot is shown in Figure 4.12. Training and validation statistics is as before (8000-2000).

The result of the training is shown in Figure 4.13. The test of the DNN has been performed on a separated sample (also with fixed x position) and resulted in the confusion matrix shown in Figure 4.14. Looking at the feature importance plot, shown in Figure 4.15, it's possible to see how the PMTs on which the network bases its decision are less clear than the fixed center dataset. Basically the movement on the y axis leaves only some PMTs to the external band on which the network can rely on its decision. When an event has a ring with PMTs in the side bands it is likely to be put in the signal class, but in all other cases it's likely to be classified as background. As expected, the fraction of miss-classifications increases. To compensate the higher difficulty of the task, we increased the number of generated

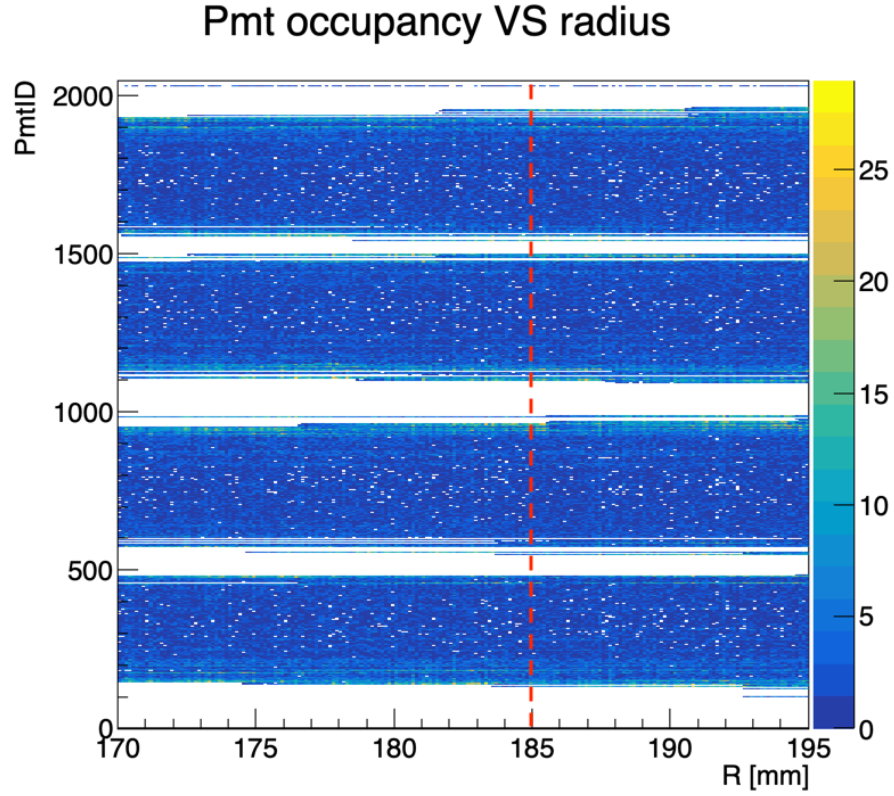


Figure 4.12: PMT occupancy vs R for the fixed x dataset. Compared to the case of fixed xy position, here there isn't a clear separation between the signal and background samples. The DNN is expected to perform worse than before due to the absence of a clear set of PMTs on which to base a decision.

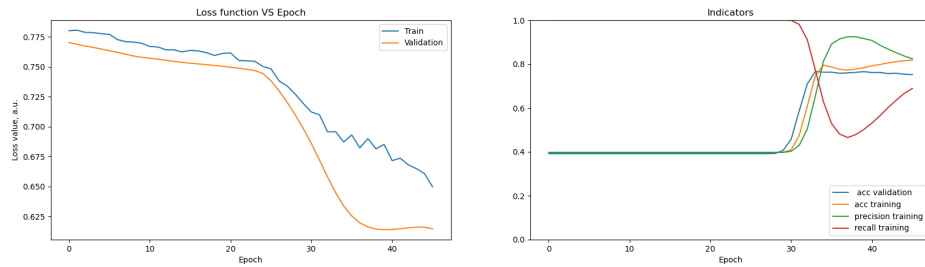


Figure 4.13: Result of the training phase on the fixed x dataset. On the plot on the left the loss function shows that more epochs are needed to reach a good separation between signal and background. However, after just few epochs the DNN starts to overfit. The same is shown in the plot on the right where the validation accuracy starts to increase before stopping.

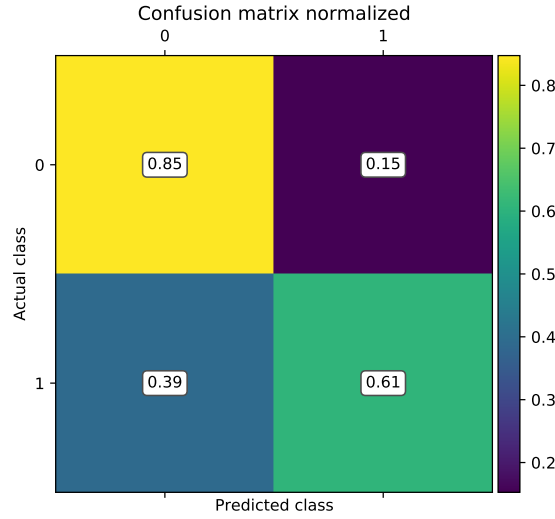


Figure 4.14: Confusion matrix obtained through predictions of the trained model (on the fixed x dataset) on a test dataset, also with x coordinate kept fixed. The matrix is normalized per row to put in evidence the fractions of right/wrong classifications in both classes.

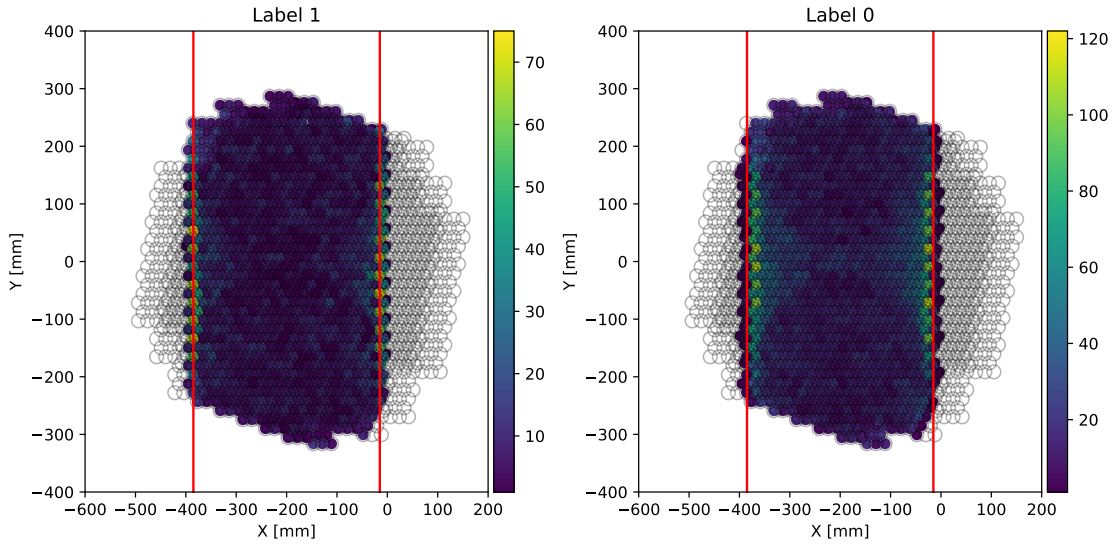


Figure 4.15: PMT occupancy plot for test events with fixed x position of the center, as they have been classified by the network, after training it on events with fixed x position of the center. On the left: occupancy for events classified as signal (or 1). On the right: occupancy for events classified as background (or 0). The red solid lines represent the coordinate $x = -200 - 185$ mm and $x = -200 + 185$ mm, where -200 is the x coordinate of the center, kept fixed, and 185 is the value defining the border between signal and background regions.

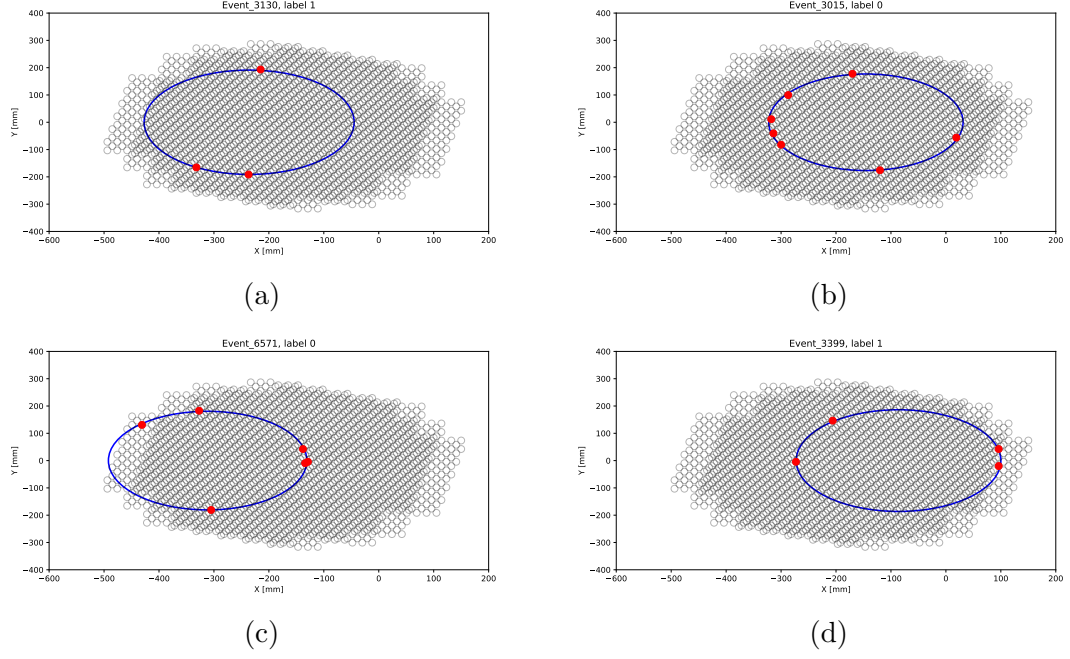


Figure 4.16: Examples for the current test. Rings with fixed y center position and variable x . Figure 4.16a, 4.16b, 4.16d are labeled 0 while 4.16c is labeled 1.

events to 50^4 , which permitted to increase the overall classification efficiency to $\sim 87\%$.

Random X, fixed Y center coordinate, single ring dataset, random radius

We performed a similar study by varying the y coordinate of the center and keeping x fixed. Examples of events from the dataset are shown in Figure 4.16. The conclusions are similar to the previous case, with a lowering of the performances both in the training (see Figure 4.17) and testing (see Figure 4.18), which is expected from the PMT occupancy plot vs R (Figure 4.19) of the training and confirmed by the occupancy plot on the test dataset (Figure 4.20).

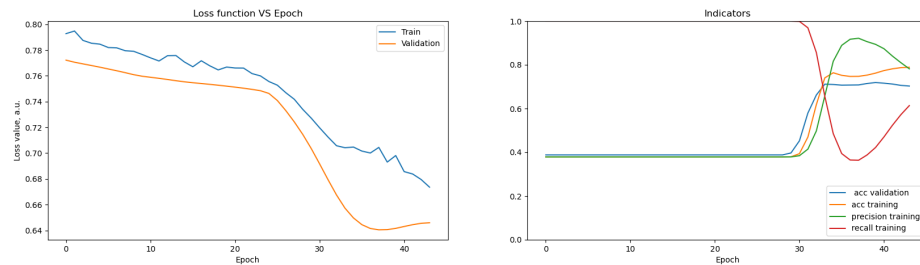


Figure 4.17: Result of the training phase on the fixed y dataset. The shapes of the curves are quite similar to the x fixed dataset, showing a period in which the DNN looks for a border to separate the signal and background (left plot) without a remarkable drop in the loss function, and a saturation just after few epochs (plot on the right).

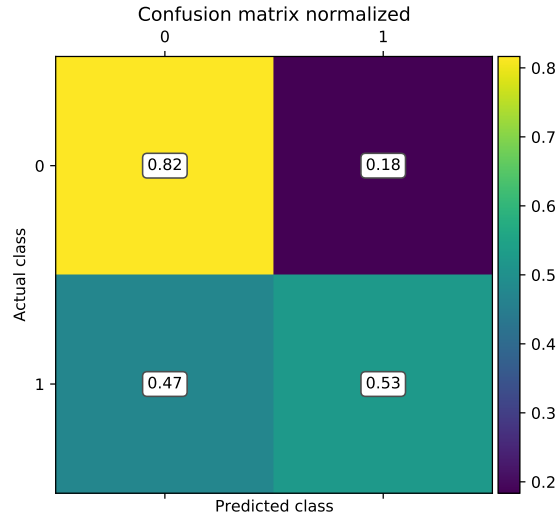


Figure 4.18: Confusion matrix on the test set. The fractions of false positive and false negative has increased wrt the fix x dataset due to the increasing difficulty in the task itself.

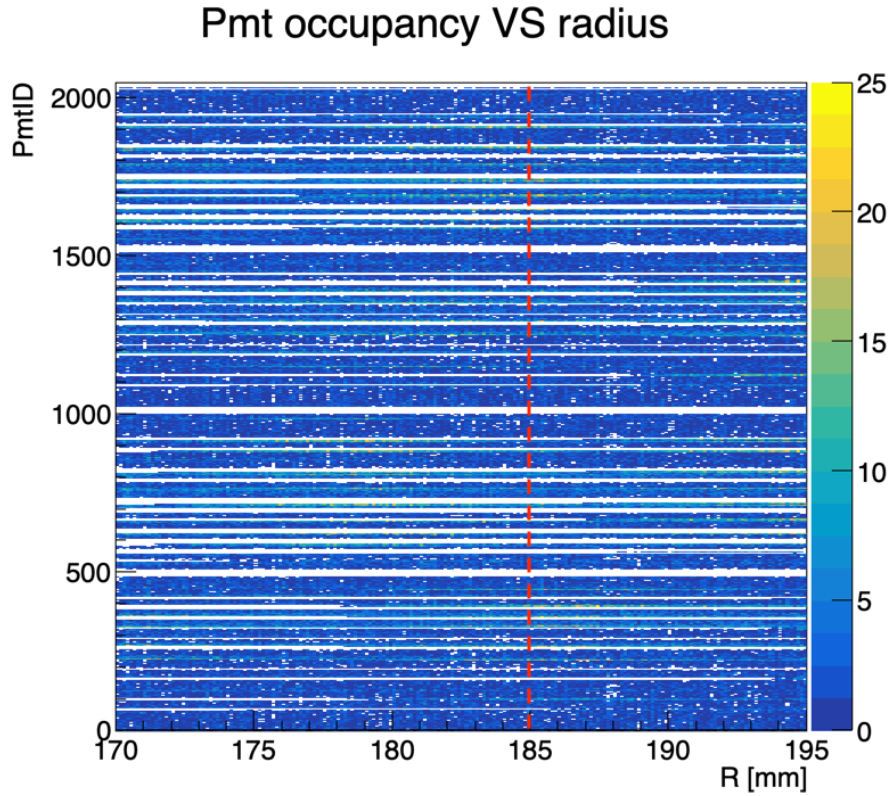


Figure 4.19: Pmt occupancy vs R. This plot shows clearly that the task is more complex wrt the fixed x case. Indeed there are no PMTs that are occupied only when a ring belongs to the signal or to the background subsample.

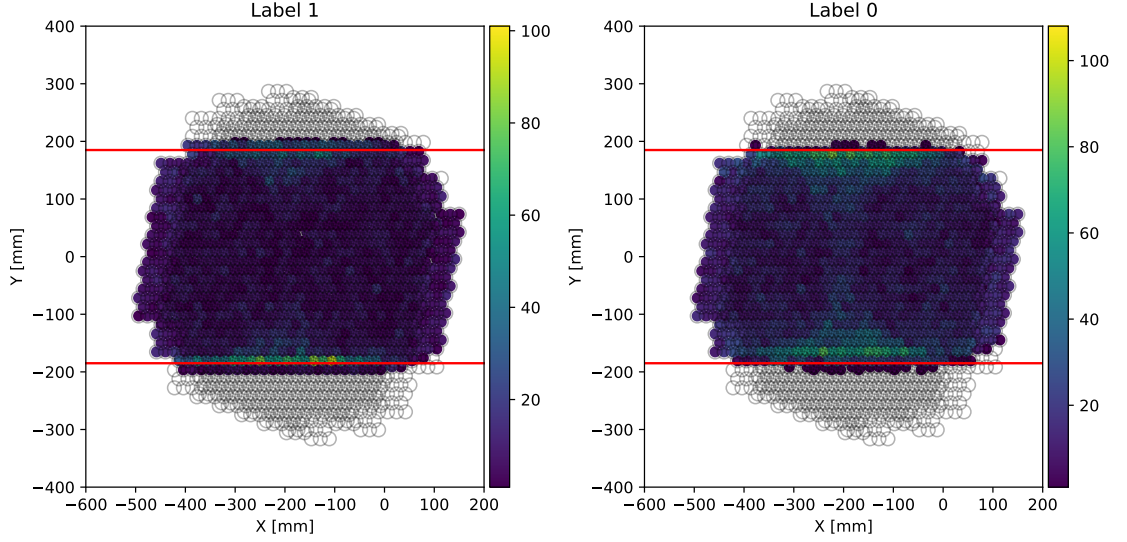


Figure 4.20: PMT occupancy plot for test events with fixed y position of the center, as they have been classified by the network, after training it on events with fixed y position of the center. On the left: occupancy for events classified as signal (or 1). On the right: occupancy for events classified as background (or 0). The red solid lines represent the coordinate $y = -185$ mm and $y = +185$ mm, where 185 is the value defining the border between signal and background regions. The coordinate y of the center is fixed and equal to 0.

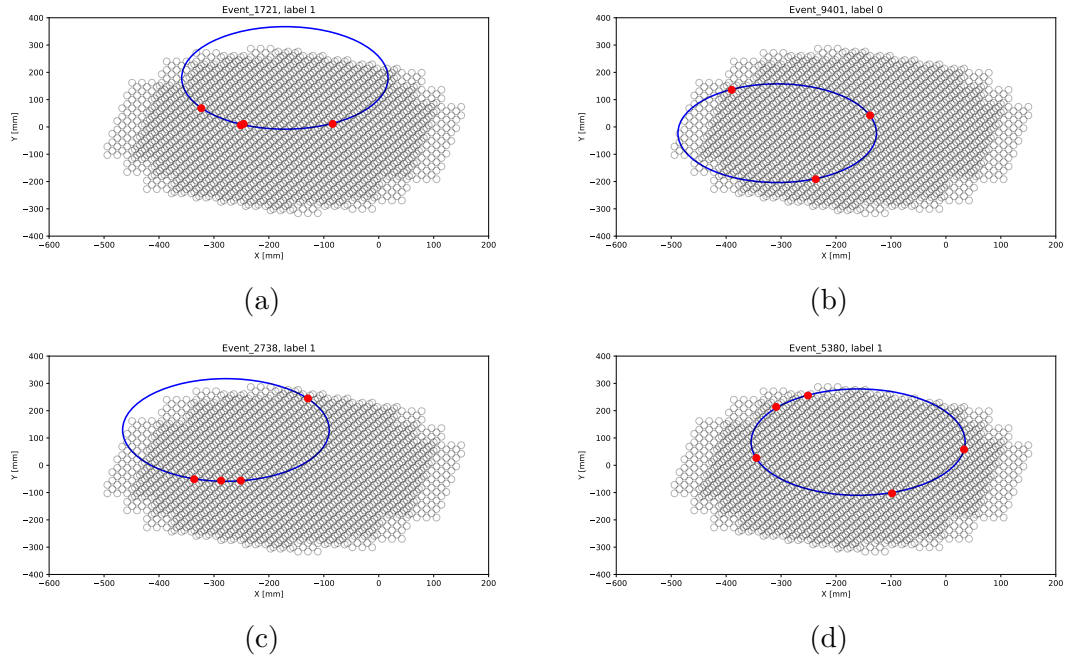


Figure 4.21: Example of events of the dataset. The center in each event occupies a different positions. However no noise hits are present, so this is an almost "real" sample.

Random center coordinate, single ring dataset, random radius

By introducing a random position for both the x and y center coordinates, this dataset is formed by events that can be considered as a first "real" case, although the absence of noise makes it still a somewhat idealized situation. Examples of events from the dataset are shown in Figure 4.21. Considering the decrease in the DNN inference capability due to the movement of only one coordinate of the center, we expect a further decrease in this case. Indeed looking at the PMT occupancy plot in Figure 4.22, all PMTs are equally populated both in signal and in background rings, making very difficult to find a criterium to disentangle the two classes.

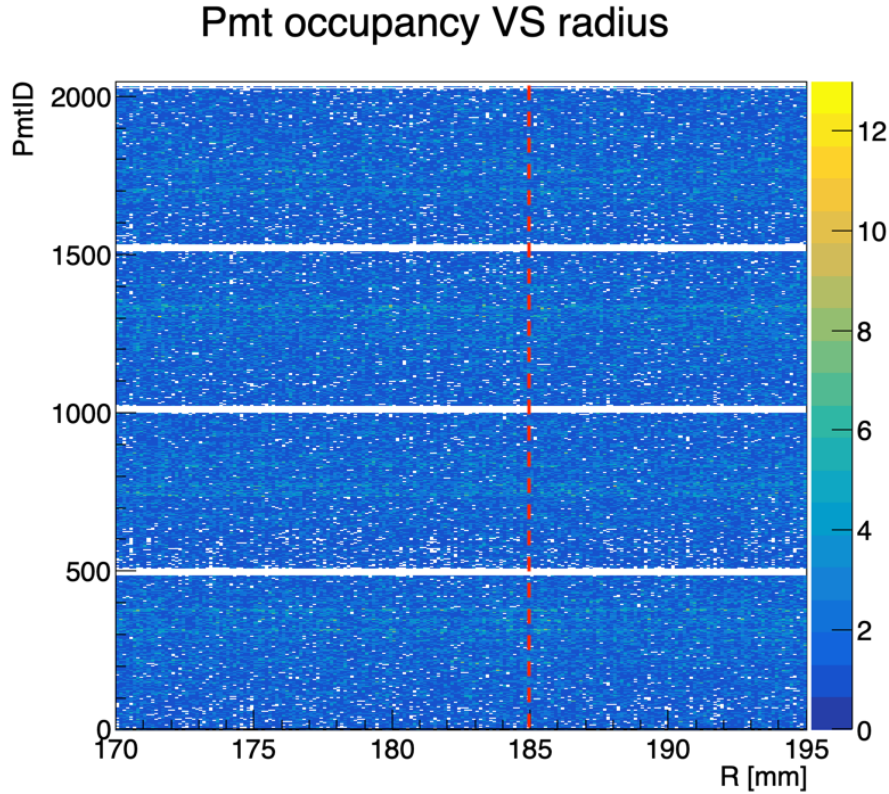


Figure 4.22: PMT occupancy vs R for the random center position dataset. The plot shows a completely uniform distribution for the occupancy as a function of the ring radius. Disentangling the two classes becomes impossible for the DNN since there is no a clear way to separate them.

The results of the training, testing and the feature importance are shown respectively in the Figures 4.23, 4.24 and 4.25. As a cross-check of our interpretation of the results, we perform a mixed training-testing study. We train the DNN on centered rings, so fix xy position of the center, while testing on random center position dataset.

We show the result just in terms of the feature importance plot in Figure 4.26. As in the fixed x coordinate study, we increased the size of the training dataset to 25×10^4 events, without any further improvement.

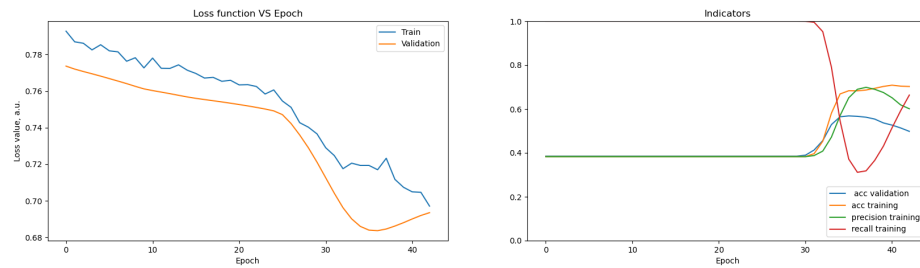


Figure 4.23: Results of the training phase on dataset with random position of the center.

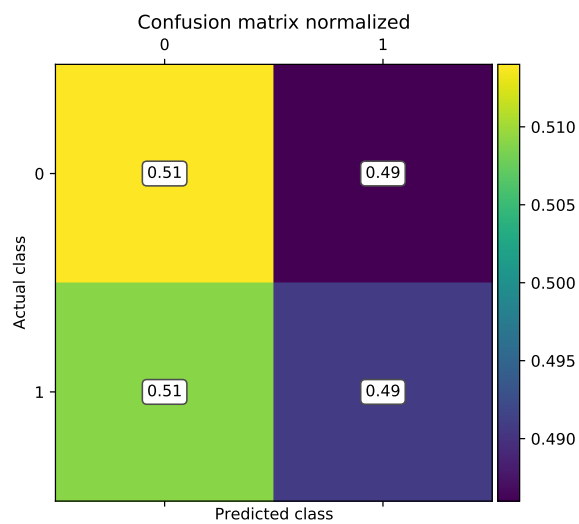


Figure 4.24: Confusion matrix on the test sample. Due to the high complexity of the task, the DNN prediction is almost random with all the scores close to 50%.

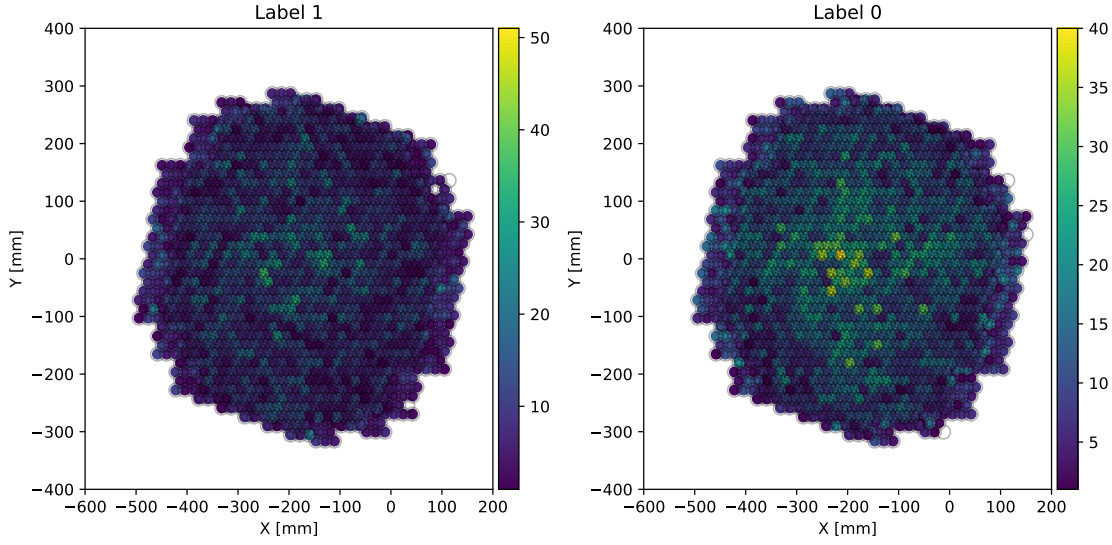


Figure 4.25: PMT occupancy plot for test events with random position of the center, as they have been classified by the network, after training it on events with random position of the center. On the left: occupancy for events classified as signal (or 1). On the right: occupancy for events classified as background (or 0). The PMTs are almost equally populated in both plots, showing that the network is not very confident during inference.

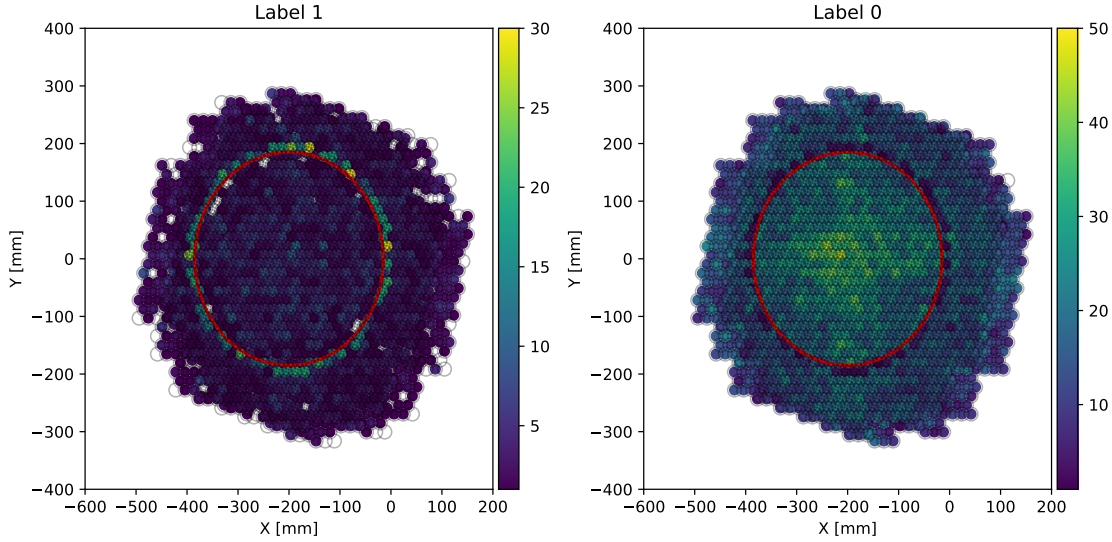


Figure 4.26: PMT occupancy plot for test events with random position of the center, as they have been classified by the network, after training it on events with fixed position of the center. On the left: occupancy for events classified as signal (or 1). On the right: occupancy for events classified as background (or 0). This puts in evidence that the network is able to disentangle the two classes only when the events are close in features to the the ones it has been trained on, even in the case they are mixed with other events.

Conclusions

Considering the obtained results, we decided to stop with this kind of approach without increasing further the dataset complexity. We verified that training directly on multi-ring events is not a good approach to solve the problem.

4.4.2 Convolutional Neural Network classifier

One of the weakest points of the DNN approach is that all the neurons of one layer are connected to the neurons of the following layer. This means that each PMT is related to all other ones, even if they are very far away one from each other. Hence the next step in our study was the implementation of a Convolutional Neural Network (CNN). These networks were first introduced in the '80, but their utility was highlighted at the beginning of 2000s due to overcome the needs of a strong computation power during training. The main idea behind CNNs is that biological vision process works through stimuli coming from a restricted region of the visual field, called the receptive field, and not by the entire figure. The structure of this kind of networks tries to mimic this behaviour. Considering that each (gray-scale) image can be translated into a matrix, the network acts directly on the matrix (and not a 1D vector) by doing a convolution. This operation is implemented using a "kernel" (or filter), that is repeatedly applied at each position, moving from the upper-left of a figure (see Figure 4.27), to the bottom-right corner. The kernel is basically a submatrix of a certain side length, containing some weights that sample the overlapping bins of the original matrix. The shift value to move the filter is called "stride". Usually multiple kernels are used on the same image, so from a single matrix multiple outputs (or channels) are obtained. At the end of the convolution process a pooling operation, which reduces the sampling output, is implemented. For example, the output of the convolution can be subdivided into 2×2 submatrices and for each one only the maximum value is taken (see Figure 4.28).

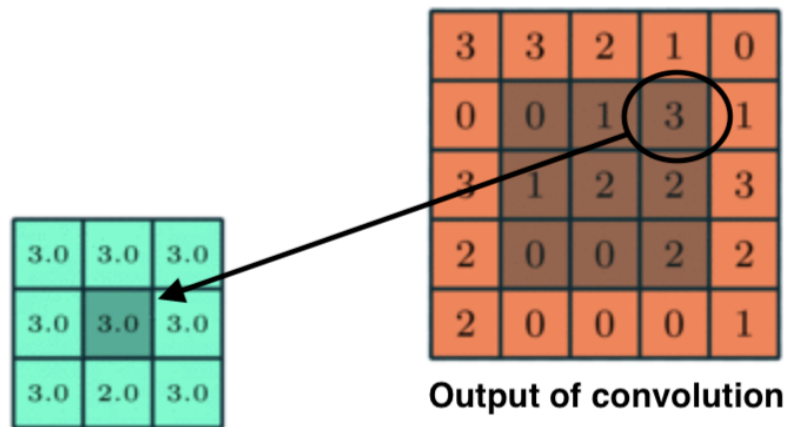


Figure 4.28: Example of a pooling operation. This case shows a max pooling, where the maximum of each 3×3 convolved image is taken as the output [43].

- **MaxPool1:** layer that for each of the 8 output images take the maximum of a 2×2 sub-matrix;
- **Convolution:** second convolution operation performed on each of the 8 channels outputs of the previous layer. We fixed a 16 channel output image (kernel size = 3, stride = 1 as before).
- **MaxPool:** size = 2;
- **Convolution:** 16 channel input image, 32 channel output image, kernel size = 3, stride = 1;
- **MaxPool:** size = 2;

At the end of the convolutional part the original image is decomposed into 32 8×15 -shaped images. The total number of pixels is $32 \times 8 \times 15 = 3840$, each representing the input to the fully connected part of the network, with the following structure:

- **Dense:** input 3840, output 960;
- **Dense:** input 960, output 240;
- **Dense:** input 240, output 60;
- **Dense:** input 60, output 2;

The activation function for all the layers is ReLU except for the last layer where it is LogSoftmax.

Dropout layers are used after each convolutional and dense layer with a probability of 0.5.

***Ad hoc* Dataset modification**

We decided to perform the same studies done for the 1D 2048 vectorial dataset, to compare the differences between the two classifier models. Hence also in this case we will explore 4 datasets, all containing a single ring per event:

- fixed ring position;
- random x with fixed y position of the center;
- random y with fixed x position of the center;
- random xy position of the center.

In the cases the radius and the number of hits of each ring in each event were variable.

Due to the technical implementation of the CNN, which acts directly on 2D input matrices, a change in the event generation was required. Instead of mapping each (real number pair) generated hit to its nearest PMT and creating a 1D vector, here the hit was mapped onto a grid of $n_x = 134$ and $n_y = 78$ coordinates, where n_x, n_y are the number of different x, y coordinates of all the PMTs of the RICH, after superimposing the two spots. The x and y coordinates of each bin are those of the official PMTs. The mapping is shown in Figure 4.30. As can be seen in Figure

4.2, where the two spots overlaps the distance between two PMT positions is less than where there is no superposition (like at the edges). Since this disposition is reproduced in the mapping, there is a distortion in the shapes of the rings, leading to ellipses.

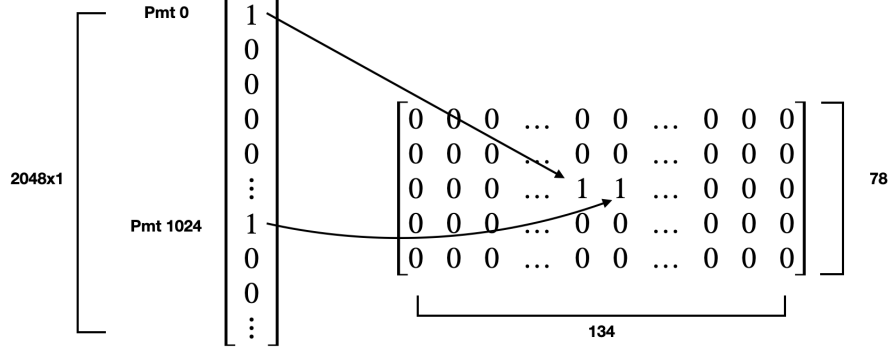


Figure 4.30: Mapping scheme for the CNN dataset. On the left there is the 2048 shaped dataset where near points can occupy positions that are far away from each other even when the two PMTs are in reality very close. On the right there is the new 2D mapping where proximity is maintained. For example if the PMT 0 and PMT 1024 were near in reality, in the 2048 1D representation they would be put very far away from each other, while they would be near in the 2D one.

Examples of simulated events with this mapping are shown in Figure 4.31.

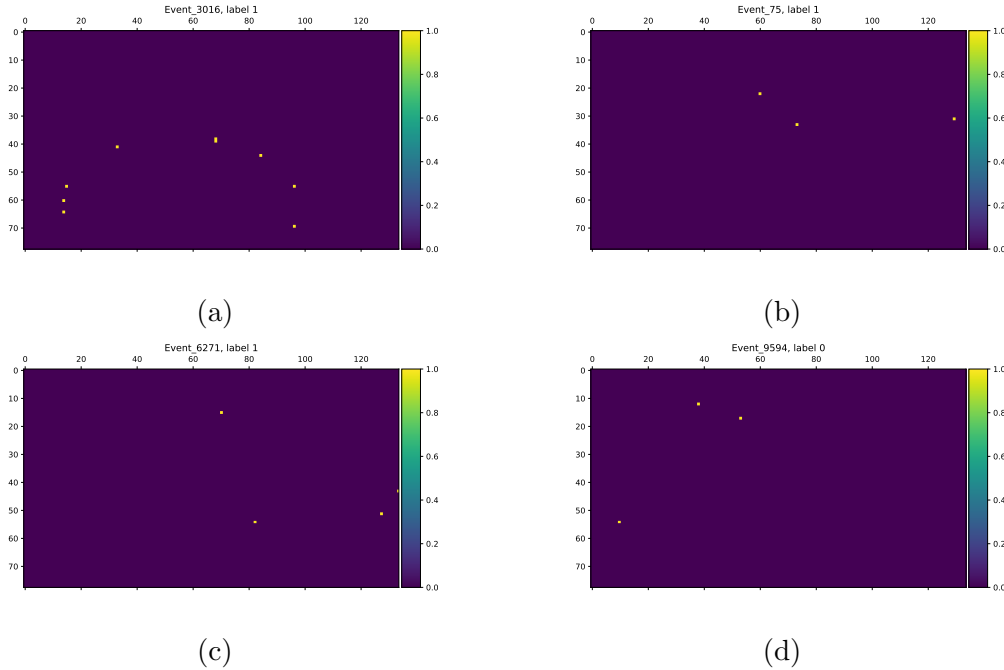


Figure 4.31: Examples of the 2D dataset. Element $(0, 0)$ is in the upper-left corner. The input of Fired PMTs emerge as yellow points on the blue background. Note that this image is directly injected into the algorithm, without reshaping. The event labeling is the same of the previous studied case, 1 if the radius is in $[185, 195]$ mm, and 0 otherwise.

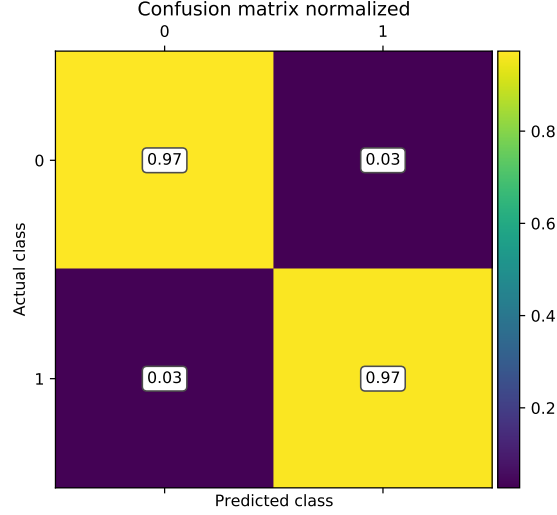


Figure 4.33: Confusion matrix on the test dataset (i.e. generated with a seed different from the training one). The matrix is normalized per row.

Fixed single ring dataset, random radius

This is the simplest ideal case which, as before, is used as a first check. The performances are very similar to those of the DNN both in the training (see Figure 4.32)

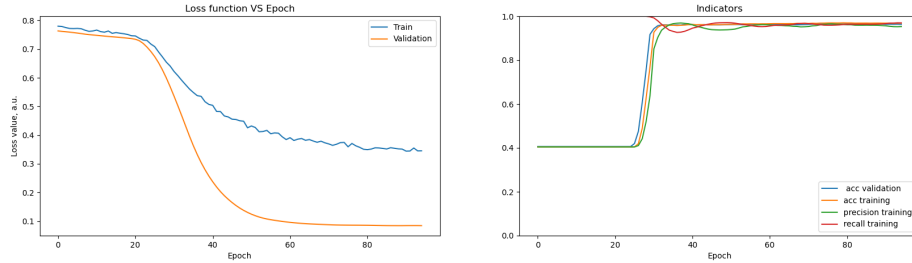


Figure 4.32: Results after training the network on the fixed-center 2D dataset. Since the task is "simple" the network rapidly saturates towards $\sim 99\%$ accuracy. In the plot on the left, the validation loss is always below the training one because of the implementation of dropout layers during the training.

and in the testing phase (Figure 4.33). Also the PMT occupancy plot for the test set indicates that the good results are not generated by random guesses, but the CNN is basing its prediction using the information coming from the PMT position in the grid (see Figure 4.34).

Fixed X, random Y center coordinate, single ring dataset, random radius

Introducing the variability of the y coordinate of the center, the performances starts to degrade as expected (see Figures 4.35 for the training and Figure 4.36 for the testing), but with a remarkable improvement in the misclassification of false positive/negative with respect to the DNN case, using the same size for the training

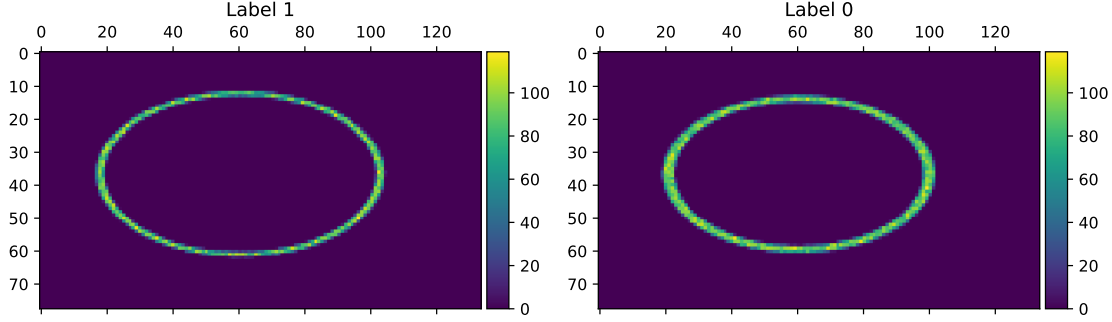


Figure 4.34: PMT occupancy plot for events of the test dataset. If the event is predicted as signal (or 1) by the trained CNN, the plot on the left is filled with the fired PMTs, while if the prediction is background (or 0) the plot on the right is filled. It's possible to see that when the CNN prediction is signal (background) the PMTs are more displaced outside (inside). Moreover the halo of the right plot is wider than that of the left plot since the background range ($[170, 185]$ mm) is wider than the signal range ($[185, 195]$ mm).

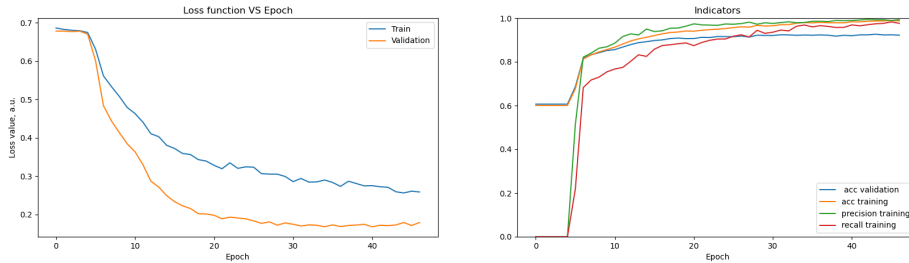


Figure 4.35: Training results on the fixed x 2D dataset. The loss drops abruptly in a way similar to the fixed-center 2D dataset.

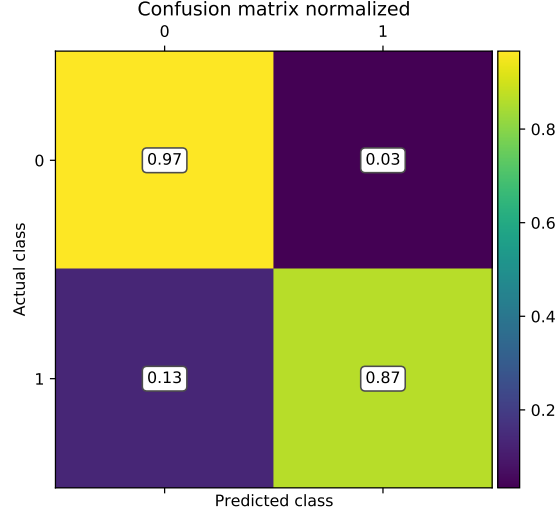


Figure 4.36: Confusion matrix for the fixed x 2D dataset. The fraction of correct classifications (true positive and true negative) increased with respect to the analog DNN classification on the same dataset category (the number of training events is the same).

dataset (10^4 events). It's interesting to compare this results with the corresponding 1D case in terms of the feature importance: looking at Figure 4.15 and 4.37 we can see how the edges again play the most important role for the prediction.

Random X, fixed Y center coordinate, single ring dataset, random radius

We saw earlier from the PMT occupancy plot 4.19 that this task is in principle more difficult with respect to the fixed x dataset and this has been confirmed by the decreasing discrimination power of the DNN. It's interesting to see that with a CNN approach the results on this dataset are equal to the fixed x one. This can be explained by the fact that the network is focusig on the position of the PMTs rather than on the full PMT occupancy, leading to a better discrimination power. Here we report just the confusion matrix of the trained model on this dataset (Figure 4.38), which shows the improvements in using a CNN for the purpose of ring classification.

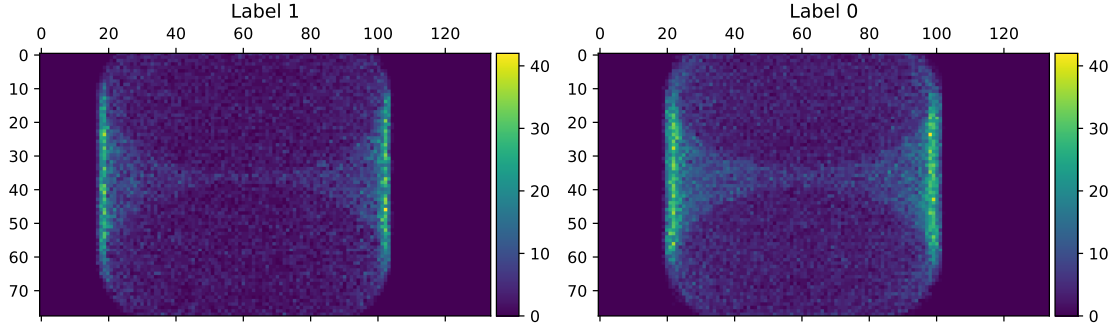


Figure 4.37: PMT occupancy plot for events of the test dataset, with fixed x position of the center. If the event is predicted as signal (or 1) by the trained CNN, the plot on the left is filled with the fired PMTs, while if the prediction is background (or 0) the plot on the right is filled. The plots show that events with hits on the edges are more likely classified as signal; the opposite for events with hits displaced more towards the innermost part of the grid. The x coordinate of the center is kept fixed so the occupancy results only in a portion of the whole grid.

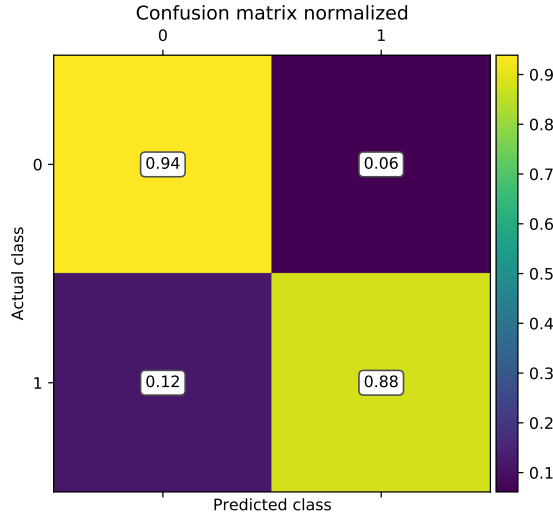


Figure 4.38: Confusion matrix of the CNN on the test dataset with fixed y position of the ring center. In this case the important point is that results are symmetric wrt the fix x dataset because the CNN focuses on localities.

Random center coordinates, single ring dataset, random radius

Despite the increasing performances with respect to the fully connected DNN algorithm, the results on the randomized xy center position dataset are still not satisfactory. Figure 4.39 shows the confusion matrix of the trained CNN on the test

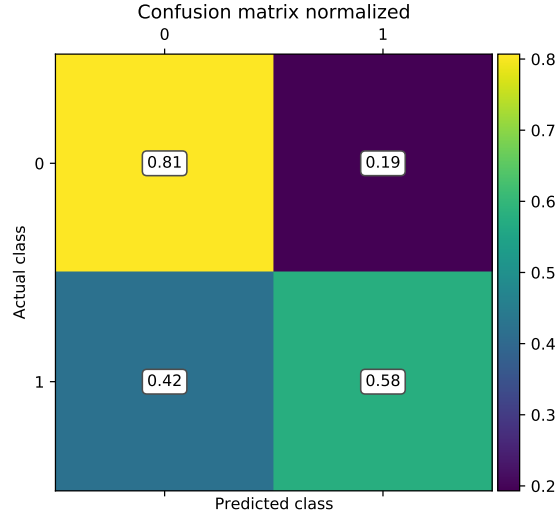


Figure 4.39: Confusion matrix generated by applying the trained CNN on the test dataset. Results are slightly better than the corresponding problem handled with the fully connected DNN, but still not satisfactory.

sample. In particular, the false positive fraction is reduced by the CNN but only by $\sim 8\%$. From the feature importance plot 4.40 one can see that the CNN has no way to disentangle the two classes.

Convolutional Neural Network classifier on Hough preprocessed events

From the results of the previous studies we learned that trying to extrapolate the labeling of the event by directly applying the DNN algorithms on the PMT spots (either in the 2048 1D vector or the 78×134 2D matrix) is such a complex task that performances are very poor even in the ideal case of single ring events without noise. Therefore we decided to try another approach involving the Hough transform as a preprocessing stage before feeding the instances of the dataset into the neural network. The Hough transform method is based on the fact that if the points belonging to a ring of radius R are used as centers for drawing circles of radius R , such circle will all cross in the original ring center position (see Figure 4.41). In our application, points belong to a grid and the center position can be recovered as the position of the bin which is most populated when all the additional "Hough circles" are used for filling each bin over which they pass. A coupling of a DNN with "Hough" preprocessing can be based on the idea that DNNs could be a sort of complex maximum detector. As we will show in the following, it turned out that this approach is not promising due to the effects of discretization causing:

- shift of the true hits position (Figure 4.42);
- shift of the hits of the drawn "Hough circles".

Since the network should be trained to classify correctly the maximum of the a grid, a reasonable choice could be to label the events in the preprocessed "Hough" dataset such that:

- 1 if $N_{max} \geq n_{hits}$

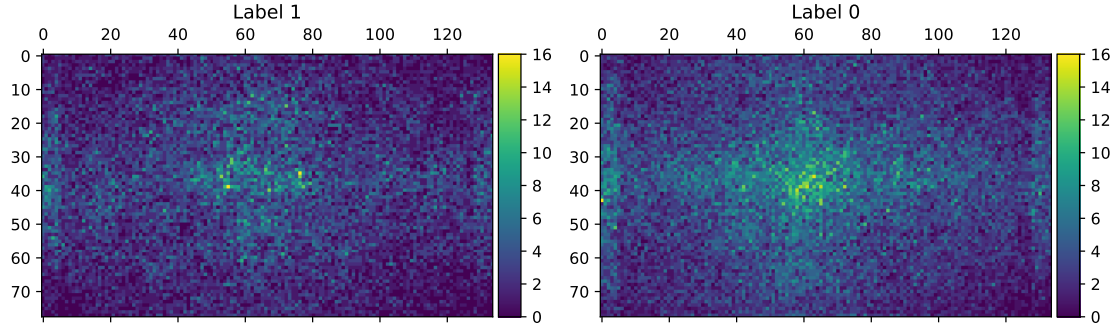


Figure 4.40: PMT occupancy plot for events of the test dataset, with fixed random position of the center. If the event is predicted as signal (or 1) by the trained CNN, the plot on the left is filled with the fired PMTs, while if the prediction is background (or 0) the plot on the right is filled. The plots show that the CNN cannot find hit configurations to disentangle the two classes, as in the cases where at least one coordinate of the center is kept fixed.

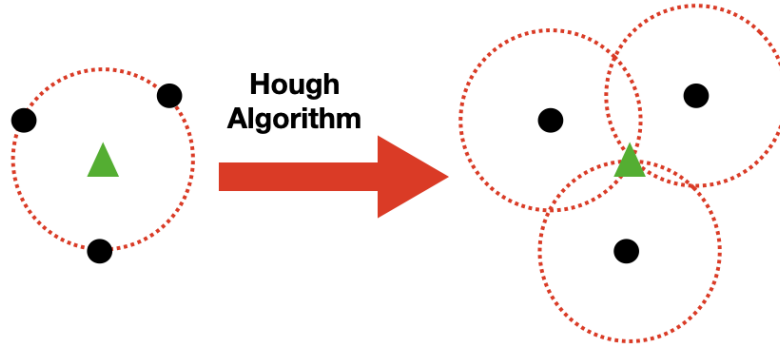


Figure 4.41: Example of the Hough preprocessing method in an ideal case.

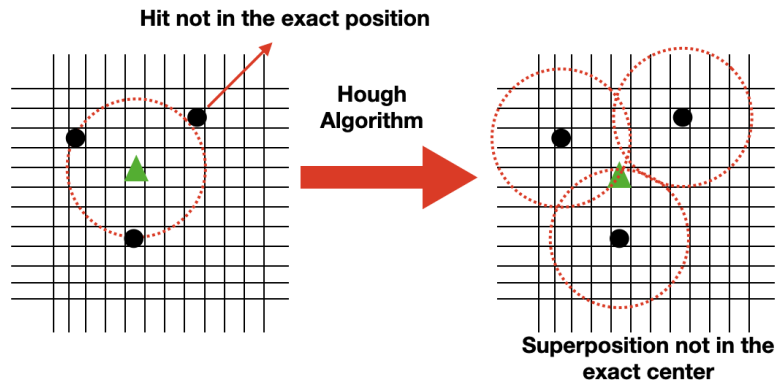


Figure 4.42: Example of the Hough preprocessing method in a real case. The shift of one hit of the true ring due to the discretization leads to an accumulation point with only 2 rather than 3 hits.

- 0 if $N_{max} < n_{hits}$

where N_{max} is the number of entries in the most populated bin and n_{hits} is the number of hits of the ring in the event. We study the effects of this convention labeling on the fixed-center dataset described earlier, for three different discretization, so called:

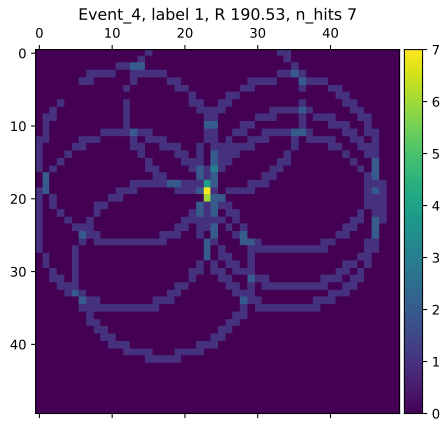
- RICH discretization (i.e. grid dimension as in the CNN study, see Figure 4.43e)
- 50×50 grid
- 130×130 grid.

For each event, we get the fired hits and we plot the "Hough rings" around each of them. Since we want to use only one radius per event, we select the value of 192 mm, after squeezing the range value for the definition of a signal event into [189, 195] mm. The obtained dataset is then relabeled as signal or background according to the previous convention. This relabeling means that the discretized Hough procedure might change the labeling of an event, in case a real ring results in all Hough circles not overlapping in a single bin, or an uncorrelated set of hits results in a bin being filled accidentally over the threshold. For example Figure 4.43c, 4.43d are respectively a false negative and a true positive because the original labels (1 in both cases, as indicated at the top of the Figure) are replaced by the "Hough" labels 0 and 1. The result of the study is shown in Figure 4.44. The plot is normalized per rows so that for each N_{hits} sub sample the sum is 1. One can see that background events are mapped correctly into background events, but many signal events are mapped into background events. The training phase would be obviously biased by this dataset, leading to very low performances in classification.

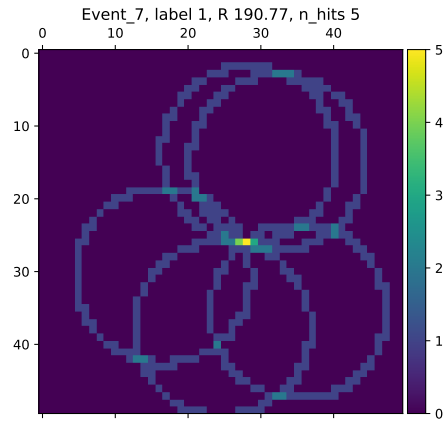
Another possibility is to use a CNN algorithm to test if the pattern with which the Hough circles overlap is distinctively different for correlated hits (from the same ring) or uncorrelated hits (from different rings). We built the dataset starting from images like the ones shown in Figure 4.45a (the level of discretization will be described in the following). Since we want that the network analyzes only the pattern around the maxima, which are the point where for correlated hits we expect the position of the interesting pattern, we select a window of predefined side and we set to zero all other bins, storing this as an element of the training dataset. Since in the original event more than one maximum can be present, more than one figure can be created with the same event number. The labeling of the event is based on the criterium that if the window contains the (mapped) position of the original center, then the image is considered as signal (or 1), otherwise it is considered as background (or 0). An example of generation is shown in Figure 4.45. We performed 9 studies, considering all the combinations of 3 possible grid discretizations (50×50 , 100×100 , 150×150) and 3 cut windows (3×3 , 11×11 , 21×21). The CNN was trained and tested on each dataset independently. Results are summarized in Figure 4.46 in the form of a confusion matrix for each tested combination (test datasets were generated with a different seed with respect to the training one).

Conclusions

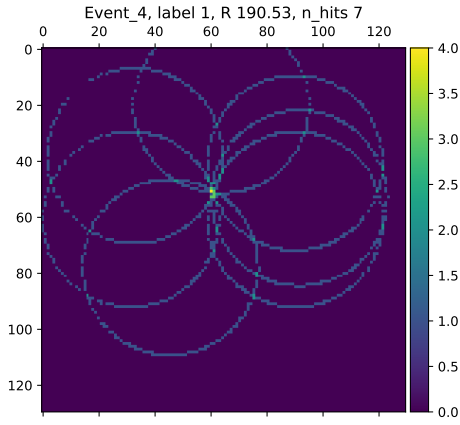
In this section we showed that the use of a CNN classifier for disentangling signal and background events improved the efficiency of the classifier with respect to the



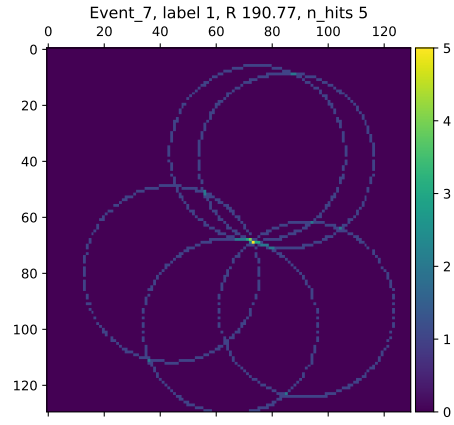
(a)



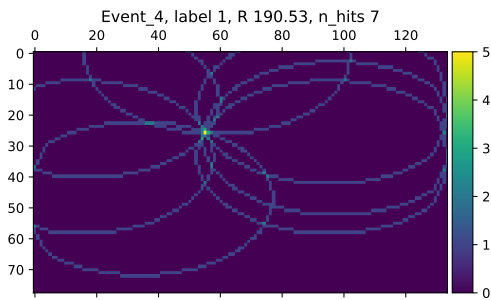
(b)



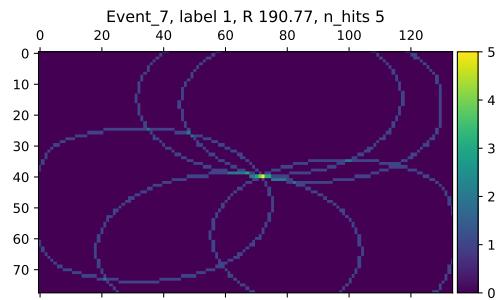
(c)



(d)



(e)



(f)

Figure 4.43: Same event pair of event preprocessed with a different grid during the application of the Hough algorithm. Hits from the original events are removed from the image. Discretization grid values: 4.43a, 4.43b 50×50 , 4.43c, 4.43d 130×130 , 4.43e, 4.43f RICH.

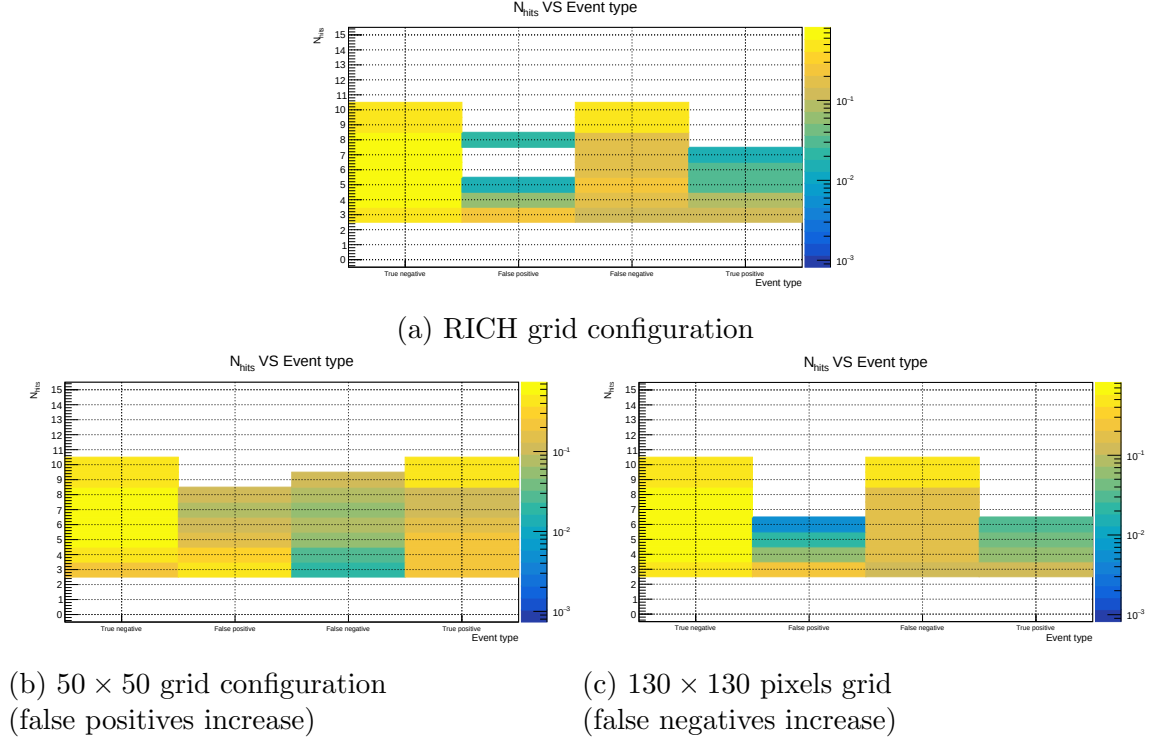


Figure 4.44: Results of the mapping between the standard dataset and the Hough one. Depending on the discretization level, the number of false positive/negatives increases.

fully connected DNN model. We also tested the possibility of implementation of the Hough preprocessing technique, and looked into the possibility of classifying events according to the pattern near the generated maxima. In both cases we found that the low efficiency of the CNN was mainly due to the dataset composition, rather than the model itself. We also performed minor tests increasing the dataset dimensions, in this case limited by the saturation of the available RAM capacity of our PCs, problem emerging due to the high grid discretization.

4.5 Autoencoders

We showed in the previous sections that the implementation of a neural network classifier capable to directly predict the (signal or background) class of an event didn't lead to satisfactory results. Probably the main reason is that we tried to solve with a single method a composite problem, i.e. ring detection+ring classification. Indeed we were asking an algorithm to classify rings from points, but requiring the algorithm to elaborate a rule to define what a ring is and then use that rule to classify rings. Considering the wide variability of the events in terms of differences in hits, center position, radius etc. the tested models failed in finding a definition satisfying all the possible cases. Some improvements might be expected by increasing the size of the dataset including all the possible events that can occur in the RICH (more than one ring per event, noise hits etc.). However we decided not to pursue such approach since, besides the need to address technical issues (more sophisticated generator, very high number of generated events, etc.), we would only have created

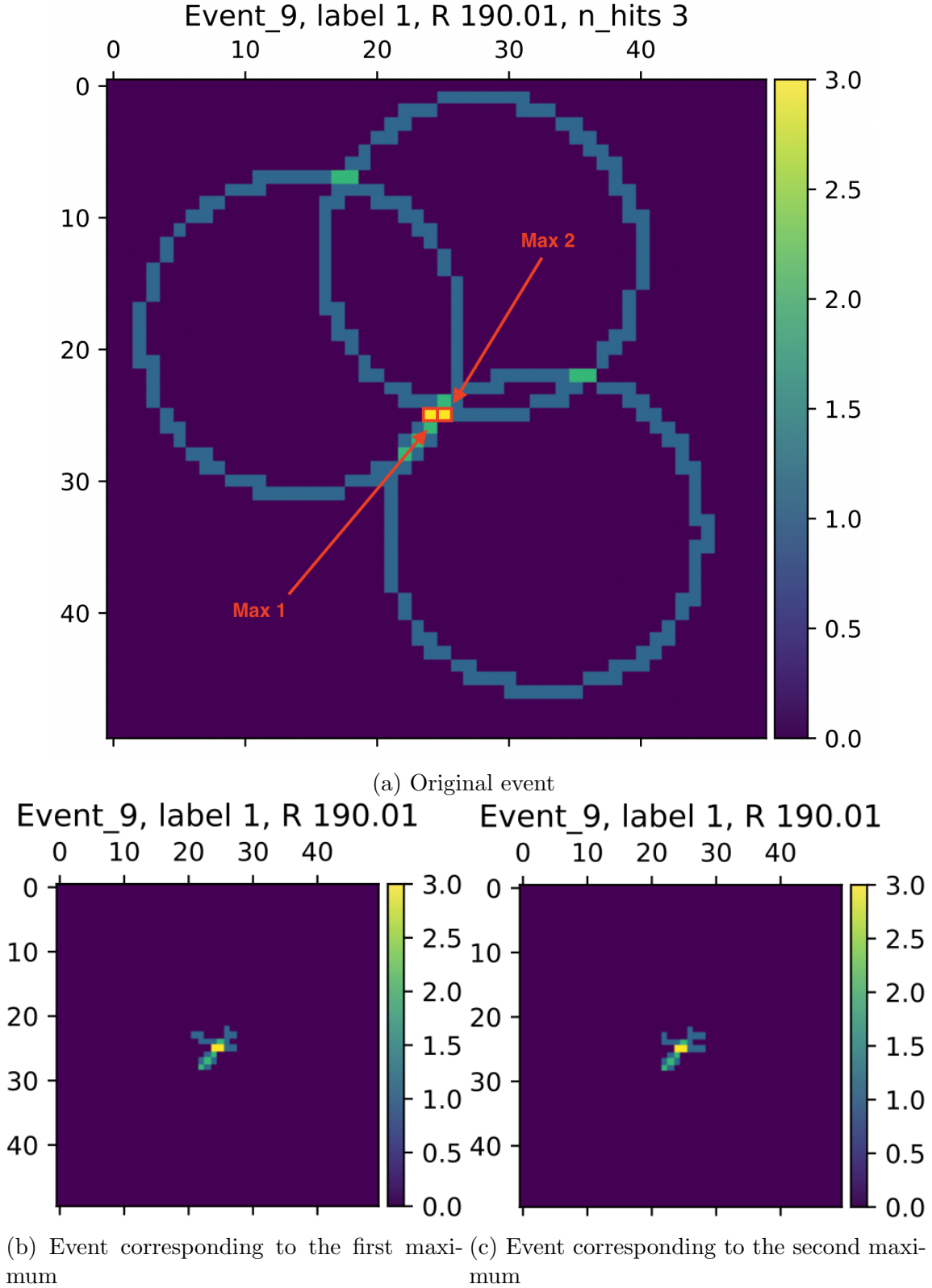


Figure 4.45: Example of the dataset generation. From the same original event, multiple sub-events can be generated. For example from the event in (a), mainly due to the discretization of the grid, 2 maxima are present, so there are two generated sub-events (b) and (c), both with label 1.

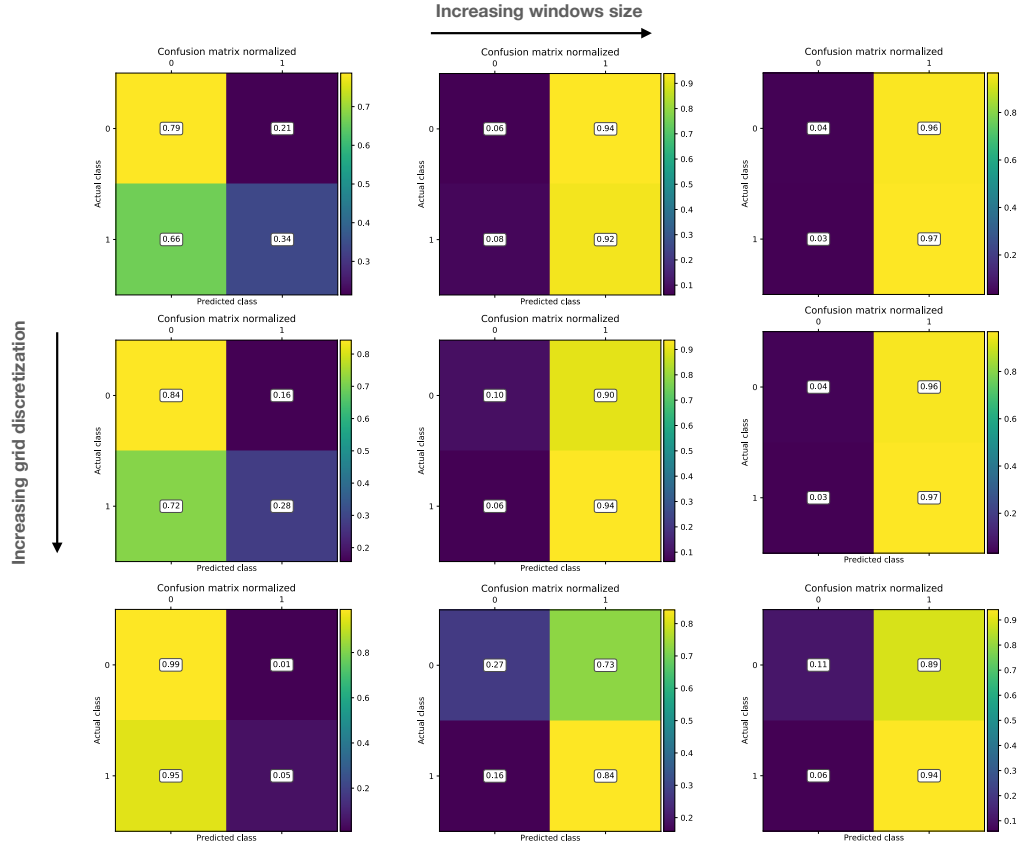


Figure 4.46: Confusion matrices showing the efficiency of each trained CNN in recognizing the pattern around the maximum where the Hough circles overlap. In the top left corner is shown the result on the dataset with grid discretization equal to 50×50 and windows size equal to 3×3 . In the horizontal axis the grid discretization is constant, while the size of the cut window increases. In the vertical axis the cut window is fix and the grid discretization increases. Considering the first column, corresponding to a 3×3 window cut around the maximum, we can see that the majority of events are classified as background (class 0) even if they are signal (class 1). This is because the network was trained with a biased dataset where the majority of instances belonged to class 0 (background). The unbalancing is due to the smallness of the cut window (3×3) that in the majority of the cases doesn't include the center of the ring in the presence of a maximum. When the grid discretization increases, the effect is accentuated since many maxima are generated. The second column and the third show the opposite trend, with all the events classified as signal. Since the window is 11×11 in the second column and 21×21 in the third, the biasing is now towards the signal class because when a maximum is found in the image the corresponding window is big enough to include very often the real center position. As before, the grid contributes increasing this effect since many spurious maxima are found and put in class 1.

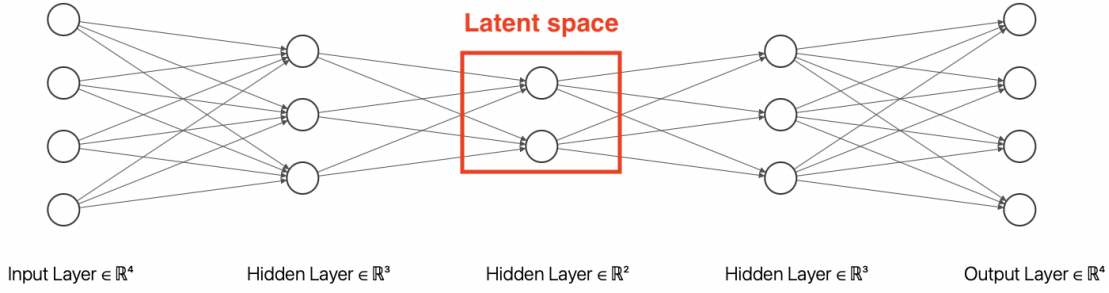


Figure 4.47: General sketch of an Autoencoder model. The model is required to reproduce its own input using only a reduced representation of it, stored in the "latent space". This purpose is reached by building a network with the output layer structure equal to the input one, and with a number of neurons in the latent space smaller than the dimension of the input data.

a model capable to efficiently look into a look-up table.

We rather decided to change our approach, dividing the global problem of event classification into two separate sub-tasks: ring hit clustering, performed using deep learning, and ring fitting (+classification), performed with analytical techniques. The main idea is that a deep learning technique for clustering should overcome the long time required by an analytical solution to classify which hits of the event should be fitted together (see [32]). This such time obviously grows with the number of hits. So our approach would be to

1. submit an event with N hits to the network;
2. (analytically) fit the hits which the network considers to be part of a ring (using a modified version of the Crawford fit documented in the Appendix B.1);
3. submit the same event to the network with only $N - M$ hits, where M is the number of hits lying within a certain distance from the previously fitted ring;
4. go to step 2 and repeat, until $N \leq 3$ or no hits are removed (when the network always provides the same answer to the same input, generating infinite iterations).

The approach used for clustering hits was to require that the network assigns to each hit of the event a weight, representing the likelihood that it belongs to a certain ring. Training a network to assign such weight means in some way that the network should be able to catch the concept of "ring". Generally this kind of problems can be solved using Autoencoders, fully connected neural networks that are used mostly for solving the "dimensionality reduction" problem, that is to detect the most important features of the input data. To perform such task they are forced to reproduce the input they are presented to, but starting from a reduced number of information. This is technically obtained with structures like the one shown in Figure 4.47. For example, Autoencoders are generally employed as a technique for noise reduction in images.

Since in the input we have partial rings (with a discrete number of hits), our idea was to feed into the network single-ring events with the fired hits, and require the

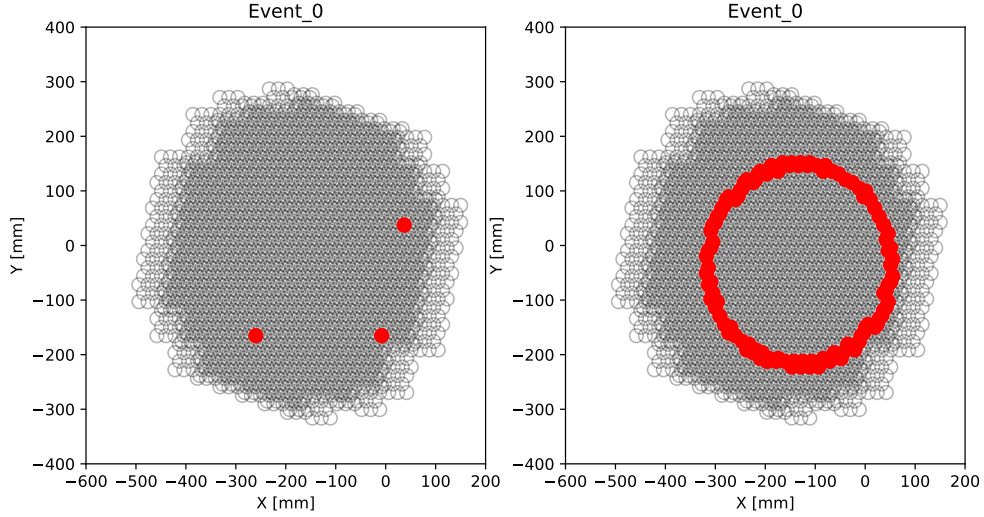


Figure 4.48: Example of the events characterising the dataset used in the Autoencoder study. Left: input with the "real" hits of the event. Right: required output with all the PMTs belonging to the ring fired.

network to predict as output the full ring, as if all the PMTs belonging to it would have fired. An example of the dataset is shown in Figure 4.48. The event is injected into the network as a 1D vector as for the fully connected DNN classifier previously described. The network structure is shown in Figure 4.49. We used a ReLU activation function for all the layers, also for the output one, because we want a continuous value to be interpreted as a weight. We tuned the model parameters using the Mean Squared Error Loss (MSELoss) defined as:

$$MSELoss(\mathbf{x}, \mathbf{y}) = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^{2048} (x_{ij} - y_{ij})^2 \quad (4.8)$$

where $\mathbf{x}$ and $\mathbf{y}$ are matrices with N rows and 2048 columns, each column representing a single PMTs; N is the number of rings (events). We used also in this case Dropout layers, to increase the generalization capabilities of the network.

We performed several studies concerning the size of the input dataset, measuring the efficiency of the network as a function of the training dataset size. The size used so far (8000 events) generates meaningless outputs, as shown in Figure 4.50. By increasing the size to 248000 events, the highest value for the efficiency on the single-ring test set was $\sim 82\%$. We show in Figure 4.51 the same event of Figure 4.50 processed after the new training. At this point, we applied our complete algorithm (i.e. including the iterative procedure for removing the fitted hits) to multi-ring events (10000 events with one, two and three-rings). A multi-ring event is classified true signal event if at least one true (reconstructed) ring has a radius falling in the electron range [185, 195] mm. We show in Figure 4.52 the application on two-ring events and in Figure 4.53 the application on three-ring events. The results are summarized in the confusion matrices shown in Figure 4.54. We would have expected a higher performance especially in the 1-ring test since in such a case there is no ambiguity on the hits to be fitted. However the problem in this case is the fit procedure rather than the Autoencoder. Results on multi-ring events are also

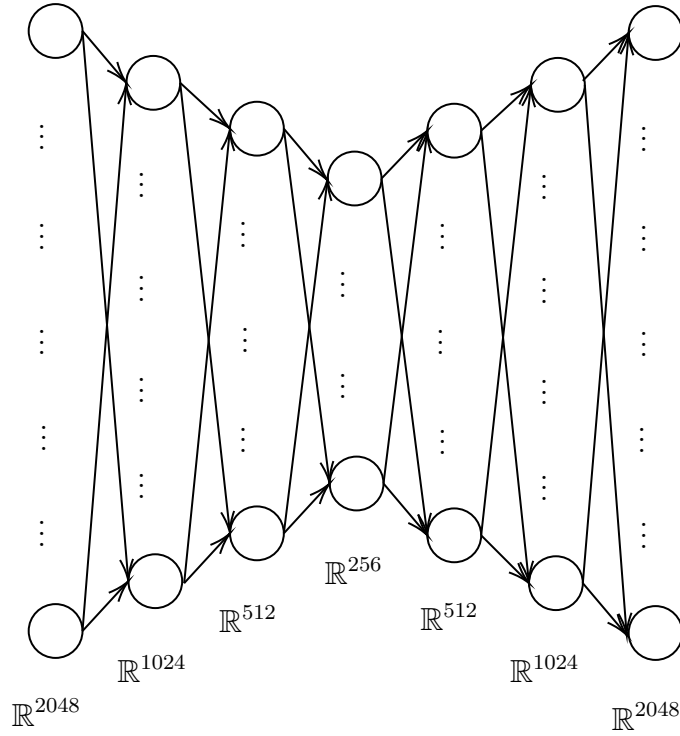


Figure 4.49: Scheme of the Autoencoder model used in the analysis. Each layer is implemented as a 1D vector with dimension $\mathbb{R}^N$.

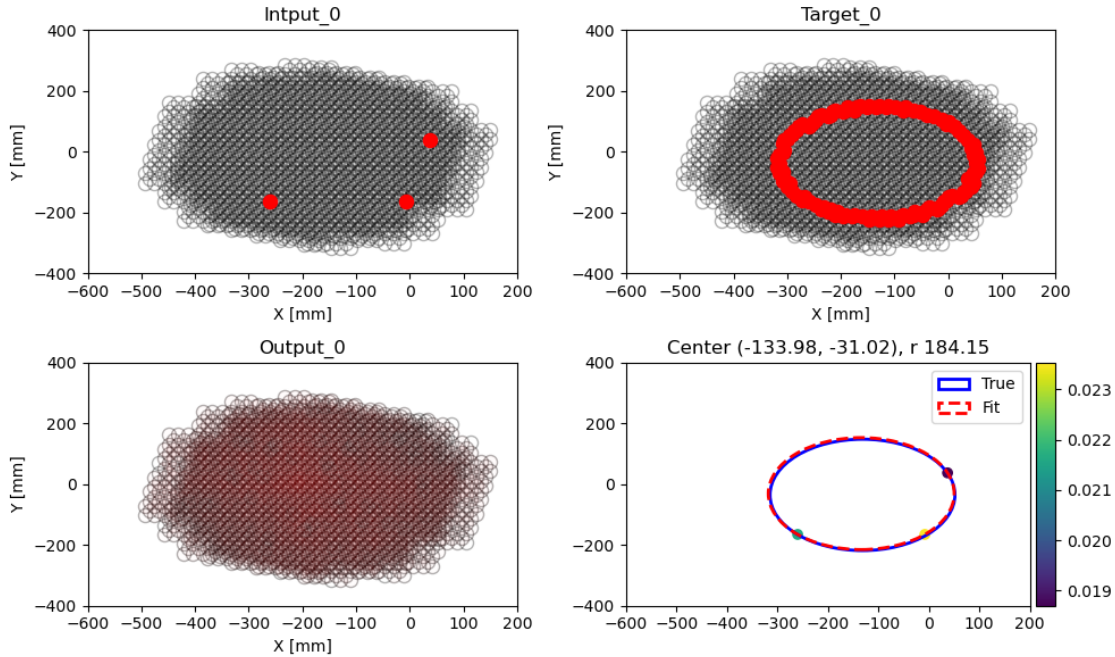


Figure 4.50: Result of the application of the Autoencoder to the first event of the test dataset. Top left: input event. Top right: the target output the network should try to reproduce. Bottom left: what the network outputs given the input. Bottom right: The fit performed on the hits of the event given the weight the network assigned to them (in the title are shown the fitted values). The Autoencoder was trained with 8000 single-ring events. The bottom left plot shows that this size of the training dataset is not enough.

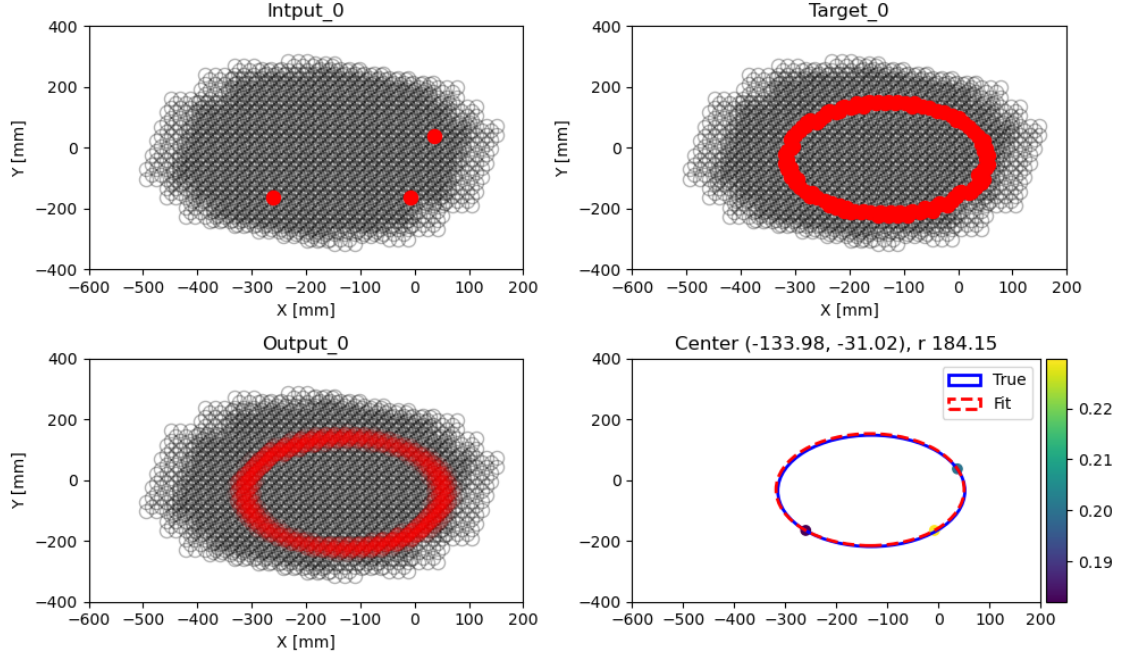


Figure 4.51: Result of the application of the Autoencoder to the first event of the test dataset. This time the Autoencoder was trained on 248000 events. Top left: input event. Top right: the target output the network should try to reproduce. Bottom left: what the network outputs given the input. Bottom right: The fit performed on the hits of the event given the weight the network assigned to them (in the title are shown the fitted values).

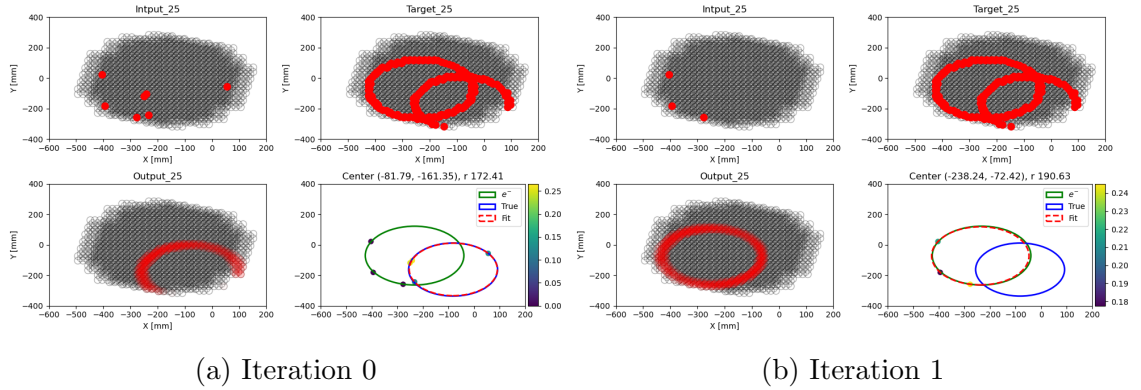


Figure 4.52: Example of application of the trained Autoencoder on a two-ring event. The Autoecoder is given the input shown in the top left part of Figure 4.52a, for which it predicts the shape shown in the bottom left part of Figure 4.52a. A Crawford fit is performed and all the hits within 10 mm of the fitted ring are removed from the event. The pruned event (top left of Figure 4.52b) is passed again to the network, which predicts the halo shown in the bottom left of Figure 4.52b. We then perform the Crawford fit again on the weighted hits. In this case the processing of the event finishes because all the hits have been used.

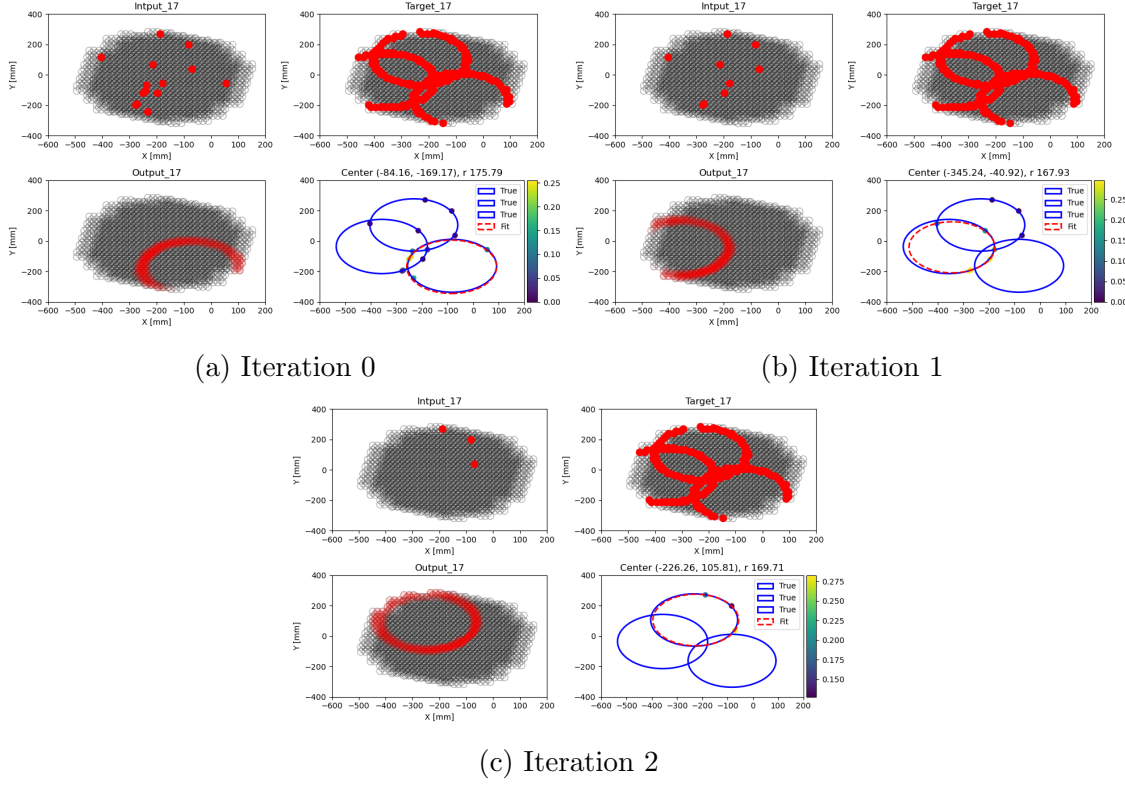


Figure 4.53: Example of application of the trained Autoencoder to a three-ring event. The Autoencoder is given the input shown in the top left part of Figure 4.53a, for which it predicts the shape shown in the bottom left part of Figure 4.53a. A Crawford fit is performed and all the hits within 10 mm of the fitted ring are removed from the event. The pruned event (top left of Figure 4.53b) is passed again to the network, which predicts the halo shown in the bottom left of Figure 4.53b. We then perform the Crawford fit on the weighted hits and subsequently remove the hits. The event is newly injected into the network (top image of Figure 4.53b) and the network outputs the halo of bottom left of Figure 4.53b. Fitting the hits with the given weights and removing them from the event completes the process.

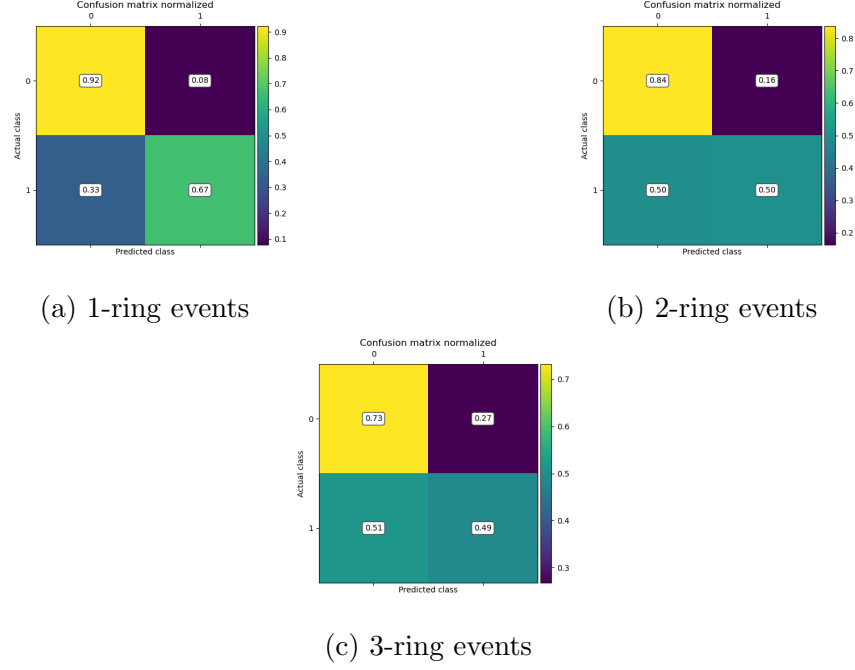
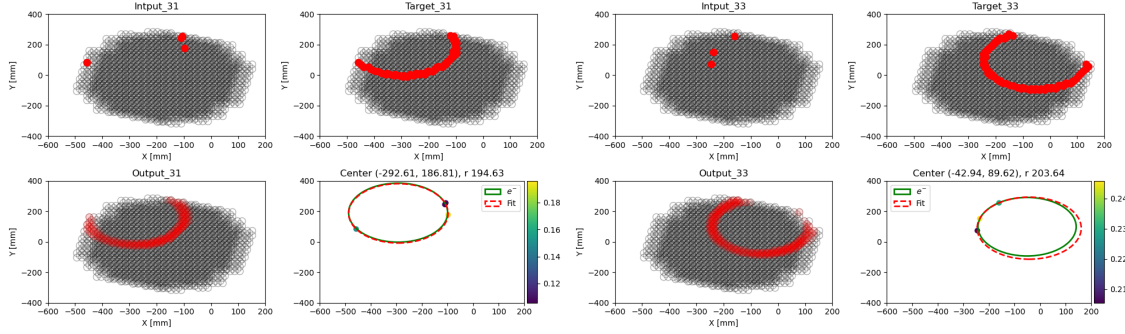


Figure 4.54: Confusion matrices showing the performance of the composite event classifier (Autoencoder+fit+classification) on (a) single-ring, (b) 2-ring and (c) 3-ring events. As usual the matrix is normalized per row.

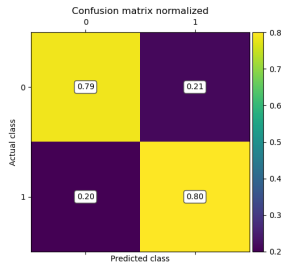
affected by such problem. For single-ring events (Figure 4.54a), on which we would have expected a very high accuracy, we obtain only $\sim 82\%$. By looking at the prediction of some events of this dataset we see that one of the possible areas of improvement could be the use of a different fit algorithm, since the Crawford one seems to perform poorly on events with an asymmetric hit disposition (e.g. hits only on half of the ring etc.) as shown in Figure 4.55.

Despite this problem of the single-ring fit, our final study concerned the test of the classifier on events in which the hits simulate the response of the RICH to specific K^+ decays, and the "true" rings are those recognized and fitted using the NA62Reconstruction package. We used 10000 events from four decay modes: $K_{2\pi D}$, $K^+ \rightarrow \pi^+ e^+ e^-$ ($K_{\pi ee}$), $K_{3\pi}$, K_{e3} . Results are shown in Figure 4.56 in the form of confusion matrices. The overall efficiency for these events ranges from a minimum of $\sim 75\%$ for the $K_{3\pi}$ sample to a maximum of $\sim 80\%$ for the $K_{2\pi D}$ one, that is far from the $\geq 90\%$ efficiency required for an algorithm to be implemented in the trigger. As a final remark, comparing the better results obtained on the physical events (Figure 4.56) with respect to the ones obtained on our multi-ring generated events (Figure 4.54), we concluded that maybe our Autoencoder was trained on more complex event than those that are normally generated in the RICH. Moreover we think that maybe the efficiency on the physical datasets could be increased somewhat. Indeed we found that in some events the ring generated by the Autoencoder was actually better in matching the position of the hits in the event with respect to the "true" ring, identified by the NA62Reconstruction package.

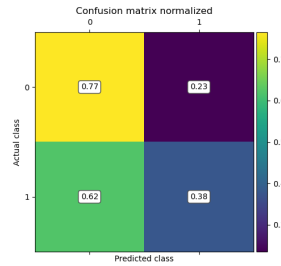


(a) Signal event with symmetric hit positions (b) Signal event with symmetric hit positions

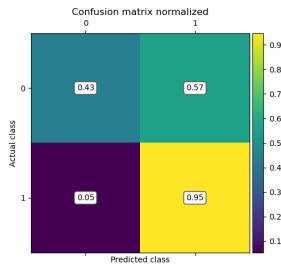
Figure 4.55: Example of the problem related to the fitting procedure for events belonging to the signal class (electron rings). As shown before, in the top left image is the input event, in the top right image the target ring, in the bottom left the weights halo produced by the network from the input and in the bottom right the Crawford fit. Note that in (a) the fit performs well and the event is classified as signal (since the fitted radius is 194 mm), while in (b) the event is put in the background class. The problem is in the fit algorithm and not in the Autoencoder output.



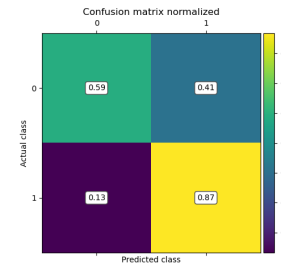
(a) $K_{2\pi D}$



(b) $K_{3\pi}$



(c) K_{e3}



(d) $K_{\pi ee}$

Figure 4.56: Confusion matrices, normalized per row, showing the result of the application of our composite classifier for the four tested K^+ decay modes.

4.5.1 Conclusions

We presented a possible implementation of an composite event classifier involving an Autoencoder for the hit clustering. While the algorithm doesn't reach yet performances adequate to an online implementation we identified some areas for improvement:

- change of ring fitting algorithm
- event generator, to generate events more similar to the real ones
- the tuning of the network structure

We consider this as the most promising approach among the described ones, and also the one with the most interpretable results.

Chapter 5

Perspectives of an automatized study of $\pi^0 \rightarrow e^+e^-\gamma$ decay through deep learning techniques

5.1 Introduction

In this Chapter we describe our efforts in the implementation of a Deep Learning algorithm able to classify signal and background events with respect to a specific analysis, using only information about reconstructed quantities from the detectors, and only few assumptions on the processed data. In the analysis sector, our approach represents something new since DL algorithms are mostly employed as automatic PID algorithms, able to merge quantities of different detectors (generally tracks from spectrometers and clusters from calorimeters) to infer the particles type. The comparison between “standard” and automatized selections will be made using the $\pi^0 \rightarrow e^+e^-\gamma$ analysis described in Chapter 2 as a benchmark.

We used a neural network model, like the one described in section 4.1, implemented using the Pytorch framework. The specific features of the model (e.g. number of neurons, number of layers, activation functions) changed during our study and are detailed later.

5.2 Datatest design

As we have already described in Chapter 4, the data preparation phase is one of the most critical in a ML project. The selected input quantities can determine the success or the failure of the model ability in generalization, i.e. ability to correctly infer the class of an event which has never seen.

First of all, MC events have been chosen for this feasibility study because the label of each event must be known. The employed lists are those described in the standard selection (section 2.4.2), excluded the biased $K_{2\pi D}$ sample, and without the overlaid option. We opted for the not overlaid MC samples because of the difficulties in the association between true and reconstructed quantities, which in this preliminary study would have required a little effort to design reliable algorithms for most of the available reconstructed quantities¹. Since the trained model is thought to be

¹As we described in Appendix A.1, such an algorithm is to date implemented only for tracks.

applied on data, an essential assumption is that MC should reproduce very well the data characteristics. Hence the employed MC events are required to pass the same filter algorithm which is present in the data. Without any reference in this kind of study, the selected input features we decided to store are the majority of those reconstructed by the detectors that are also used by the standard approach. Referring to the described Dalitz selection, in particular from section 2.4.3 to 2.4.7, this means storing quantities from: CEDAR, Spectrometer, CHOD, NewCHOD, LKr. High level quantities, like Lorentz four-momenta or three-momenta have been unpacked and stored through their components, so basically the algorithm doesn't know how they are linked (i.e. it doesn't know that $p = \sqrt{p_x^2 + p_y^2 + p_z^2}$). Some basic geometric properties (like spectrometer beam hole position, Z position along the beam line etc.) of the considered detectors are also included. At this point the dataset design would in some way reflect the ML model we want to use to perform the study. The use of a neural network imposed a static number of neurons in the input layer, leading to the necessity of fixing the number of input features we could give to the network for training and testing. Once we established the physical quantities we wanted to use for the study, considering that the number of candidates of different detectors in the same event and number of candidates of the same detector but in different events can vary, fixing the number of input quantities translated in the necessity to fix the number of maximum storable candidates for a each detector: we started with 10 candidates for each quantity. This cut doesn't differ to much with respect to a "standard" analysis in which events with a relatively high number of events are generally rejected. We selected 73 quantity of interest, like STRAW track momentum, position of LKr clusters etc. for a total of 730 input feature. The dataset structure is shown in Figure 5.1. The fixed number of candidates per event

$P_{track}^{(0)}$	...	$P_{track}^{(10)}$	$\chi^2 \text{ Track}^{(0)}$	...	$\chi^2 \text{ Track}^{(10)}$	$E^{(0)}$	...	$E^{(10)}$	...
1									
2									
$\vdots$									
N									

Figure 5.1: Dataset structure in the initial phase of our study. Each column represent an input quantity while each row a different instance (or event).

imposed to establish a convention for those events with less than 10 candidates in an event. We established the absence of a value with 0 (this technique is usually know as *padding*). For the output layer we used 9 classes, the signal plus the simulated decay channels in Table 2.1. We implemented Sigmoid functions, defined as

$$\text{Sigmoid}(z_k^l) = \frac{1}{1 + e^{z_k^l}} \quad (5.1)$$

for the hidden layers of the model, while we used NLLLoss (see 4.5) with the number of classes to be classified $C = 10$ for the output layer. In (5.1) we used the same formalism adopted in section 4.4.1. The use of Sigmoids made necessary to scale the input features in the $[0, 1]$ range to avoid the saturation of the neurons. The

rule applied for input feature i was

$$x_i^s = \frac{x_i - \min}{\max - \min} \quad (5.2)$$

where $\min$ and $\max$ are the minimum and maximum values for the i th feature. We used fixed reference values for the maxima and minima, according to the specific quantity. For example for geometrical quantities they correspond to the geometrical limit of a specific detector. For other quantities, like momenta or energies, we set a large cut of 90 GeV/ c and 100 GeV respectively. Quantities that were fixed to 0 in the dataset assembling were not modified. We performed various tests varying the model hyperparameters (learning rate, number of neurons etc.) but we always obtained a "saturation" of the model, expressed as a constant output value of the accuracy.

We opted for a redefinition of the dataset structure to overcome the difficulties of our first implementation, mainly the use of default values. In the new implementation we concentrated more on the topology of the event we want to identify, i.e. $K_{2\pi D}$ events (signal) among all the other decay channels (backgrounds). The cleanest detectable $K_{2\pi D}$ event is composed by three tracks and one "free" (not associated to any track) energy cluster. Hence, necessarily we have to consider the candidates from the STRAW spectrometer and the electromagnetic calorimeter. Moreover we can cut all events where $N_{tracks} < 3$ and $N_{clusters} < 1$, as it's generally done in the analysis to clean the sample from useless events.² Other reconstructed quantities from other detectors could be added later. To treat events where the number of tracks > 3 and the number of "free" energy cluster is > 1 we used a combinatorial method: we paired all the possible combination of 3 tracks

$$\binom{N_{tracks}}{3} \quad (5.3)$$

with all the possible combinations of 1 cluster

$$\binom{N_{clusters}}{1} \quad (5.4)$$

labelling signal the combination containing all the reconstructed candidates of the Dalitz particles. Here is an example with $N_{tracks} = 4$ and $N_{cluster} = 2$, with $track_0$ generated by the e^+ , $track_1$ generated by the e^- and $track_2$ generated by the π^+ and $cluster_0$ generated by the photon:

$$\begin{aligned} (track_0, track_1, track_2) + cluster_0 &\rightarrow signal \\ (track_0, track_1, track_2) + cluster_1 &\rightarrow background \\ (track_0, track_1, track_3) + cluster_0 &\rightarrow background \\ (track_0, track_1, track_3) + cluster_1 &\rightarrow background \\ (track_0, track_2, track_3) + cluster_0 &\rightarrow background \\ (track_0, track_2, track_3) + cluster_1 &\rightarrow background \\ (track_1, track_2, track_3) + cluster_0 &\rightarrow background \\ (track_1, track_2, track_3) + cluster_1 &\rightarrow background \end{aligned}$$

²To stress more about this, this cut is present at the level of the filter algorithm so it's not removable at all.

We found the implementation of such a dataset very flexible since, even in the case of many candidates in a detector, this would only add more combinations to the list. The generalization with the inclusion of other detectors is also straightforward: it only implies to combine all the combinations of candidates of the added detector with all the combinations already present. For example in the case we want to use two candidates from the RICH

$$\begin{aligned}
& (track_0, track_1, track_2) + cluster_0 + (RICHcand_0) \\
& (track_0, track_1, track_2) + cluster_0 + (RICHcand_1) \\
& (track_0, track_1, track_2) + cluster_1 + (RICHcand_0) \\
& (track_0, track_1, track_2) + cluster_1 + (RICHcand_1) \\
& \vdots \\
& (track_1, track_2, track_3) + cluster_1 + (RICHcand_1)
\end{aligned}$$

We tried to build such a dataset, with only spectrometer and electromagnetic energy clusters, finding that even processing the not overlay samples, produced a huge amount of combinatorial background, whose distributions completely overlaps the signal one. For this reason we decided to reduce more the complexity of the problem, selecting only events with exactly three tracks and one clusters, in such a way that the number of combinations in an event were reduced to one. The acceptance of events satisfying this condition are shown in Table 5.1 for each decay channel. The

Decay channel	Acceptance (%)
$K^+ \rightarrow \pi^+\pi^0, \pi^0 \rightarrow e^+e^-\gamma$	≈ 2
$K^+ \rightarrow \pi^+\pi^0, \pi^0 \rightarrow \gamma\gamma$	≈ 2
$K^+ \rightarrow \pi^+\pi^0\pi^0, \pi^0 \rightarrow \gamma\gamma, \pi^0 \rightarrow e^+e^-\gamma$	$\approx 10^{-3}$
$K^+ \rightarrow \pi^+\pi^0\pi^0, \pi^0 \rightarrow \gamma\gamma, \pi^0 \rightarrow \gamma\gamma$	$\approx 6 \times 10^{-4}$
$K^+ \rightarrow \pi^+\pi^+\pi^-$	≈ 5
$K^+ \rightarrow \pi^0e^+\nu, \pi^0 \rightarrow \gamma\gamma$	$\approx 7 \times 10^{-4}$
$K^+ \rightarrow \pi^0e^+\nu, \pi^0 \rightarrow e^+e^-\gamma$	≈ 2
$K^+ \rightarrow \pi^0\mu^+\nu, \pi^0 \rightarrow \gamma\gamma$	$\approx 9 \times 10^{-4}$
$K^+ \rightarrow \pi^0\mu^+\nu, \pi^0 \rightarrow e^+e^-\gamma$	≈ 3

Table 5.1: MC samples used in the study. The signal is highlighted in red. In the second column is shown the acceptance of MC events passing the cut of having only three tracks and one energy cluster in the event.

new dataset structure is shown in Figure 5.2 We took 1/3 of the whole dataset for the training and the remaining part for the test. We scaled our dataset as before, using the defined minima and maxima for each quantity. The distributions of the quantities we stored are shown in Figure 5.3. We report the results for the model reaching the best results after the tuning of the hyperparameters. We used a neural

$P_{track}^{(0)}$	$P_{track}^{(1)}$	$P_{track}^{(2)}$	χ^2 Track ⁽⁰⁾	χ^2 Track ⁽¹⁾	χ^2 Track ⁽²⁾	...	$E^{(0)}$	...	Label
1									1
2									0
⋮									⋮
N									0

Figure 5.2: Dataset structure based on the new combinatorial method. Each column represents an input quantity while each row a different instance (or event). Since we are fixing the detectors we wanted to use in the analysis (Spectrometer, Calorimeter) and the maximum number of candidates for each detector (3, 1), each Spectrometer quantity is present with index 0, 1, 2, while the only one cluster has always index 0. Moreover each line corresponds to a different event since only one 3-track, cluster pairing is possible. The last column indicates to which class the event belongs to: 1 denotes signal events, while 0 the background ones.

network with 36 input features:

$P^{(i)}, P_x^{(i)}, P_y^{(i)}, P_z^{(i)}$ Momentum of the tracks and its components,

$E_{track}^{(i)}, \chi_{track}^{2(i)}, Q_{track}^i$ Energy of the cluster associated to a track, χ^2 , Charge

$S^{(i)}x_{track}, S^{(i)}y_{track}$, track slopes at the vertex in the x and y coordinates

$x_{track}^{(i)}, y_{track}^{(i)}$ track position at STRAW chamber 1

$E^{(0)cluster}, x_{cluster}^{(0)}, y_{cluster}^{(0)}$, LKr cluster energy and position on the front face of the detector

where each quantity with index i is related to the track and so it is present with $i = 0, 1, 2$. We chose 6 hidden layers, with 10-7-7-5-5-2 neurons in each layer, and only one output layer. We also used ReLU activation for the hidden layers (defined in section 4.2), but the Sigmoid function for the output layer. This change was needed since the cost function we chose to evaluate the network training was the Binary Cross Entropy Loss, or BCELoss defined as

$$\text{BCELoss}(\hat{\mathbf{y}}, \mathbf{y}) = \frac{1}{M} \sum_{i=1}^M l_i, \quad l_i = -y_i [\log \hat{y}_i + (1 - \hat{y}_i) \log(1 - \hat{y}_i)] \quad (5.5)$$

which can be considered a NLLLoss with only two possible classes. BCELoss is well defined for classifications if the $\hat{y}_i$ are limited in $[0, 1]$ and the Sigmoid function is suitable for this. We used the Adam algorithm [31] for the minimization and we adopted *BatchNormalization* [30] and Dropout [27] layers: the former to speed-up the training phase, ensuring that the output of each hidden layer is re-normalized in the $[-1, 1]$ range before entering the subsequent layer, and the latter to have a better generalization power of the network. During the training the 8% of the sample is used as the validation sample to tune the parameters (see 4.2 for the definition of validation sample). The training of the network stops automatically when the loss function on the validation sample stops to decrease.

The result of the training is shown in Figure 5.4. The network reaches a $\sim 95\%$ of accuracy on the validation sample and this can be considered a good result since the training dataset was unbalanced with $\approx 23\%$ of signal events and $\approx 77\%$ of background ones. To check that our model is behaving well, we checked the distributions

it produced for some reference quantities letting the network to fill each distribution according to its prediction on the events of the test dataset. On the same events we also checked the verdict of the standard selection. The distributions are shown in Figure 5.5. It's interesting to note that the acceptance of signal events increases by a factor ≈ 4 but the standard approach is better in background suppression (red solid line in each plot). To compare the level of purity of the two selection we used the $S/\sqrt{N}$. With the present dataset composition we get a value ≈ 5 times higher for the standard analysis with respect to the neural network one. In Table 5.2 we put the estimated values for the composition of the final sample considering N_{K^+} decays computed by taking the signal sample as the reference decay channel:

$$N_{K^+} = \frac{N_{gen}^S}{BR(K^+ \rightarrow \pi^+ \pi_D^0)} \approx 19 \times 10^9 \quad (5.6)$$

Decay channel	N_{NN}	N_{STD}
$K^+ \rightarrow \pi^+ \pi^0, \pi^0 \rightarrow e^+ e^- \gamma$	586980	138283
$K^+ \rightarrow \pi^+ \pi^0, \pi^0 \rightarrow \gamma \gamma$	25270	0
$K^+ \rightarrow \pi^+ \pi^0 \pi^0, \pi^0 \rightarrow \gamma \gamma, \pi^0 \rightarrow e^+ e^- \gamma$	1465	0
$K^+ \rightarrow \pi^+ \pi^0 \pi^0, \pi^0 \rightarrow \gamma \gamma, \pi^0 \rightarrow \gamma \gamma$	59	0
$K^+ \rightarrow \pi^+ \pi^+ \pi^-$	62497	0
$K^+ \rightarrow \pi^0 e^+ \nu, \pi^0 \rightarrow \gamma \gamma$	743	31
$K^+ \rightarrow \pi^0 e^+ \nu, \pi^0 \rightarrow e^+ e^- \gamma$	23102	0
$K^+ \rightarrow \pi^0 \mu^+ \nu, \pi^0 \rightarrow \gamma \gamma$	857	0
$K^+ \rightarrow \pi^0 \mu^+ \nu, \pi^0 \rightarrow e^+ e^- \gamma$	27432	647

Table 5.2: Estimated number of events passing the neural network selection (second column) and the standard one (third column), starting from a initial number of kaons as explained in the text. From the table it's clear that the neural network should be tuned better to reduce the background contamination, which is almost absent in the standard selection.

We performed other studies including other quantities as the vertex position or the, E/p of the tracks etc. but the result remains unsatisfactory. We also tried to increase the size of the newtork, adding layers or neurons in each layer, with the only result that the model overfitted or doesn't learn because the "vanishing gradient problem", that is the phenomenon that due to the high number of layers in the network, causes a progressive reduction of the gradients which are needed to update the weights.

5.3 Conclusions

In this chapter we reported our preliminary study about a new possible way of performing a physics analysis of a specific decay channel using a DL algorithms to select signal events among a certain dataset. We showed how the problem is difficult to face with the simplest DL algorithm such as Feed Forward neural networks due to the fixed number of input features, which requires the design of datasets which are not free from assumptions (e.g. the fixed number of candidates in the event and the necessary *padding* in the first implemented dataset or the fixed topology of

the event in the second one). Anyway we are satisfied to have at least found a way to overcome all the presented problems and to have extracted at least the positive fact that the network can in principle increase the acceptance of the signal events, without any prior knowledge on the physics underlying the events.

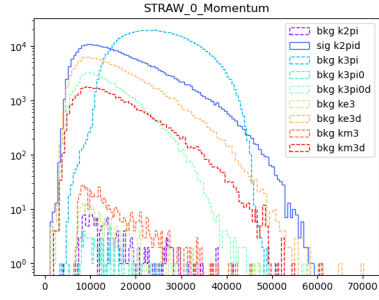
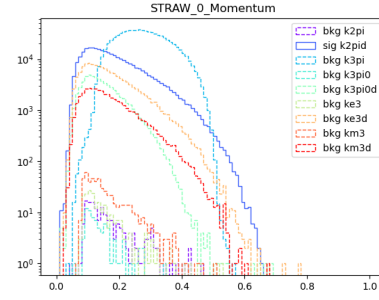
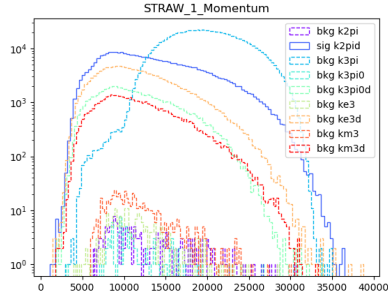
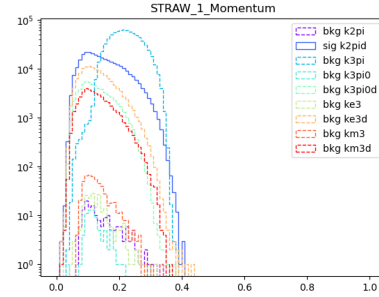
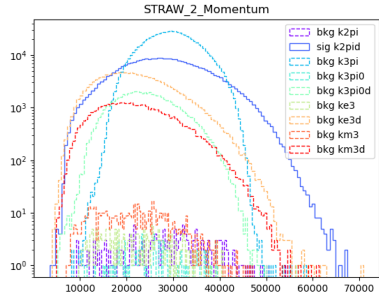
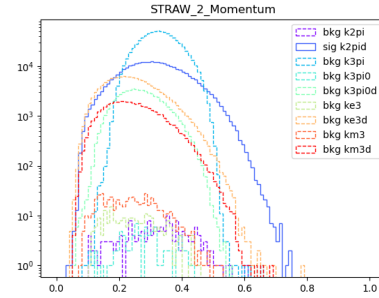
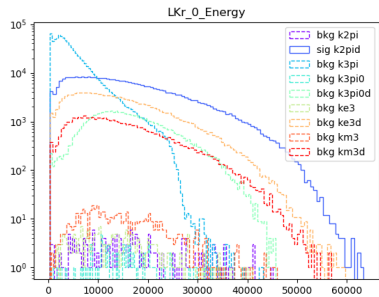
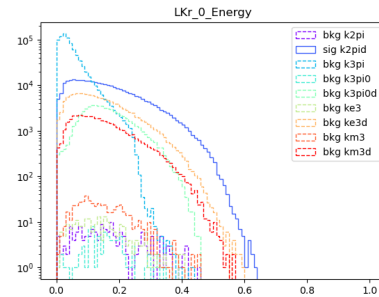

 (a) $P^{(0)}$ [MeV/c]

 (b) $P^{(0)}/90000$ [MeV/c] scaled

 (c) $P^{(1)}$ [MeV/c]

 (d) $P^{(1)}/90000$ [MeV/c] scaled

 (e) $P^{(2)}$ [MeV/c]

 (f) $P^{(2)}/90000$ [MeV/c] scaled

 (g) $E^{(0)}$ [MeV]

 (h) $E^{(0)}/100000$ [MeV] scaled

Figure 5.3: Distributions of some of the quantities composing the dataset. On the left are shown the momenta of the three tracks and the energy of the single cluster with their standard ranges. On the right are shown the same distributions but after the scaling procedure. The signal distribution is shown with a solid line.

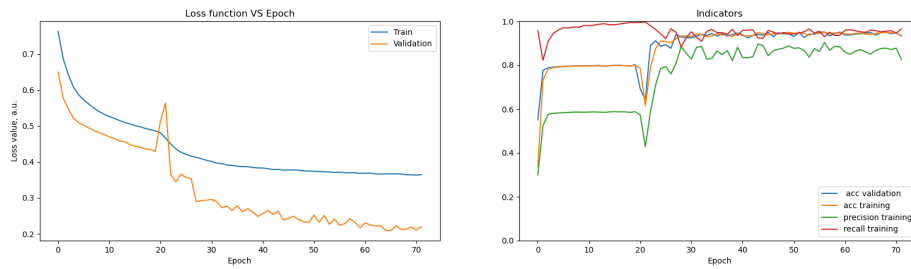


Figure 5.4: Results of the training phase. On plot on the left are shown the loss functions computed from training and for the validation samples during each epoch (i.e. after the network run on the whole training dataset). On the right plot are show the performance indicators as the accuracy, precision and recall. The overall accuracy on the validation sample is $\sim 95\%$.

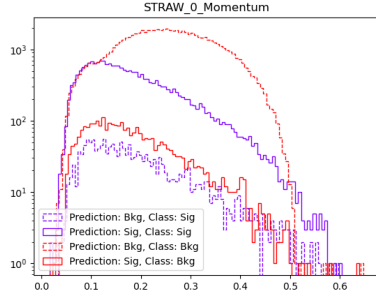
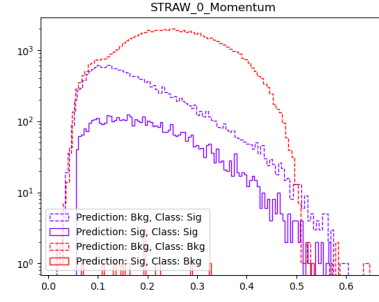
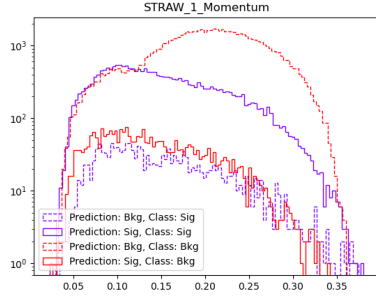
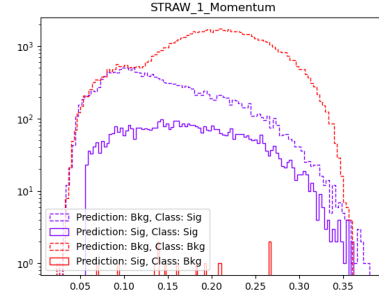
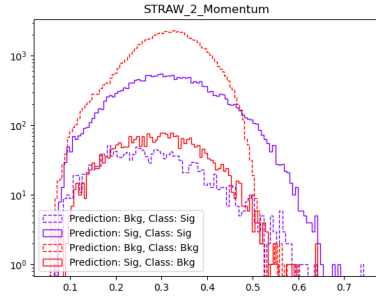
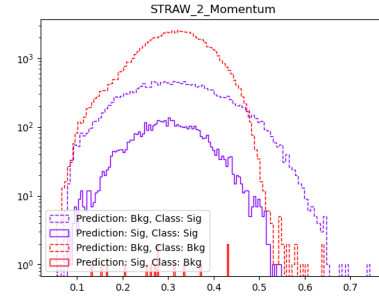
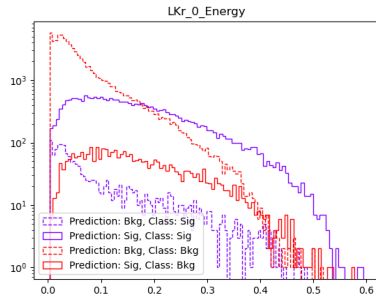
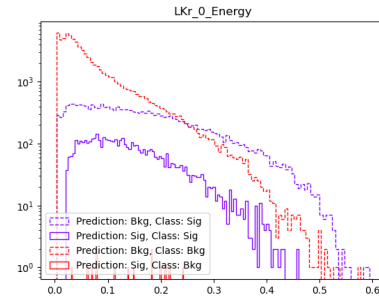

 (a) $P^{(0)}$ [MeV/c]

 (b) $P^{(0)}/90000$ [MeV/c] scaled

 (c) $P^{(1)}$ [MeV/c]

 (d) $P^{(1)}/90000$ [MeV/c] scaled

 (e) $P^{(2)}$ [MeV/c]

 (f) $P^{(2)}/90000$ [MeV/c] scaled

 (g) $E^{(0)}$ [MeV]

 (h) $E^{(0)}/100000$ [MeV] scaled

Figure 5.5: Distributions of some of the quantities composing the whole dataset. On the left we plot the distributions of the quantities divided in the two classes according to predicted/true class of each event. On the right we do the same for the output of the classification but from the "standard" selection. It's visible the improve in acceptance (solid purple line) generated by the event selection performed by the neural network.

Chapter 6

Conclusions

In our work we performed four different studies, all centered on the π^0 Dalitz decay. The collection of 8.6×10^6 events leads to a preliminary result on the TFF slope of the π^0 Dalitz decay

$$a \pm \delta a_{stat} \pm \delta a_{sys} = (3.56 \pm 0.29 \pm 1.35) \times 10^{-2} \quad (6.1)$$

Despite the reduction of the statistical uncertainty, the systematics effect dominated the precision of the measurement. In any case, our measurement to date is compatible with the VMD model and confirms a positive slope for the form factor. Our analysis was presented during various internal meetings in the collaboration, but the (6.1) result to date is not an official NA62 result.

We developed a new method for the LKr energy fine calibration, which to date has become the default calibration method, thanks to the fact that it reaches comparable results with respect to the previous adopted procedures in terms of quality of the data, but requiring less than 1/4 of the time during a data-processing round. This has to be taken into account considering that the experiment is taking data, and a fast processing of a Run might be fundamental to spot eventual problems and soon react.

We showed how Deep Learning methods might be implemented to improve the L1 trigger algorithms of the NA62 experiment, helping in the reduction of downscaling factors, to date one of the major limits to collect high statistics, especially in the analysis of multitrack decays.

Finally, we put the basis for a complete new way of performing a physics analysis, without any prior knowledge about the physics under study. Despite the intrinsic interest, the immediate consequence of this approach would be to extract from the algorithm the laws used during the inference, which could in principle reveal hints of new physics.

Acknowledgments

As the reader has noted, the author of the thesis used for the whole description of the developed work the first personal pronoun "we". This choice has a precise motivation: the author has imaged to write this thesis together with all the people that, with any kind of help, contributed to its final form.

Specific thanks go to my supervisor Prof. Marco Sozzi for the thesis review and correction; to Prof. Gianluca Lamanna for the very stimulating chats about Machine Learning (which sometimes took a more philosophical aspect); to Mgr. Michal Koval for supervising me in the study of the π^0 standard analysis; to Karim Massri for including me in the NA62 processing group and to help me in the implementation of the new calibration technique; to all the people of the NA62 collaboration in general.

I want to thank all of my friends, met in all the years spent at the university, for all the time spent together, all the discussions (about physics but also not...) and all the support they gave me. Special thanks go to Damiano Zanetti for explaining me theoretical physics in a very simple way.

I want to thank my family which always supported me.

Finally I want to thank Jessica, my girlfriend, who I think could be able to present this thesis without problems since She was there in every moments, the good ones but above all the bad ones.

Appendix A

A.1 Data/MC comparison

In this appendix we discuss the general approach adopted for the Data/MC superposition used in the analysis. As described in section 1.5, available MC lists can be "simple", meaning that only one decay per event is present or "overlaid", particular samples designed to mimic the real data pileup, in which more than one decay is present in the same event. According to which MC type is used during the analysis, a different method should be adopted in order to scale MC samples between them and with respect to data. We put in evidence how the re-scaling of MC samples is needed because the number of generated events in each MC sample is not linked to the BR , so for example the same number of events passing the whole selection for two different MC samples, when re-weighted with their BR , could differ a lot¹. In the "simplest" case, if we processed m MC samples, we prepared m different histograms and we filled the corresponding one according to the list the event belonged to (e.g. if the processed event belongs to the $K^+ \rightarrow \pi^+ \pi_D^0$ sample, we put an entry in the histogram of the $K^+ \rightarrow \pi^+ \pi_D^0$ sample). Knowing the number of generated events in that MC (N_g^i), it's possible to scale any other MC with

$$f_i = \frac{BR_i}{N_g^i}, \quad i = \text{MC samples included in the analysis} \quad (\text{A.1})$$

This approach cannot be used for the overlaid samples. Indeed, in this case, the reconstructed event is not necessarily the one indicated by the MC sample; for example, we could have reconstructed a $K_{3\pi}$ event processing a $K_{2\pi D}$ file. The main problem is that we have to classify the reconstructed event by checking which MC particles have generated it. This information is stored even if all the events are merged together as one, since each "simple" event can be separated from the others by a specific code (streamID). An algorithm, present in the NA62 analysis framework, has been developed to accomplish this task. It is able to assign to each reconstructed track the most probable MC particle which could have generated it. This algorithm can also fail, returning a specific message. When this happens we reject the event since it is not possible to assign the reconstructed event to a generated one. The rate of failure is very low. For example, in the last step of the selection 2.4 we found $\sim 0.007\%$ not classified. Since the signature of the $K_{2\pi D}$ decay is a three track event, we required that all the tracks of the event are assigned to MC particles. Moreover we required that the assigned MC particles belong to

¹In the last period of this work was produced an inclusive MC sample in which the K^+ decays were simulated respecting the real BR values. We did not exploit it.

the same merged event. When this criteria are fulfilled, we checked the streamID and we classify correctly the event, thus filling the correct distribution. In (A.1) Also the normalization partially is affected by the presence of overlaid events: for example the total number of generated $K_{2\pi D}$ should count, the $K_{2\pi D}$ present in the $K_{2\pi D}$ files, but also the $K_{2\pi D}$ events from MC files of other samples (e.g. $K_{3\pi}$) if a $K_{2\pi D}$ event is fully reconstructed. We solve this problem by updating a different counter (one for each MC sample). So if the number of generated events in a certain sample is N_g^i and we reconstruct a pure event of that sample from the processing of another MC sample we update N_g^i by 1.

The scaling with respect to Data is performed by adding all the rescaled MC together and taking the integral of the sum as nMC . Then each MC is scaled by $nData/nMC$, where $nData$ is the integral of the distribution for the data sample.

Appendix B

B.1 Performance indicators

Here we provide the definitions of some performance indicators used in the analysis. Considering a binary classification with a class called signal and the other background, we call

- True Negative (TN): event **belonging to the background** class and **predicted as member of the background** class
- False Positive (FP): event **belonging to the background** class and **predicted as member of the background** class
- False Negative (FN): event **belonging to the signal** class and **predicted as member of the background** class
- True Positive (TP): event **belonging to the signal** class and **predicted as member of the signal** class

All these quantities can be arranged into a matrix, called **confusion matrix**, that in our case is defined with the convention shown in Figure B.1. We always normalize this matrix per rows, to show the fraction of good/wrong classifications in each class. Other quantities used in the text are

$$\text{Accuracy} = \frac{TP}{TN + FP + FN + TP} \quad (\text{B.1})$$

$$\text{Precision} = \frac{TP}{FP + TP} \quad (\text{B.2})$$

$$\text{Recall} = \frac{TP}{FN + TP} \quad (\text{B.3})$$

The accuracy describes the absolute fraction of correctly classified events; the precision describes the "credibility" level of the model; the recall is a measure of the sensitivity of the model to signal events.

B.2 Generalization of the Crawford fit

We used a modified version of the Crawford fitting algorithm described in [14], since we have to perform the fit using hits with different weights. Using same notation of

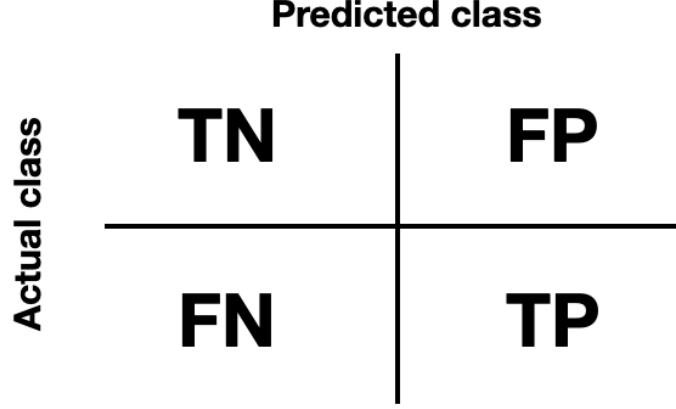


Figure B.1: Definition of confusion matrix for a binary problem. "Actual" class indicates the real class of an event, while "Predicted" class stands for the class which the model assigned to the event.

[14], we have a set of N coordinates (x_i, y_i) and we want to minimize the condition.

$$S = \sum_{i=1}^N ((x_i - x_0)^2 + (y_i - y_0)^2 - R^2)^2 \quad (\text{B.4})$$

When it is assigned a weight w_i to each hit, this relation changed into

$$S = \sum_{i=1}^N w_i ((x_i - x_0)^2 + (y_i - y_0)^2 - R^2)^2 \quad (\text{B.5})$$

and for this the following relations

$$\frac{\partial S}{\partial x_0} = 4 \sum_{i=1}^N (x_0 - x_i) ((x_i - x_0)^2 + (y_i - y_0)^2 - R^2) = 0 \quad (\text{B.6})$$

$$\frac{\partial S}{\partial y_0} = 4 \sum_{i=1}^N (y_0 - y_i) ((x_i - x_0)^2 + (y_i - y_0)^2 - R^2) = 0 \quad (\text{B.7})$$

$$\frac{\partial S}{\partial R} = -4 \sum_{i=1}^N R ((x_i - x_0)^2 + (y_i - y_0)^2 - R^2) = 0 \quad (\text{B.8})$$

are replaced by the more general ones

$$\frac{\partial S}{\partial x_0} = 4 \sum_{i=1}^N w_i (x_0 - x_i) ((x_i - x_0)^2 + (y_i - y_0)^2 - R^2) = 0 \quad (\text{B.9})$$

$$\frac{\partial S}{\partial y_0} = 4 \sum_{i=1}^N w_i (y_0 - y_i) ((x_i - x_0)^2 + (y_i - y_0)^2 - R^2) = 0 \quad (\text{B.10})$$

$$\frac{\partial S}{\partial R} = -4 \sum_{i=1}^N w_i R ((x_i - x_0)^2 + (y_i - y_0)^2 - R^2) = 0 \quad (\text{B.11})$$

Using the updated constraint that

$$\sum_{i=1}^N w_i ((x_i - x_0)^2 + (y_i - y_0)^2 - R^2) = 0 \quad (\text{B.12})$$

one can find that the values for x_0 , y_0 , R to fit coordinates with weights can be obtained by the replacements

$$\sum_{i=1}^N x_i^2 \rightarrow \sum_{i=1}^N w_i x_i^2 \quad (\text{B.13})$$

$$\sum_{i=1}^N x_i y_i \rightarrow \sum_{i=1}^N w_i x_i y_i \quad (\text{B.14})$$

$$\sum_{i=1}^N y_i^2 \rightarrow \sum_{i=1}^N w_i y_i^2 \quad (\text{B.15})$$

$$\sum_{i=1}^N x_i^3 \rightarrow \sum_{i=1}^N w_i x_i^3 \quad (\text{B.16})$$

$$\sum_{i=1}^N x_i^2 y_i \rightarrow \sum_{i=1}^N w_i x_i^2 y_i \quad (\text{B.17})$$

$$\sum_{i=1}^N x_i y_i^2 \rightarrow \sum_{i=1}^N w_i x_i y_i^2 \quad (\text{B.18})$$

$$\sum_{i=1}^N y_i^3 \rightarrow \sum_{i=1}^N w_i y_i^3 \quad (\text{B.19})$$

modifying also the prescription to perform the fit weighted center of gravity of the points:

$$\sum_{i=1}^N x_i = \sum_{i=1}^N w_i x_i \quad (\text{B.20})$$

$$\sum_{i=1}^N y_i = \sum_{i=1}^N w_i y_i \quad (\text{B.21})$$

Bibliography

- [1] J Allison et al. “Geant4 Developments and Applications”. In: 53 (Feb. 2006), pp. 270–278.
- [2] A. V. Artamonov et al. “New Measurement of the $K^+ \rightarrow \pi^+ \nu \bar{\nu}$ Branching Ratio”. In: *Phys. Rev. Lett.* 101 (19 Nov. 2008), p. 191802. DOI: 10.1103/PhysRevLett.101.191802. URL: <https://link.aps.org/doi/10.1103/PhysRevLett.101.191802>.
- [3] J.R. Batley et al. “First observation and study of the $K^\pm \rightarrow \pi^\pm \pi^0 e^+ e^-$ decay”. In: *Physics Letters B* 788 (2019), pp. 552–561. ISSN: 0370-2693. DOI: <https://doi.org/10.1016/j.physletb.2018.11.046>. URL: <https://www.sciencedirect.com/science/article/pii/S037026931830892X>.
- [4] H. J. Behrend et al. “A Measurement of the π^0 , η and η' electromagnetic form-factors”. In: *Z. Phys.* C49 (1991), pp. 401–410. DOI: 10.1007/BF01549692.
- [5] R. Bjorklund et al. “High Energy Photons from Proton-Nucleon Collisions”. In: *Phys. Rev.* 77 (2 Jan. 1950), pp. 213–218. DOI: 10.1103/PhysRev.77.213. URL: <https://link.aps.org/doi/10.1103/PhysRev.77.213>.
- [6] K. L. Brown and F. C. Iselin. “Decay TURTLE (Trace Unlimited Rays Through Lumped Elements): A Computer Program for Simulating Charged Particle Beam Transport Systems, Including Decay Calculations”. In: (1974).
- [7] R. Brun and F. Rademakers. “ROOT: An object oriented data analysis framework”. In: *Nucl. Instrum. Meth. A* 389 (1997). Ed. by M. Werlen and D. Perret-Gallix, pp. 81–86. DOI: 10.1016/S0168-9002(97)00048-X.
- [8] *ROOT - An Object Oriented Data Analysis Framework*. Proceedings AIHENP’96 Workshop, Lausanne. See also ”ROOT” [software], Release v6.22/06, 27/11/2020, <https://doi.org/10.5281/zenodo.3895860>. Sept. 1996.
- [9] Andrzej J. Buras et al. “ $K^+ \rightarrow \pi^+ \nu \bar{\nu}$ and $K_L \rightarrow \pi^0 \nu \bar{\nu}$ in the Standard Model: status and perspectives”. In: *JHEP* 11 (2015), p. 033. DOI: 10.1007/JHEP11(2015)033. arXiv: 1503.02693 [hep-ph].
- [10] Patrizia Cenci. “The neutral pion form factor at NA62”. In: *Journal of Physics: Conference Series* 770.1 (2016), p. 012035. URL: <http://stacks.iop.org/1742-6596/770/i=1/a=012035>.
- [11] N. Chernov. *Circular and Linear Regression: Fitting Circles and Lines by Least Squares*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis, 2010. ISBN: 9781439835906. URL: <https://books.google.it/books?id=GaKTQgAACAAJ>.

- [12] E. Cortina Gil et al. “Measurement of the very rare $K^+ \rightarrow \pi^+ \nu \overline{\nu}$ decay”. In: *Journal of High Energy Physics* 2021.6 (2021), p. 93. DOI: 10.1007/JHEP06(2021)093. URL: [https://doi.org/10.1007/JHEP06\(2021\)093](https://doi.org/10.1007/JHEP06(2021)093).
- [13] Eduardo Cortina Gil et al. “The Beam and detector of the NA62 experiment at CERN”. In: *JINST* 12.05 (2017), P05025. DOI: 10.1088/1748-0221/12/05/P05025. arXiv: 1703.08501 [physics.ins-det].
- [14] J.F. Crawford. “A non-iterative method for fitting circular arcs to measured points”. In: *Nuclear Instruments and Methods in Physics Research* 211.1 (1983), pp. 223–225. ISSN: 0167-5087. DOI: [https://doi.org/10.1016/0167-5087\(83\)90575-6](https://doi.org/10.1016/0167-5087(83)90575-6). URL: <https://www.sciencedirect.com/science/article/pii/0167508783905756>.
- [15] R H Dalitz. “On an Alternative Decay Process for the Neutral π -Meson”. In: *Proceedings of the Physical Society. Section A* 64.7 (July 1951), pp. 667–669. DOI: 10.1088/0370-1298/64/7/115. URL: <https://doi.org/10.1088/0370-1298/64/7/115>.
- [16] V. Fanti et al. “The beam and detector for the NA48 neutral kaon CP violation experiment at CERN”. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 574.3 (2007), pp. 433–471. ISSN: 0168-9002. DOI: <https://doi.org/10.1016/j.nima.2007.01.178>. URL: <https://www.sciencedirect.com/science/article/pii/S0168900207002719>.
- [17] F. Farzanpay et al. “Measurement of the slope of the π^0 electromagnetic form-factor”. In: *Phys. Lett. B* 278 (1992), pp. 413–418. DOI: 10.1016/0370-2693(92)90577-Q.
- [18] J. Fischer et al. “Measurement of the slope of the π^0 -electromagnetic form factor”. In: *Physics Letters B* 73.3 (1978), pp. 359–363. ISSN: 0370-2693. DOI: [https://doi.org/10.1016/0370-2693\(78\)90534-8](https://doi.org/10.1016/0370-2693(78)90534-8). URL: <https://www.sciencedirect.com/science/article/pii/0370269378905348>.
- [19] H. Fonvieille et al. “Dalitz decay. $\pi^0 \rightarrow \gamma e^+ e^-$ and the π^0 electromagnetic transition form factor”. In: *Physics Letters B* 233.1 (1989), pp. 65–68. ISSN: 0370-2693. DOI: [https://doi.org/10.1016/0370-2693\(89\)90616-3](https://doi.org/10.1016/0370-2693(89)90616-3). URL: <https://www.sciencedirect.com/science/article/pii/0370269389906163>.
- [20] R. Frühwirth. “Application of Kalman filtering to track and vertex fitting”. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 262.2 (1987), pp. 444–450. ISSN: 0168-9002. DOI: [https://doi.org/10.1016/0168-9002\(87\)90887-4](https://doi.org/10.1016/0168-9002(87)90887-4). URL: <https://www.sciencedirect.com/science/article/pii/0168900287908874>.
- [21] Nikolay Gagunashvili. “Comparison of weighted and unweighted histograms”. In: (May 2006). DOI: 10.1142/9781860948985_0010.
- [22] Murray Gell-Mann and Fredrik Zachariasen. “Form Factors and Vector Mesons”. In: *Phys. Rev.* 124 (3 Nov. 1961), pp. 953–964. DOI: 10.1103/PhysRev.124.953. URL: <https://link.aps.org/doi/10.1103/PhysRev.124.953>.

- [23] Alberto Gianoli et al. “The NA48 LKr calorimeter readout electronics”. In: *Nuclear Science, IEEE Transactions on* 47 (May 2000), pp. 136–141. DOI: 10.1109/23.846133.
- [24] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. <http://www.deeplearningbook.org>. MIT Press, 2016.
- [25] S. Gorbunov and I. Kisel. “Elastic net for stand-alone RICH ring finding”. In: *Nucl. Instrum. Meth. A* 559 (2006). Ed. by J. Blumlein et al., pp. 139–142. DOI: 10.1016/j.nima.2005.11.215.
- [26] Aurlien Gron. *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. 1st. O’Reilly Media, Inc., 2017. ISBN: 1491962291.
- [27] Geoffrey E. Hinton et al. *Improving neural networks by preventing co-adaptation of feature detectors*. 2012. arXiv: 1207.0580 [cs.NE].
- [28] T. Husek and J. Novotný K. Kampf. “Radiative corrections to the Dalitz decay $\pi^0 \rightarrow e^+e^-\gamma$ revisited”. In: *Phys. Rev. D* 92 (5 Sept. 2015), p. 054027. DOI: 10.1103/PhysRevD.92.054027. URL: <https://link.aps.org/doi/10.1103/PhysRevD.92.054027>.
- [29] Tomáš Husek, Evgueni Goudzovski, and Karol Kampf. “Precise Determination of the Branching Ratio of the Neutral-Pion Dalitz Decay”. In: *Physical Review Letters* 122.2 (Jan. 2019). ISSN: 1079-7114. DOI: 10.1103/physrevlett.122.022003. URL: <http://dx.doi.org/10.1103/PhysRevLett.122.022003>.
- [30] Sergey Ioffe and Christian Szegedy. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. 2015. arXiv: 1502.03167 [cs.LG].
- [31] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: 1412.6980 [cs.LG].
- [32] G. Lamanna. “Almagest, a new trackless ring finding algorithm”. In: *Nucl. Instrum. Meth. A* 766 (2014). Ed. by T. Sumiyoshi et al., pp. 241–244. DOI: 10.1016/j.nima.2014.05.073.
- [33] L.G. Landsberg. “Electromagnetic decays of light mesons”. In: *Physics Reports* 128.6 (1985), pp. 301–376. ISSN: 0370-1573. DOI: [https://doi.org/10.1016/0370-1573\(85\)90129-2](https://doi.org/10.1016/0370-1573(85)90129-2). URL: <http://www.sciencedirect.com/science/article/pii/0370157385901292>.
- [34] C. M. G. Lattes, G. P. S. Occhialini, and C. F. Powell. “Observations on the Tracks of Slow Mesons in Photographic Emulsions. 1”. In: *Nature* 160 (1947), pp. 453–456. DOI: 10.1038/160453a0.
- [35] B. E. Lautrup and J. Smith. “Radiative Corrections to Decays with a Dalitz Pair”. In: *Phys. Rev. D* 3 (5 Mar. 1971), pp. 1122–1135. DOI: 10.1103/PhysRevD.3.1122. URL: <https://link.aps.org/doi/10.1103/PhysRevD.3.1122>.
- [36] C. Lazzeroni et al. “Measurement of the π^0 electromagnetic transition form factor slope”. In: *Physics Letters B* 768 (2017), pp. 38–45. ISSN: 0370-2693. DOI: <https://doi.org/10.1016/j.physletb.2017.02.042>. URL: <http://www.sciencedirect.com/science/article/pii/S0370269317301557>.

-
- [37] R. Meijer Drees et al. “Measurement of the π^0 electromagnetic transition form factor”. In: *AIP Conf. Proc.* 243 (1992). Ed. by Willem T. H. van Oers, pp. 657–659. DOI: 10.1063/1.41578.
 - [38] K. O. Mikaelian and J. Smith. “Radiative Corrections to the Decay $\pi^0 \rightarrow \gamma e^+ e^-$ ”. In: *Phys. Rev. D* 5 (7 Apr. 1972), pp. 1763–1773. DOI: 10.1103/PhysRevD.5.1763. URL: <https://link.aps.org/doi/10.1103/PhysRevD.5.1763>.
 - [39] N.Lurkin. *NA62 Framework*. <https://na62-sw.web.cern.ch>.
 - [40] Collaboration NA62. *2017 NA62 status report to the CERN SPSC*. Tech. rep. Geneva: CERN, Mar. 2017. URL: <https://cds.cern.ch/record/2256924>.
 - [41] Andreas Nyffeler. “Precision of a data-driven estimate of hadronic light-by-light scattering in the muon $g - 2$: Pseudoscalar-pole contribution”. In: *Phys. Rev. D* 94.5 (2016), p. 053006. DOI: 10.1103/PhysRevD.94.053006. arXiv: 1602.03398 [hep-ph].
 - [42] Adam Paszke et al. “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach et al. Curran Associates, Inc., 2019, pp. 8024–8035. URL: <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>.
 - [43] Sumit Saha. *A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way*. 2018. URL: <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>.
 - [44] Hideki YUKAWA. “On the Interaction of Elementary Particles. I”. In: *Proceedings of the Physico-Mathematical Society of Japan. 3rd Series* 17 (1935), pp. 48–57. DOI: 10.11429/ppmsj1919.17.0_48.
 - [45] P.A. Zyla et al. “Review of Particle Physics”. In: *PTEP* 2020.8 (2020), p. 083C01. DOI: 10.1093/ptep/ptaa104.