# Analysis of KalmanFilter3D module

Giovanni Cerretani[1,2], Denis Perevalov[1]
[1] *Fermi National Accelerator Laboratory,* [2] *Università di Pisa*

September 26, 2013

**Abstract**

The NOvA-ART module KalmanFilter3D has been used to reconstruct $\mu^{\pm}$ in simulated neutrino interaction events both in the far detector and in the near detector. We developed a module in order to compare the reconstructed energy and position in $xy$-plane with the original ones in order to have an estimation of the efficiency of this algorithm. We also try a new muon tagging algorithm for neutrino interaction events.

# Introduction

In this paper we present applications of NOvA-ART module KalmanFilter3D on simulated neutrino interactions to reconstruct muon tracks. In the first chapter the NOvA experiment is briefly described. In the second chapter we present the theory of Kalman filters and some practical examples including the NOvA reconstructing algorithm KalmanFilter3D. In the third chapter and the following ones we present some results of the application of this module to neutrino events in the NOvA detectors, obtained from the comparison between Montecarlo simulation data and reconstructed tracks.

# 1 NOvA project

NOvA is a $\nu_\mu \to \nu_e$ oscillation appearance experiment at Fermilab, that use NuMI neutrino and two detectors placed off-axis: a 330 metric-ton near detector at Fermilab (placed 1 km from source) and a much larger 14,625 metric-ton far detector in Minnesota (810 km from Fermilab) just south of the U.S.-Canada border. The detectors are made up of 344,000 cells of plastic PVC filled with liquid scintillator and with an optical fiber to collect photons inside of it. Each cell in the far detector measures 3.9 cm wide, 6.0 cm deep and 15.5 m long.

In this off-axis location we find a large flux of neutrinos at an energy of 2 GeV the energy, at which oscillation $\nu_\mu \to \nu_e$ is expected to be at a maximum.

The first aim of NOvA is the measurement of the little known mixing angle $\sin^2 2\theta_{13}$, that control $\nu_\mu \to \nu_e$ oscillation. The probability of this oscillation in vacuum is

$$P_{vac}\left(\nu_\mu \to \nu_e\right) = \sin^2\theta_{23}\sin^2 2\theta_{13}\sin^2\Delta_{atm}$$

where $\Delta_{atm} = 1.27\Delta m^2_{32}L/E$. If the experiment is performed at the peak of this probability, i.e. when $\sin^2\Delta_{atm} = 1$, then

$$P^{MAX}_{vac}\left(\nu_\mu \to \nu_e\right) = \sin^2 2\theta_{13}\sin^2 2\theta^2_{13} \approx \frac{1}{2}\sin^2 2\theta^2_{13}$$

Actually, neutrinos propagate into the Earth, and matter induced contributions to the propagation amplitude are non-negligible. Furthermore, matter effects have opposite sign for neutrinos and anti-neutrinos and for normal versus - inverted neutrino mass hierarchies. The matter effects can be thus used to distinguish the two possible three neutrino mass hierarchies. [2]

Another purpose of the experiment is the observation of CP violation in lepton physics: NuMI is able to generate both a $\nu_\mu$ and a $\bar{\nu}_\mu$ beam, and different results can lead to the measurement of a complex phase in neutrino mixing matrix.

# 2 Kalman filters

Generically, Kalman filter is an algorithm that uses a series of measurements observed over time, containing noise (random variations) and other inaccuracies, and produces estimates of unknown variables that tend to be more precise than those based on a single measurement alone. It is used in a wide range of engineering and econometric applications from radar to GPS navigation systems and autopilot, from computer vision to estimation of structural macroeconomic models, and is an important topic in control theory and control systems engineering.

## 2.1 A little of formalism

We assume a target moving radially away or toward our radar in a one dimensional world, with $x_n$ representing the slant range to the target at time $n$. To simplify, we assume the target's velocity is constant. In this way we can predict the position and the velocity of the target at the second scan using this simple differential equations:

$$\begin{cases} x_{n+1} &= x_n + T\,\dot{x}_n \\ \dot{x}_{n+1} &= \dot{x}_n \end{cases} \tag{1}$$

where $T$ is the scan-to-scan period. These equations are called the "system dynamic model". Now we need to show how to improve our estimate of the target position and velocity after an observation is made of the target position at some time $n$ and at successive times. We suppose to measure, at time $n+1$, a position $\bar{x}_{n+1}$ greater of the expectation found from equations 1. In this case, we can update the velocity $\dot{x}_{n+1}$ that will be used for the prediction of the next position:

$$\dot{x}_{n+1} = \dot{x}_n + h_n \left( \frac{\bar{x}_{n+1} - x_{n+1}}{T} \right)$$

where $h_n$ is a weight that takes in account the precision of our measurement, i.e. its error. With considerations alike, in 1960 Rudolf Kalman developed the algorithm of the same name. [3]

An interesting thing in Kalman filtering is that a filter also updates the total current state error, so that in principle the total error should decrease in stable systems.

## 2.2 KalmanFilter3D in NOvA

An algorithm that uses a Kalman filter has been developed to reconstruct of the tracks inside the NOvA detectors. The measurement used in the Kalman filter is the information given by the detectors and consists of the number of photo-electrons found in a single cell. In a first estimation, this number is proportional to the energy lost by a charged particle inside of it (even if it depends also on the position of the particle with respect to the end of the optical fiber). Furthermore, we know that the average energy loss in the matter is given by $E = \frac{dE}{dx}x$ where $x$ is length of the path and $\frac{dE}{dx}$ is the average energy loss in a certain material. In this way, we can use the combination of photons collected in a cell and its position to find the path that optimize the measurement of $E$.

In this algorithm, the state of the system that is propagated is represented by a 5 dimension vector

$$a_k = \left( x_k, y_k, \left.\frac{dx}{dz}\right|_k, \left.\frac{dy}{dz}\right|_k, E_k \right)$$

while a measurement consist of a collection of cells that are hit.

First of all, KalmanFilter3D needs to be initialized with an existing track. This can be taken from other existing algorithm that reconstruct the track only using geometrical information: examples are "FuzzyKVertex", "KalmanTrackMerge", "BPFitter track". The

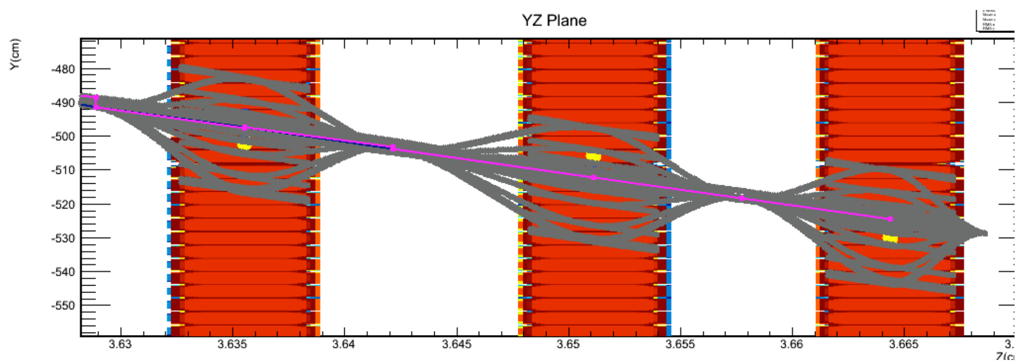



Figure 1: Graphical explanation of how KalmanFilter3D works.

efficiency of the filter depends weakly on which first guess track is used; KalmanTrack-Merge seems to optimize the algorithm, so its tracks have been used as first guesser track for all the results presented in this paper.

Then, the algorithm tries to minimize plane by plane the $\chi^2$ of a function that takes in account both the propagated state and measurement, and their respective errors:

$$\chi^2 = \underbrace{\left(a_k - f_{k-1}\left(a_{k-1}\right)\right)^T C^{-1}\left(a_k - f_{k-1}\left(a_{k-1}\right)\right)}_{\chi^2 \text{ of Kalman state}} + \underbrace{\left(m_k - h_k\left(a_k\right)\right)^T G\left(m_k - h_k\left(a_k\right)\right)}_{\chi^2 \text{ of measurement}}$$

The first term takes in account the prediction of the state made with the Kalman filter: $a_k$ is the "filtered" state, $f_{k-1}$ is the physical propagator, $C$ is the total covariance $5 \times 5$ matrix. The second term is the $\chi^2$ of the measurement: $m_k$ is the value of the measure, $h_k$ is the propagator and $G$ is the inverse of the measurement noise covariance $5 \times 5$ matrix.

The minimization is made using MINUIT, and a graphical explanation is given in figure 1: for each plane, the algorithm tries several trajectories around the projection from the previous states, and takes the one that minimizes the $\chi^2$.

# 3 Fixing "oscillating muons"

Before this work, KalmanFilter3D was used only for events with a single muon track with energy of 2 GeV. For the first time KalmanFilter3D algorithm has been used to reconstruct muon tracks in real $\nu_\mu$ events both in the near detector and in the far detector. In this case, the energy of the muons can be $> 2$ GeV.

Looking at the tracks in the event displayer, we noticed a strange behavior of the muon tracks, that seem to "oscillate" as shown in figure 2. This was due to the absence of the covariance matrices for tracks with energy $> 2$ GeV, used in the Kalman filter.

In figure 3 is shown the same track after the fix, that consists on the extension of the covariance matrix at 2 GeV for tracks at higher energies, being these almost independent from energy in this range. Now the algorithm is able to reconstruct track for a wider range of energy: this problem is no longer present for muons with energy up to $15 \div 20$ GeV, that is the maximum energy for a muon completely contained in the far detector.

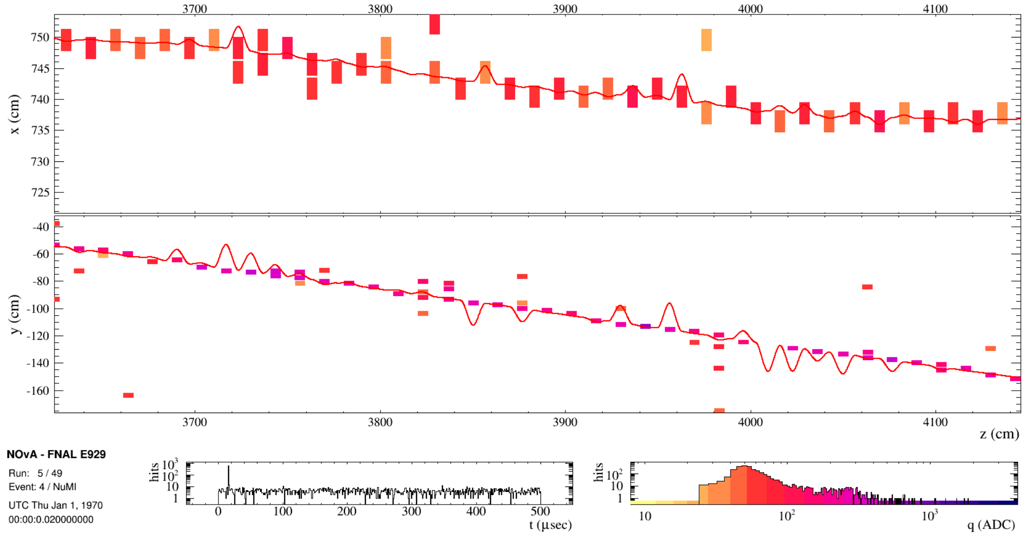



Figure 2: Oscillating muon

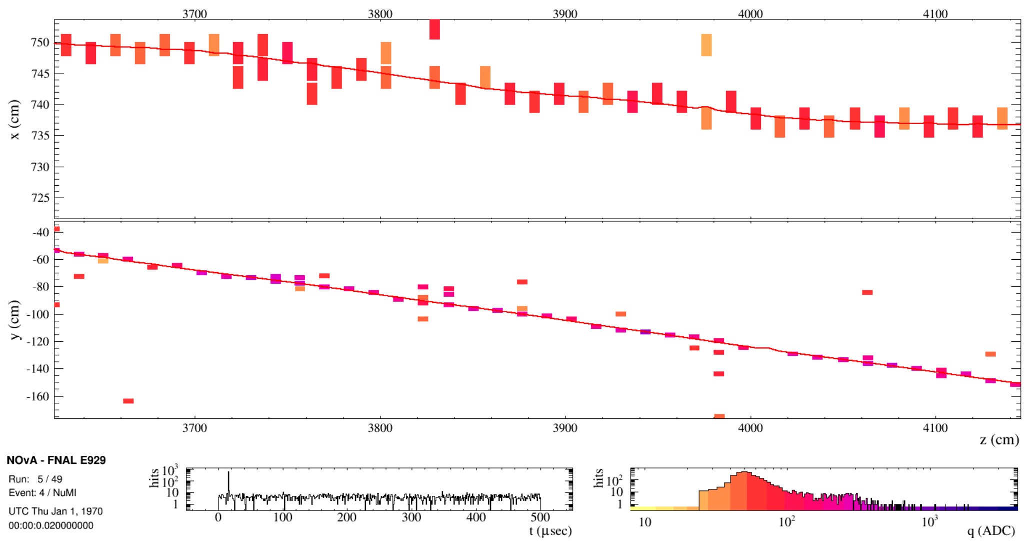



Figure 3: Oscillating muon fixed

# 4 Reconstruction of $\nu_\mu$ events

## 4.1 Montecarlo data and choice of events

The code has been modified to allow several kind of guesser tracks: now we're able to use more sophisticated tracks, that can consist not only of the initial and the final point (FuzzyKVertex tracks), but also of points in the middle (BPFitter, KalmanTrackMerge, ...).

The set of data used is S13-06-18, forward horn current, non-swap (for far detector). We've used 500 files for far detector each of them consisting in 3000 neutrino interactions and, 900 file for ND, each containing 1000 events. An important thing is that the near detector used in this simulations consists only on about half the plane: those farther from the source (without the initial 640 cm of detector), and the muon catcher.

For the analysis, we've looked only at $\mu^{\pm}$, that usually are one for each interaction. To find the muon in a event we selected the track with the highest purity with respect to the original muon, using the BackTracker service. For the analysis of the far detector it has been very simple: there is only a neutrino interaction per event (data are already triggered), and then a single muon track. Some problems arise in the near detector that, although is smaller than the far one, collect more because of the higher flux of neutrino: usually, there are more than a neutrino interaction in a window.

In both the cases, we also discarded events with all the tracks fully contained in the fiducial volume of the detector, that is so defined:

- FD: 50 cm of edge on all the sides;
- ND: 50 cm of edge on all the sides, 5 cm of edge on the back side where there is muon catcher.

Purity is found as follows: each hit of the reco object is tested to see if it corresponds to a particular generated particle, and the total number that match is divided by the total number of hits in the reconstructed object. Basically, the purity is a number between 0 and 1 that state the affinity between a track and a generated particle.

If more than a track was matching with the muon, we selected the most energetic one. This kind of choice has turned out to give good results in most the cases.

Eventually we've tried another tagging algorithm, that consisted in choosing the most energetic track in a certain event without looking at the purity. Results will be shown later in this chapter.

## 4.2 Far detector

A first tentative to use KalmanFilter3D on $\nu_\mu$ events has be made in the far detector. We've compared both the energy and the plane-by-plane position of the reconstructed track with the real track.

### 4.2.1 Energy

The energy of a charged particle leaving a track inside the detector can be reconstructed at least in two ways. The first consists in finding a relationship between the length of the track and its real energy. In figure 5 we can see that they are directly proportional:

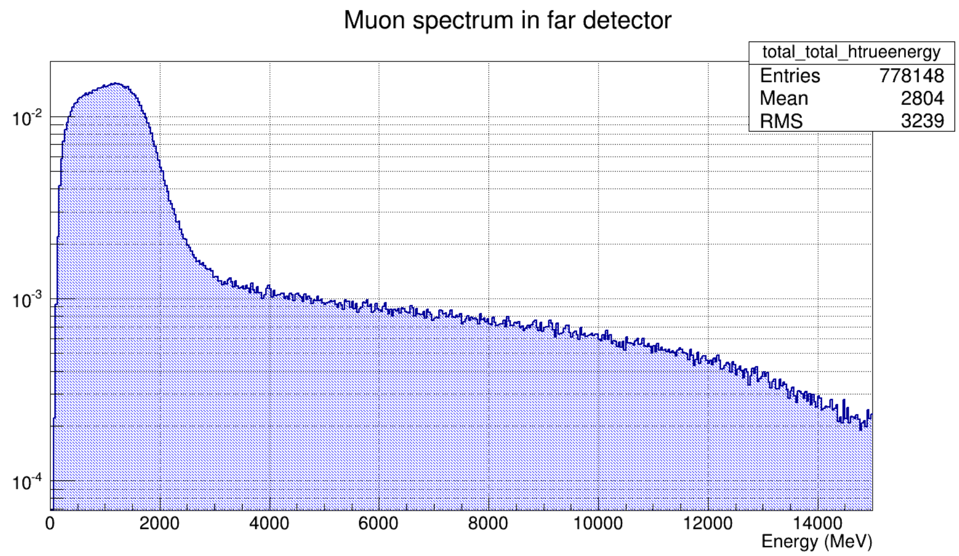



Figure 4: Spectrum of muons in the far detector, using the real energy.

$E(L) = \frac{dE}{dl} \cdot L$ with $\frac{dE}{dl} = 2.089 \pm 0.001\,{}^{\mathrm{MeV}}/_{\mathrm{cm}}$ is a good approximation, and a fit with a second grade polynomial is not needed. This coefficient represents the average energy loss in the far detector.

The second way to get the deposited energy is to integrate step by step the track with the $\frac{dE}{dl}$ in the current medium, that can be liquid of scintillators, PVC, glue, air, ... This is made by KalmanFilter3D algorithm, using its own track.

A first comparison between these two ways is shown in the histogram in figure 6. It collects the fraction $f$ of reconstructed energy for $\mu$ tracks, i.e. $f = \frac{E_{reco} - E_{true}}{E_{true}}$. No evident differences emerge from this histogram: both the algorithms tend to overestimate the energies in the same way.

In figure 7 are shown the behaviors of these two energy reconstructions in function of the energy. It turns out that KalmanFilter3D works better for $E > 3\,\mathrm{GeV}$.

According to the spectrum of muons (figure 4), most of them ($\sim 58\%$) have energy between 0.5 GeV and 2 GeV, and in this range the differences are relatively small.

### 4.2.2 Tentative to fix the track

KalmanFilter3D allows to fix some parameters of $\left(x, y, \frac{dx}{dz}, \frac{dy}{dz}, E\right)$ during the reconstruction. To check the efficiency of the Kalman filter, a tentative has been made that consisted in fixing all the parameters except for the last one, relative to the energy, though constrained in the range $[0.99, 1.01]$. Basically, in this configuration the output track and the first guesser track (i.e. KalmanTrackMerge track) are the same.

In figure 8 is shown the fraction of reconstructed energy measured with this configuration. It is clear that the efficiency is worse, and from figure 9, where the behavior as function of the muon energy is displayed, we can see that the "fixing position" method tends to overestimate the energy of particle up to $6\,\mathrm{GeV}$.

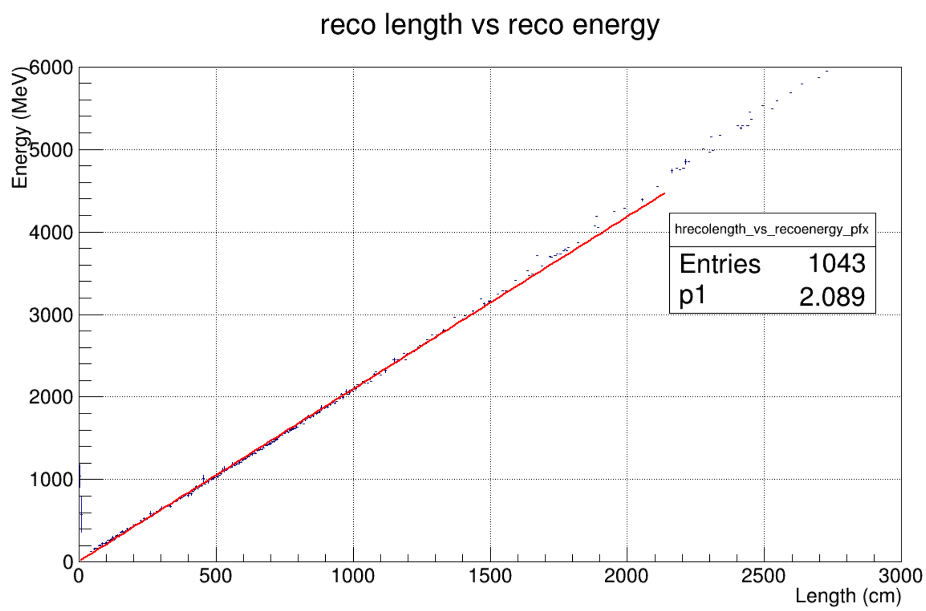



Figure 5: Dependence between length and deposited energy of track.

### 4.2.3 Position

As regards the reconstruction of position of the tracks in the space (for now omitting the time coordinate), we proceed as follow.

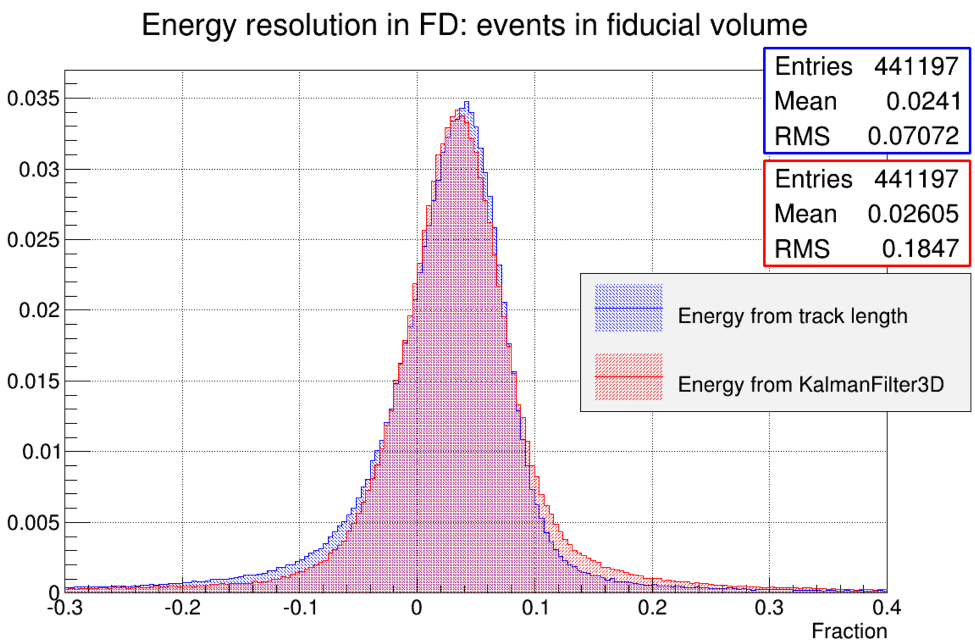
First of all, we need to know the position of the real generated particle in the middle of a cell. The sim::Particle class provides the position of the track in some point whose $z$-coordinate unnecessarily correspond to the middle point of a cell. So we extrapolate the position of the particle in $xy$-plane in a certain cell using the two closest known points, connecting them with a straight line.

Then, we calculate the distance in the $xy$-planes from this point to the position of the reconstructed track at the same $z$, doing this plane by plane, for each track.

In figure 10 there are histograms to compare the spatial efficiency of KalmanFilter3D compared to KalmanTrackMerge. We can compare this results also with the dimension of a cell, that is 3.9 cm wide: the behaviors are very similar (even if KalmanTrackMerge seems to work a little better), and both of them usually find a point at least within the right cell.

## 4.3 Near detector

As regards the near detector, the main difference is that it contains is a muon catcher at the end that consists the same pairs of $x$-planes and $y$-planes, spaced out by a steel layer 5 cm wide. This make impossible to reconstruct the energy of a particle from the length of its track: the average $\frac{dE}{dx}$ of the muon catcher is higher than the rest of the detector due to the presence of steel.




| Method | Mean | Sigma |
|---|---|---|
| Energy from track length | 3.48 % | 3.81 % |
| Energy from KalmanFilter3D | 3.38 % | 3.94 % |

Figure 6: Fraction of energy reconstructed in the far detector and fit parameters. Only the peaks (without tails) have been fitted

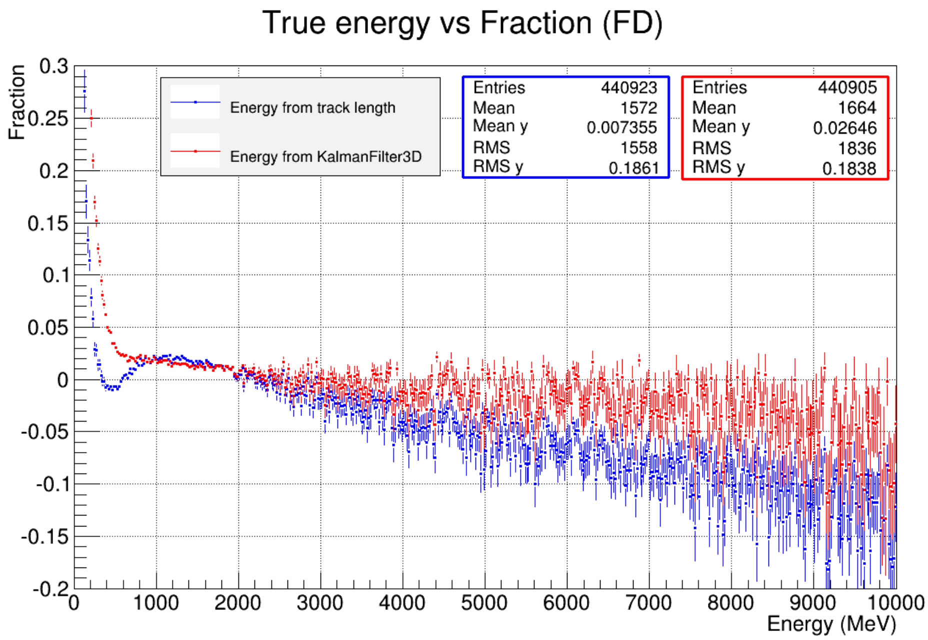



Figure 7: Fraction of energy reconstructed in the far detector

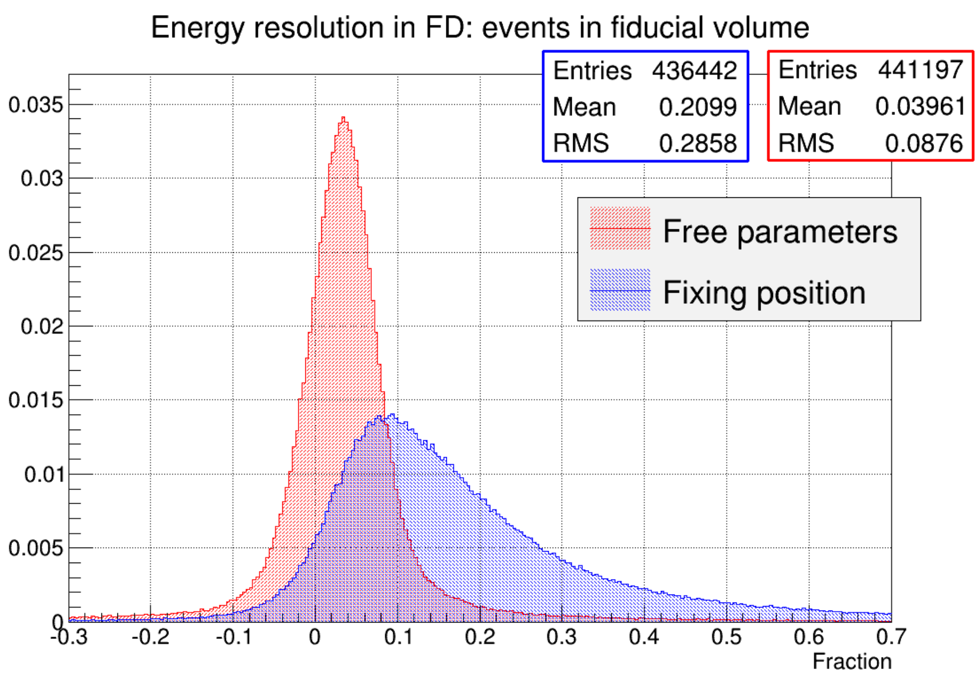



Figure 8: Fraction of energy reconstructed in the far detector with fixed parameters

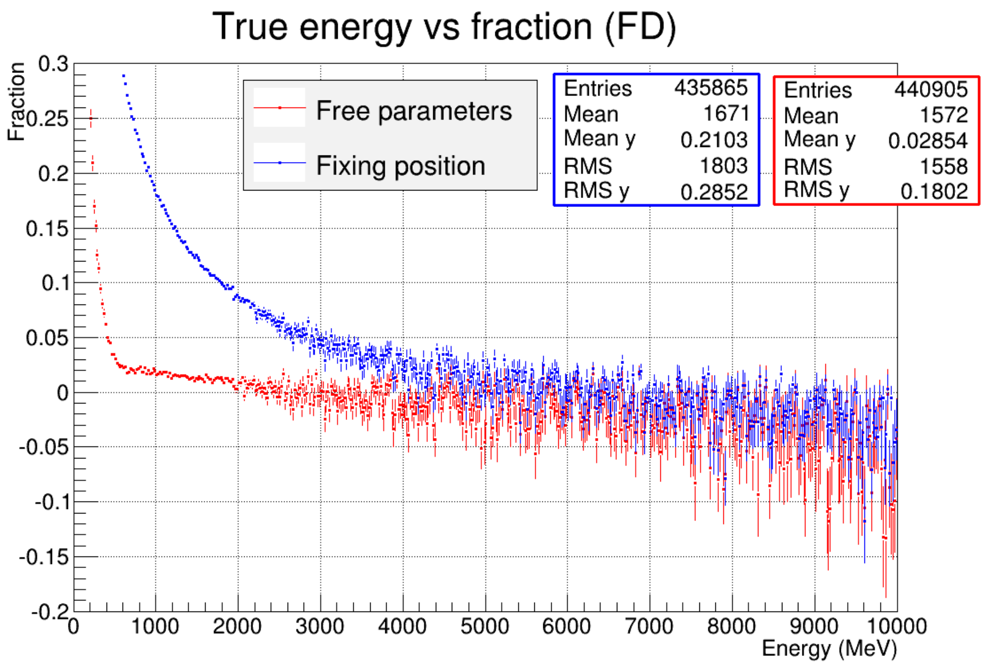



Figure 9: Fraction of energy reconstructed in the far detector with fixed parameters

In figures 11 and 12 we are showing only the energy of a particle as measured by KalmanFilter3D, both with fix parameters and with fixed track. It's easy to see that in this configuration (i.e. consisting only on about half the planes plus the muon catcher), the near detector cannot contain muons with energy $> 2$ GeV. The algorithm tends to underestimate the energy of this particles, and so it's explained the peak in $-0.9$ in figure 11. In the more interesting range of energies $[0.5\,\text{GeV}, 2\,\text{GeV}]$ we can see that KalmanFilter3D works very well.

Eventually, a tentative to reconstruct the track from the length has been made splitting tracks in two parts: the one contained in the standard detector, and the other contained in the muon catcher. Assuming the muon catcher to be made of 40% by steel and 60% by scintillator (cf. figure 14), as a very rough approximation we can assume the average $\frac{dE}{dx}$ of the muon catcher to be

$$\frac{dE}{dx} = 0.4 \cdot \left.\frac{dE}{dx}\right|_{steel} + 0.6 \cdot \left.\frac{dE}{dx}\right|_{scintillator} = 7.253\,{}^{\text{MeV}}/_{\text{cm}}$$

Actually $\left.\frac{dE}{dx}\right|_{steel} = 15.0\,{}^{\text{MeV}}/_{\text{cm}}$ isn't very accurate, moreover as the choice of 40% and 60%, but in figure 13 we can see that the in the range of energies $[0.5\,\text{GeV}, 2\,\text{GeV}]$ the response of this reconstruction is flatter than KalmanFilter3D. A better evaluation of the average $\frac{dE}{dx}$ hasn't been done, but we expect to get even better results.

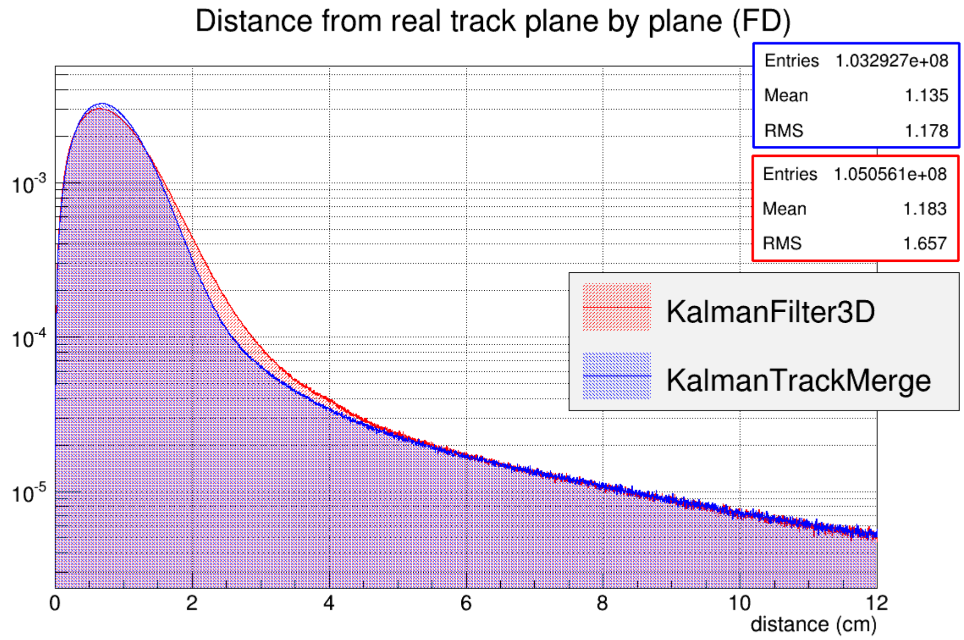



Figure 10: Histograms of the distance between real and reco track using KalmanFilter3D and KalmanTrackMerge in the far detector

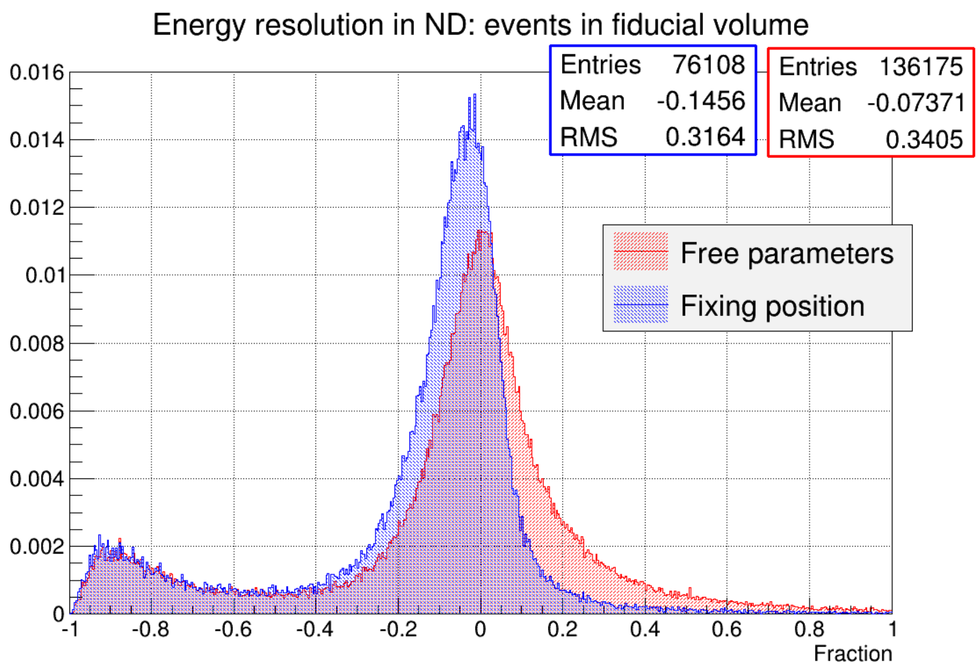
### 4.3.1 Position

The shapes of the histograms in figure 15 are very similar of those measured in the far detector (figure 10). The two only remarkable things are a strange change of slope near 1.5 cm for KalmanTrackMerge, and the presence of very small peaks in 3.9 cm, 7.8 cm and 11.7 cm for KalmanFilter3D.

While is quite difficult to figure out the reason of the former, to explain the latter we can note that 3.9 cm is one of the dimension of a PVC cell. [5] When a particle passes a plane through the PVC between two cells (that have a thickness of $\sim 3.5$ mm [6]), it doesn't emit any $\gamma$ in the scintillator being it a dead zone. In this case, KalmanFilter3D, minimizing its $\chi^2$ function, finds a path on the dead zone next to the true one (figure 16). It's unlikely that it chooses the second dead zone or a farther one but sometimes it can happen: this explains the peaks on 7.8 cm and 11.7 cm. Actually this effect is weakly present also in the far detector, nearby 4 cm.

However this doesn't happen for the KalmanTrackMerge tracks.

In order to provide a better explanation, in figure 17 there are two similar histograms, that show separately the distances between real and reco track in $x$ and $y$ coordinates, respectively in $x$-planes and $y$-planes, using KalmanFilter3D. Distances are defined as $\Delta x = x_{reco} - x_{real}$ and $\Delta y = y_{reco} - y_{real}$. We can see that the mean on the $x$-planes (0.73 mm) is greater than $y$-planes (0.49 mm), and the histograms aren't symmetrical with respect to 0 cm.




| Method | Mean | Sigma |
|---|---|---|
| Energy with fixed track | $-3.5\,\%$ | $7.3\,\%$ |
| Energy with free parameters | $-0.1\,\%$ | $9.8\,\%$ |

Figure 11: Fraction of energy resolution in the near detector and fit parameters. Only the peaks (without tails) have been fitted. In blue with the length interpolation, in red with the Kalman Filter

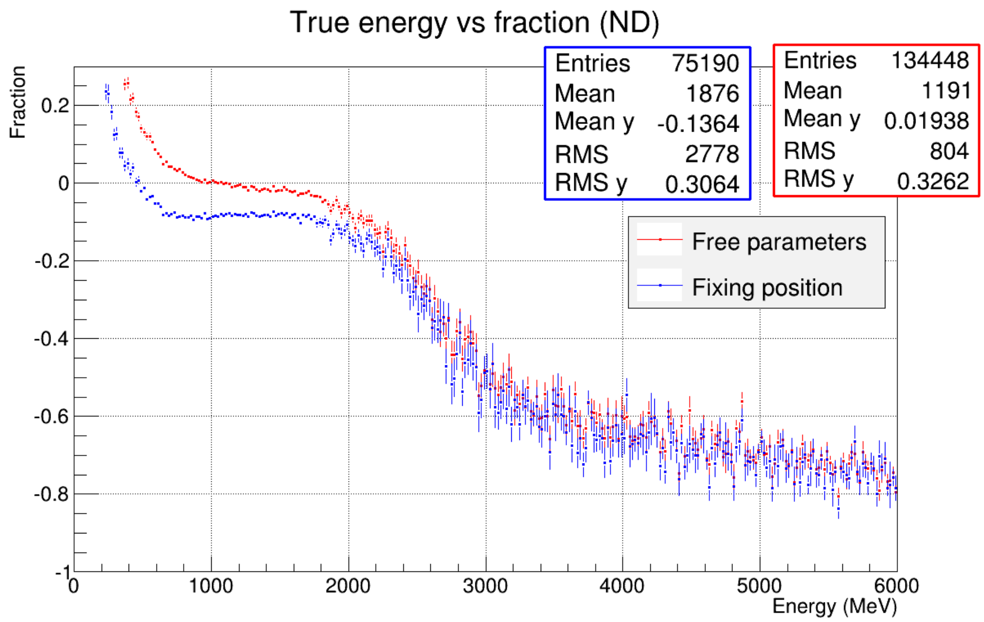



Figure 12: Fraction of energy resolution in the near detector as function of the reconstructed energy.

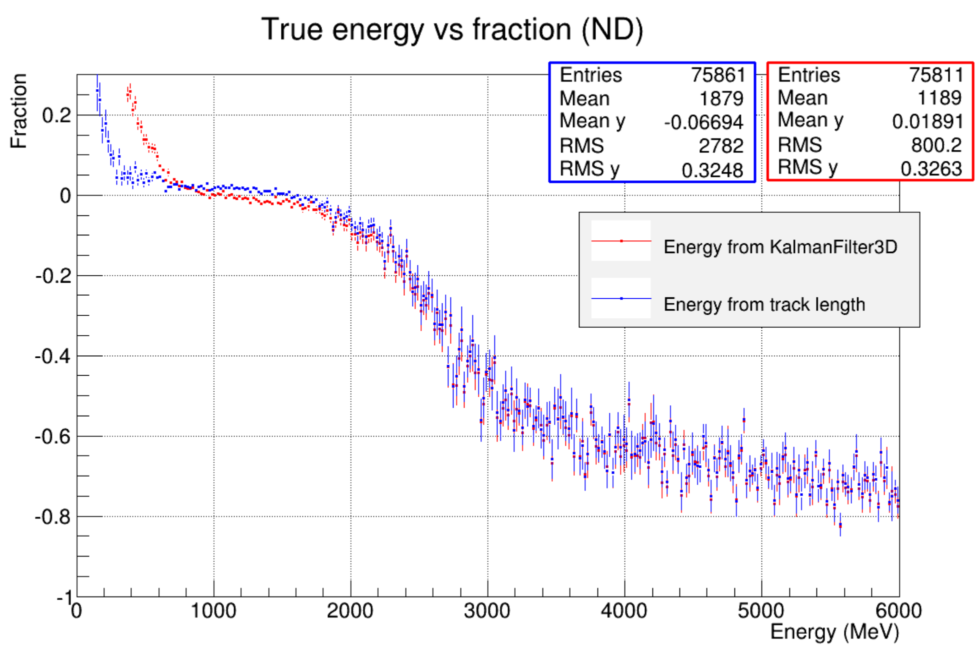



Figure 13: Fraction of energy resolution in the near detector as function of the reconstructed energy. In blue with the length interpolation, in red with the Kalman Filter.

Figure 14: Muon catcher in the near detector

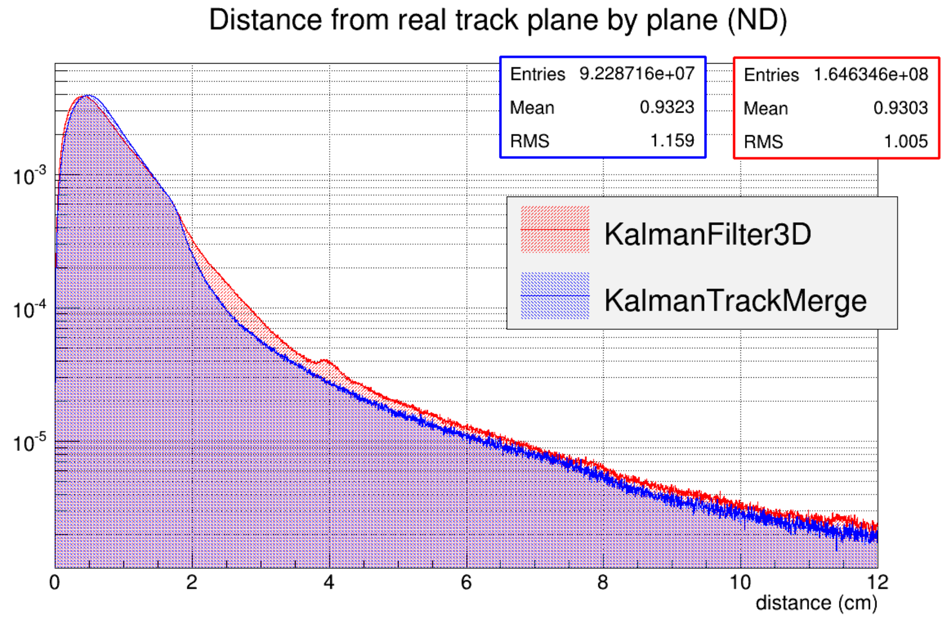



Figure 15: Histograms of the distance between real and reco track in $xy$-plane using KalmanFilter3D and KalmanTrackMerge in the near detector

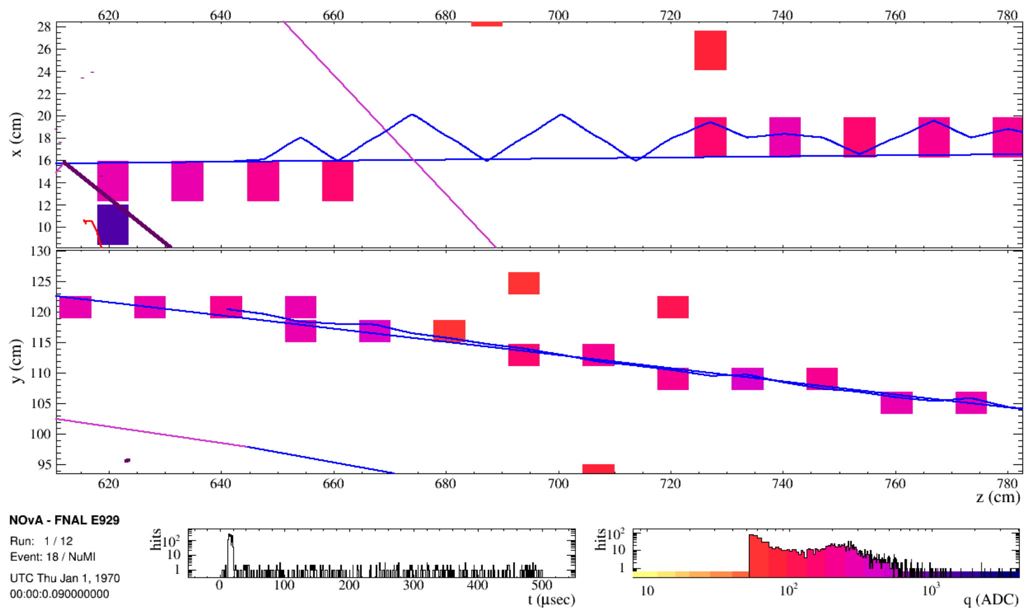



Figure 16: Event where the particle passed through dead zone, and KalmanFilter3D failed to reconstruct it

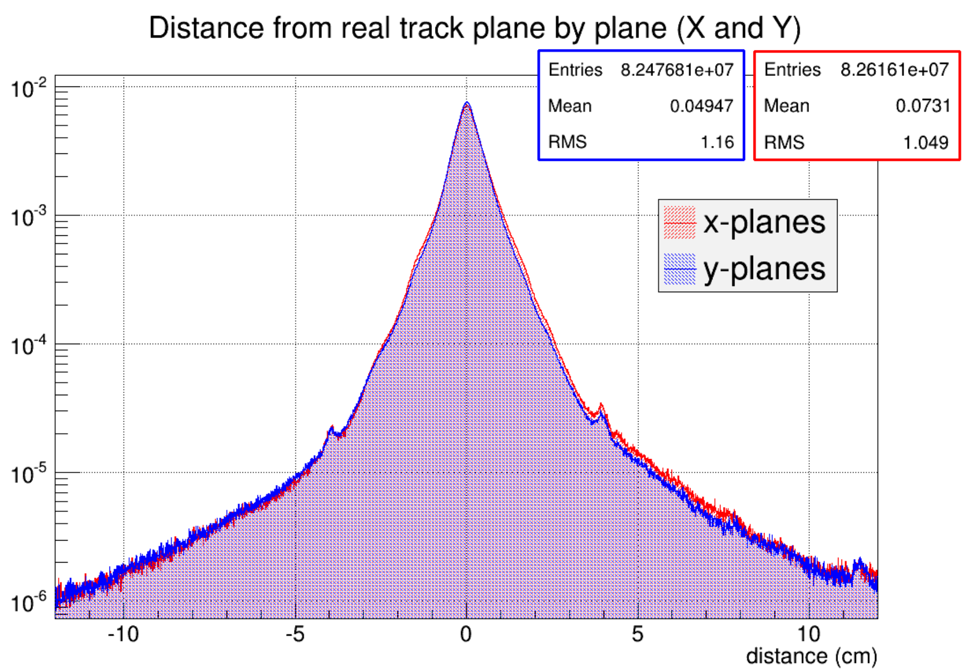



Figure 17: Histograms of the distance between real and reco track separately in $x$ and $y$ coordinates, using KalmanFilter3D

| Particle | Percentage in FD | Percentage in ND |
|---|---|---|
| $\mu^-$ | $93.8 \pm 1.7\,\%$ | $97.5 \pm 1.7\,\%$ |
| $\gamma$ | $2.3 \pm 0.3\,\%$ | $0.8 \pm 0.1\,\%$ |
| $p$ | $1.9 \pm 0.3\,\%$ | $1.0 \pm 0.2\,\%$ |
| $\pi^+$ | $1.6 \pm 0.2\,\%$ | $0.7 \pm 0.1\,\%$ |

Table 1: Particles that have the highest purity with the most energetic track in a slice

# 5 Muon tagging

In a real event we don't know which particles have generated a certain track. A very simple way to tag a $\mu$ in a slice can be done choosing the most energetic track.

To check the efficiency of this hypothesis, we applied this tagging system at some events in both the far detector and the near detector with a beam of $\nu_\mu$ (i.e. forward horn current), taking in account only slices with all the tracks contained in the fiducial volume and with at least a real $\mu^-$ in the Montecarlo simulation.

Results of the analysis are displayed in table 1.

# 6 Conclusion

In this paper we've presented the behaviors of KalmanFilter3D in neutrino events reconstruction. The results for the far detector are good. As regard energy reconstruction, KalmanFilter3D works a little better than the other algorithms, especially being flatter in figure 7 in the most interesting range of energy. On the other hand, position reconstruction works a little worse than KalmanTrackMerge.

Energy reconstruction is good also in near detector, but we have shown that a simple algorithm with rough parameters, that uses average $dE/dl$, gives better results than KalmanFilter3D (figure 13). Position reconstruction in near detector is good but present some strange peaks that have to be investigated.

We've proposed also a new tagging algorithm in chapter 5, that seems to be very efficient even though simple, especially in near detector.

# 7 What's next

- Try to explain the strange shape of blue histogram in figure 8.
- The bug that cause peaks in figures 15 and 17 should be fixed.
- Understand why histograms in figure 17 aren't symmetrical with respect to 0: they have more events on the positive axes for both $y$-planes and $x$-planes. A possible explanation can be hidden in how the algorithm compensate the number of photoelectrons as function of the distance from the APDs.
- Adjust the average energy loss in muon catcher, to improve energy resolution from length in near detector (blue dots in figure 13).
- It should be useful to see how the values in table 1 depend on particle momentum.

# References

[1] http://www-numi.fnal.gov/

[2] Letter of Intent to build an Off-axis Detector to study neutrino oscillations with NuMI Neutrino Beam (2002)

[3] Eli Brookner, Tracking and Kalman Filtering Made Easy (1998)

[4] Denis Perevalov, New Kalman Filter (2013)

[5] Ken Heller, PVC Extrusion Modules (2009)

[6] Sarah R Phan-Budd, PVC Extrusion Dimensions Validation (2012)