WA105

Benchmarking distributions

Elisabetta Pennacchio, IPNL

WA105 SB meeting, July 6th 2016

Outline:

1. Description of benchmarking distributions
2. How to get and run the code
3. Purity measurement from muon tracks

1. Description of benchmarking distributions

6.3 Benchmarking

In order to perform a systematic validation of the software after each new development and to evaluate the impact of any new code addition on the physics results and on program stability, a set of benchmarking procedures and testing tools was defined. The basic idea was to develop an automatic and quick procedure to be executed to validate new software releases and every modification where main structural or coding changes were introduced.

The input to this benchmarking system is a set of fundamental distributions sensitive to basic quantities related to simulation effects and reconstruction performance produced by a standard simulation setup, in particles gun mode, and a reference analysis program. The analysis of these distributions and the comparison with the reference set of histograms corresponding to previous stable versions of the code helps in evaluating software improvements and in detecting and fixing unexpected problems.

This procedure has been set up for both the simulation/reconstruction QSCAN packages of the $3 \times 1 \times 1$ m^3 and $6 \times 6 \times 6$ m^3 detectors. The benchmarking procedure examines several basic analysis quantities at the simulation and reconstruction level. At the GEANT simulation level, about 10

spsc report, April 2016

Links to previous presentations:

- Identification of benchmark histograms and control samples to check simulation results
  SB meeting, October 7th,2015
  https://laguna.ethz.ch/indico/getFile.py/access?contribId=5&resId=0&materialId=slides&confId=161

- News on software utilities (related to the design of the software versions validation system)
  SB meeting, October 21st 2015
  https://laguna.ethz.ch/indico/getFile.py/access?contribId=1&resId=0&materialId=slides&confId=165

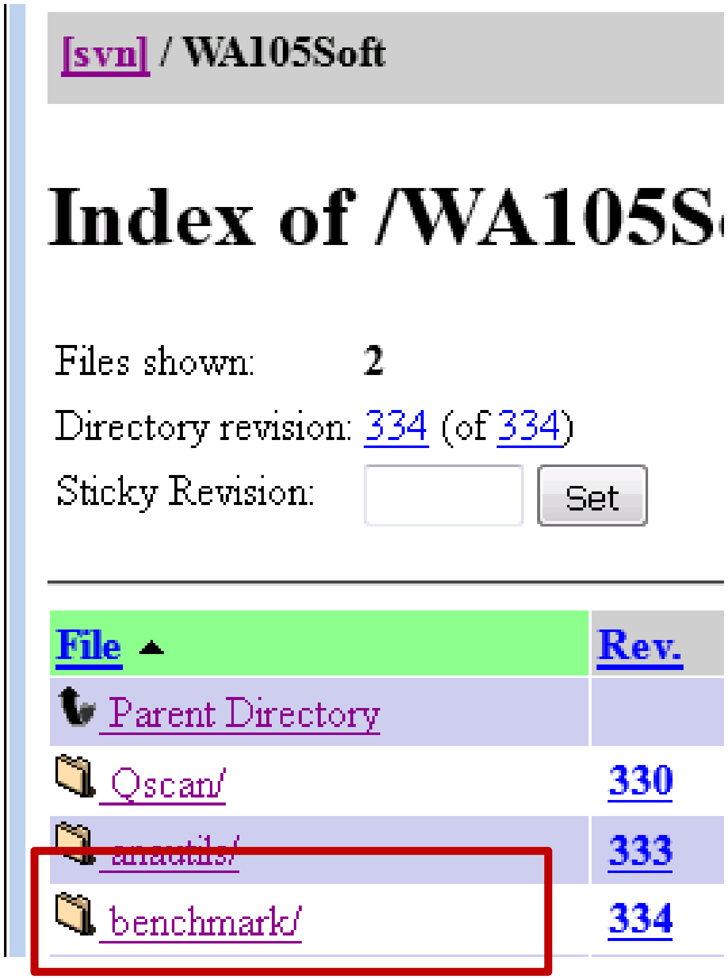
- Benchmarking distributions of QSCAN
  SB meeting, November 18th 2015
  https://laguna.ethz.ch/indico/getFile.py/access?contribId=0&resId=0&materialId=slides&confId=175

- Benchmarking distributions of QSCAN (and some technical information)
  SB meeting, December 2nd 2015
  https://laguna.ethz.ch/indico/getFile.py/access?contribId=4&resId=0&materialId=slides&confId=177

- Update on hit reconstruction
  SB meeting January 27th, 2016
  https://laguna.ethz.ch/indico/getFile.py/access?contribId=3&resId=0&materialId=slides&confId=185

- Software organization at CCIN2P3 and CERN
  WA105 General Meeting, March 8th, 2016
  http://laguna.ethz.ch/indico/getFile.py/access?contribId=3&sessionId=4&resId=0&materialId=slides&confId=170

The code is available on the svn server:



How to compile and run

```
s -rtl WA105Soft/benchmark/

2 pennacc lbno 512 Jul  4 09:42 src
2 pennacc lbno 512 Jul  4 09:42 inc
1 pennacc lbno 419 Jul  4 09:42 Makefile
```

```
ls -rtl

: 2 pennacc lbno    512 Jul  4 09:49 inc
- 1 pennacc lbno    419 Jul  4 09:49 Makefile
: 2 pennacc lbno    512 Jul  4 09:51 src
: 1 pennacc lbno 303504 Jul  4 09:51 bench.exe
```

```
bash-4.1$ bench.exe -h
analysis program to read root files produced with Qscan and root files obtained running  reconstruction (recotask)
Syntax:
           bench.exe -f afile.root -p ipass [-e nev]


 Options :
             -f input filename : afile.root
              -p ipass is an integer, to select  different levels of analysis
                 ipass=0 only a dump of the run header is produced
                 ipass=1 histograms related to event generation are filled
                 ipass=2 histograms related to RawData are produced
                 ipass=11 histograms related to Hit Reconstruction are filled
                 ipass=12 histograms related fo 2DTrack Reconstruction are filled
             [] denotes an optional argument

               -e :  number of events to process
                     (optional, default: process all events)

 2 output files are obtained:
    1) bench_afile_ipass.root: histograms file (only if pass>0)
    2) bench_afile_ipass.listing  dump of RunHeader and of some events info
```

Output

Example of execution on raw data (simulation output) for **3x1x1** and **6x6x6** (ipass=2)

```
bash-4.1$ bench.exe -f input/rawdata_311.root -p 2 -e 10
----->file input/rawdata_311.root  will be read
you select ipass=2:  histograms related to  RawData will be filled
```

```
bash-4.1$ bench.exe -f input/rawdata_666.root -p 2 -e 10
----->file input/rawdata_666.root  will be read
you select ipass=2:  histograms related to  RawData will be filled
bash-4.1$ ls -rtl
total 368
drwxr-sr-x 2 pennacc lbno    512 Jul  4 09:49 inc
-rw-r--r-- 1 pennacc lbno    419 Jul  4 09:49 Makefile
drwxr-sr-x 2 pennacc lbno    512 Jul  4 09:51 src
-rwxr-xr-x 1 pennacc lbno 303504 Jul  4 09:51 bench.exe
drwxr-sr-x 2 pennacc lbno    512 Jul  4 11:03 input
-rw-r--r-- 1 pennacc lbno   1574 Jul  4 11:04 bench_rawdata_311_pass2.listing
-rw-r--r-- 1 pennacc lbno  13250 Jul  4 11:04 bench_rawdata_311_pass2.root
-rw-r--r-- 1 pennacc lbno   1807 Jul  4 11:11 bench_rawdata_666_pass2.listing
-rw-r--r-- 1 pennacc lbno  22673 Jul  4 11:11 bench_rawdata_666_pass2.root
bash-4.1$
```

Output files

Dump of output logfile:

Raw data analysis

```
     file  input/rawdata_311.root   will be analized

 ##########################################################

 Folders, Trees, branches available:
  TGeo   1
  RunHeader 1
     -->theRunHeader  1
     -->theGeomConfig 1
     -->theBeamInfo 1
     -->theHitReco 0
  EventTree 1
     -->theEventHeader 1
     -->CRPRawData  1
     -->CRPVoxelData 1
     -->LRORawData 1
     -->Hits  0  0
     -->Tracks 0  0


  ================== RUN HEADER DUMP:

 --> file produced on: Fri Jul  1 19:02:40 2016

 --> run Number 123456

 --> Detector specification:
     Configuration File  WA105_1x1x3_1.config
     purity(ms):  3

     longitudinal diffusion (L)  0
     transerse diffusion (T)     0

 --> Electric field configuration:
     uniform EField, value    500  (V/cm)

 --> No Light Simulation

 --> Particle generation:
     initial seed 123
     input file selection:  mu-,  gun mode

     Geant 3 has been selected

 ========== GEOMETRY CONFIGURATION: WA105_1x1x3_1.config

    DETECTOR WA105_3x1x1
    RAWDATATYPE  USHORT  1
    ELECMODEL ETHZ  5  12
    CRM number 1
      CRM 0  0  0  50  100  300  10  0.3125 0.3125  320  960  1675
    time sampling 0.4
    NOISE not set  0  0  -1

  ==================   EVENT HEADER DUMP: ========================
Total number of events:  4000
 ----> 10 events will be analyzed
```

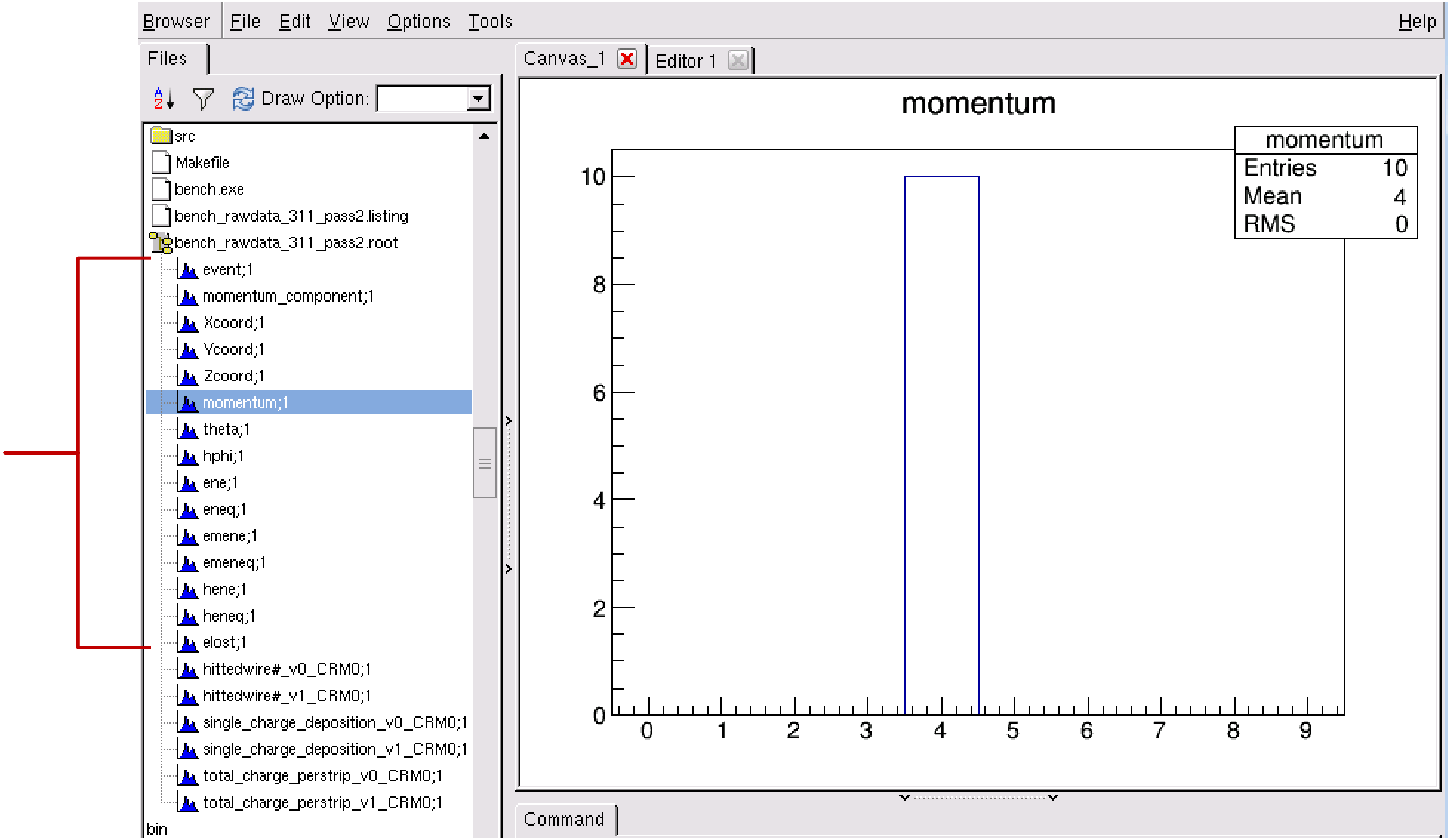
Trees available in the file

Initial conditions for events generation

Geometry configuration

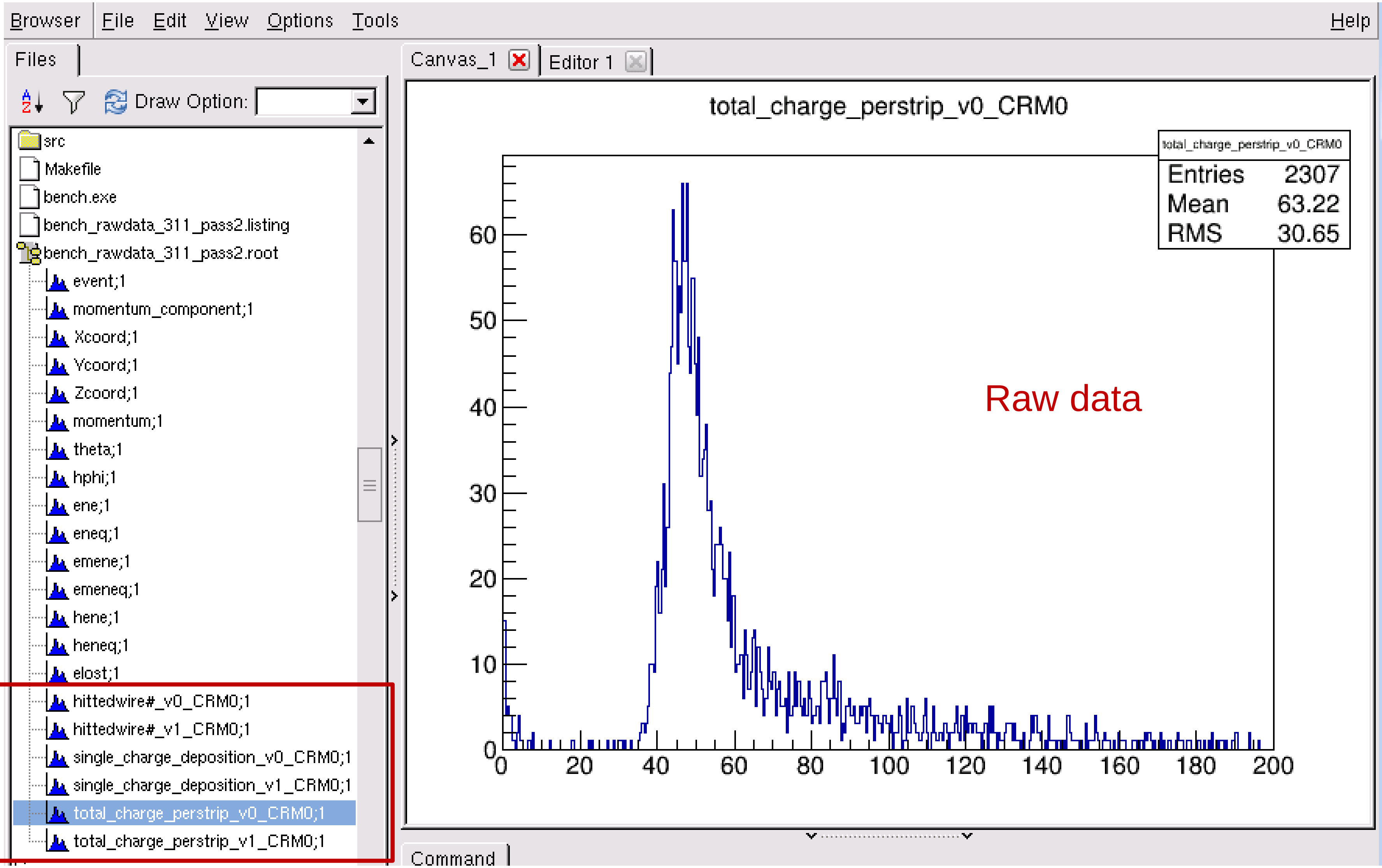
Output ROOT file (examples for 3x1x1):

Raw data analysis



Distributions related to generation quantities

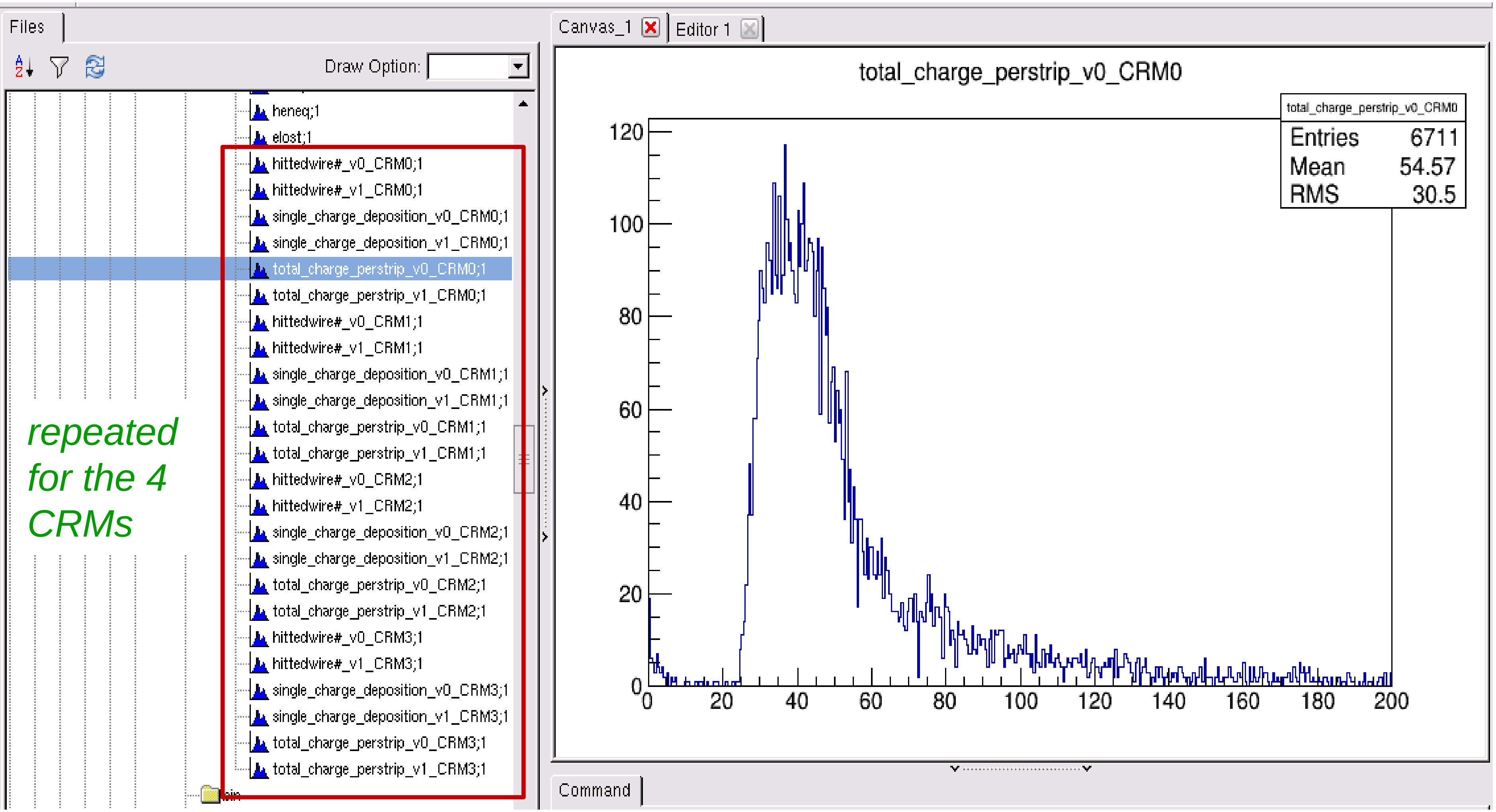
Output ROOT file (examples for 3x1x1):



Raw data: charge depositions on strips, total charge collected by strips for view 0/1

Example of events generated with 6x6x6 geometry :

Raw data analysis



- Benchmark distributions are produced for 4 CRM.
- This is the same for all processing steps *(raw data analysis or reco data analysis)* .
- In the following slides examples on 3x1x1 only will be shown

Example of execution on reconstructed data (ipass=11 and ipass=12)

Reco data analysis

```
bash-4.1$ bench.exe -f input/reco_311.root -p 11 -e 10
----->file input/reco_311.root  will be read
you select ipass=11: histograms related to  Hit  reconstruction   will be filled
bash-4.1$ bench.exe -f input/reco_311.root -p 12 -e 10
----->file input/reco_311.root  will be read
you select ipass=12: histograms related to 2DTrack reconstruction will be filled
bash-4.1$ ls -rtl
total 880
drwxr-sr-x 2 pennacc lbno    512 Jul  4 09:49 inc
-rw-r--r-- 1 pennacc lbno    419 Jul  4 09:49 Makefile
drwxr-sr-x 2 pennacc lbno    512 Jul  4 09:51 src
-rwxr-xr-x 1 pennacc lbno 303504 Jul  4 09:51 bench.exe
drwxr-sr-x 2 pennacc lbno    512 Jul  4 11:03 input
-rw-r--r-- 1 pennacc lbno   1574 Jul  4 11:04 bench_rawdata_311_pass2.listing
-rw-r--r-- 1 pennacc lbno  13250 Jul  4 11:04 bench_rawdata_311_pass2.root
-rw-r--r-- 1 pennacc lbno   1807 Jul  4 11:11 bench_rawdata_666_pass2.listing
-rw-r--r-- 1 pennacc lbno  22673 Jul  4 11:11 bench_rawdata_666_pass2.root
-rw-r--r-- 1 pennacc lbno   2157 Jul  4 12:07 bench_reco_311_pass11.listing
-rw-r--r-- 1 pennacc lbno  14345 Jul  4 12:07 bench_reco_311_pass11.root
-rw-r--r-- 1 pennacc lbno   2157 Jul  4 12:08 bench_reco_311_pass12.listing
-rw-r--r-- 1 pennacc lbno  24055 Jul  4 12:08 bench_reco_311_pass12.root
bash-4.1$
```

Hit rec.

Track rec.

Now these are 2 different steps→ the code can be modified to run on hit and track reconstruction results in one step.

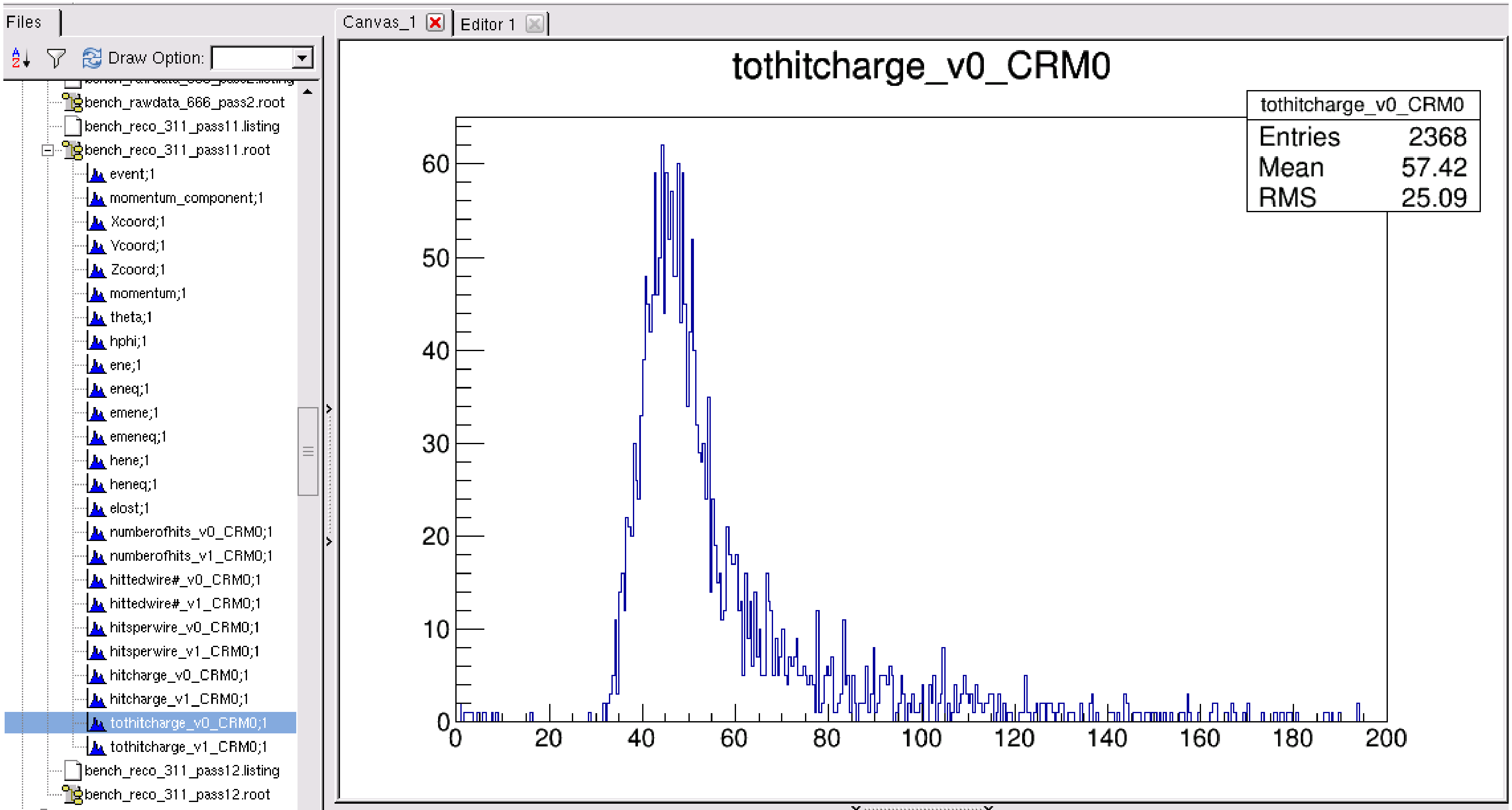
Reco data analysis

Ascii logfiles → name of input file (raw data)
→ date on which the reconstruction was run
→ reconstruction parameters (for tracking some information are to be added)

```
============================  HIT RECONSTRUCTION ===================================

 input file: WA105_311_test1_lt.root
 Algorithm selected  AlgoMultiHit
 hit reconstruction run on :Sat Jul  2 23:35:30 2016

Hit reconstruction parameters :
  2.5  2.5  9999  9999  3  10  10
ROI definition
  20  4  0.5  -0.1  5  10
============================  TRACKING  ===================================
```

```
============================  HIT RECONSTRUCTION ===================================

 input file: WA105_311_test1_lt.root
 Algorithm selected  AlgoMultiHit
 hit reconstruction run on :Sat Jul  2 23:35:30 2016

Hit reconstruction parameters :
  2.5  2.5  9999  9999  3  10  10
ROI definition
  20  4  0.5  -0.1  5  10
============================  TRACKING  ===================================

 input file: WA105_311_test1_lt.root
 Algorithm selected  AlgoMultiCRTrack
 track reconstruction run on :Sat Jul  2 23:35:30 2016

Track reconstruction parameters :
```

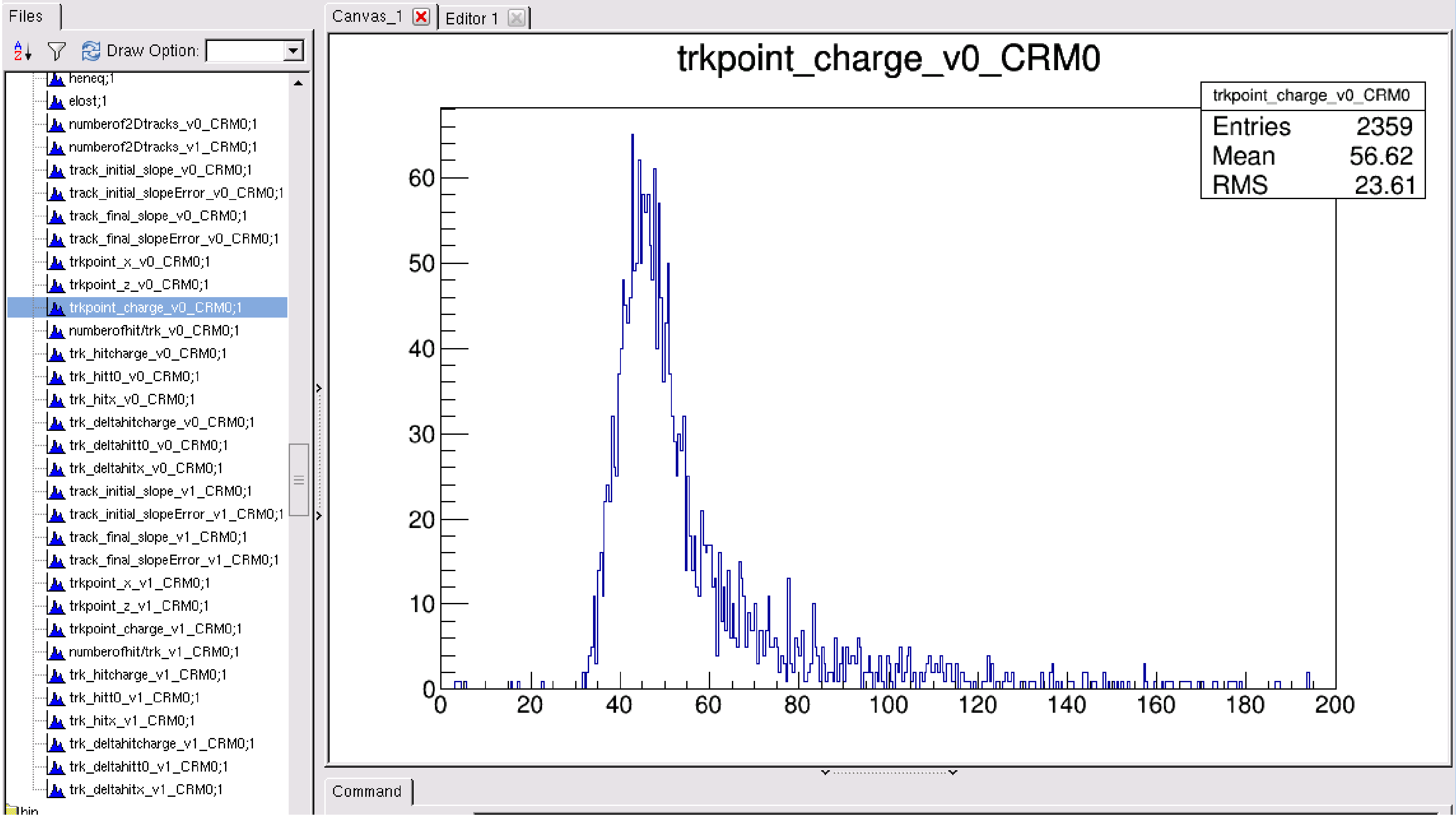
ipass=11, hits reconstruction:



- number of reconstructed hits,
- strips with at least one hit
- # hits/strip,
- hits charge,
- total charge on strips from hits reconstruction

ipass=12, tracks reconstruction:

Reco data analysis



- **Number of tracks**
- **Track slope**
- **Info from points associated to reconstructed track: charge, x(y) position, z position**
- **Info for hits belonging to track : charge, x(y) position, z position**
- **Info for delta rays belonging to track: charge, x(y) position, z position**

Conclusions on benchmarking

- The code is available on the svn
- It provides also examples on how to read the output root files and access stored quantities
- It works both for 3x1x1 and 6x6x6 geometries
- New distributions can be added, following the progress in the reconstructions and needs to check specified distributions

De takuya.hasegawa@kek.jp
Sujet **Benchmarking of our WA105 software**
Pour LBNOdemo-all@cern.ch, lbnodemo-sb-sci-physl@cern.ch

Répondre | Répondre à tous | Tr

Dear Colleagues,

As was discussed at the BI-WEEKLY science board meeting on 7-October,
science board propose to establish the process of benchmarking of our
WA105 software. Please look at the slide

https://laguna.ethz.ch/indico/getFile.py/access?contribId=5&resId=0&materialId=slides&confId=161

https://laguna.ethz.ch/indico/getFile.py/access?contribId=3&resId=0&materialId=slides&confId=161

According to the discussion, we would like to hear your idea on
benchmarking. Your proposal will be compiled and discussed at bi-weekly
meeting.

To avoid unnecessary e-mail traffic, let's utilize e-group lbnodemo-sb-
sci-phys@cern.ch for the communication. If you are interested in this
discussion but not subscribed to lbnodemo-sb-sci-phys@cern.ch, please
subscribe.

Thank you for your contribution in advance.

Best regards
Takuya

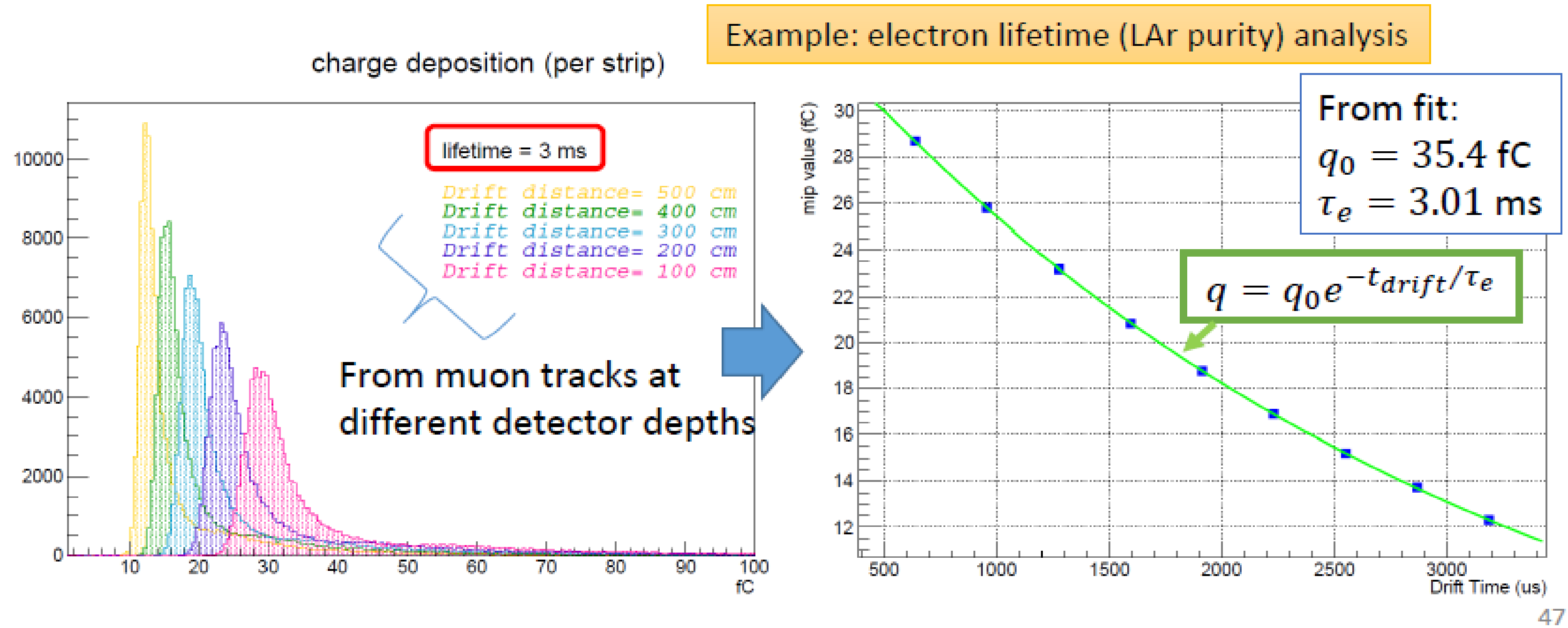
Mail sent of October 7th,2015 by Hasegawa-San

Purity measurement for muon tracks

- One of the task of the online monitoring is the measurement of the purity.

- this measurement is performed using cosmic rays tracks

- The feasibility of this measurement has been tested using raw data (horizontal tracks), and results have been presented at the Science Board meeting (11/18/2015), at the General meeting (03/08/2016), and have also been included in the SPSC report.



SPSC presentation, April 2106

- This study has been repeated with reconstructed tracks.
(the code used for tracking reconstruction is available on the svn)

Analysis methodology

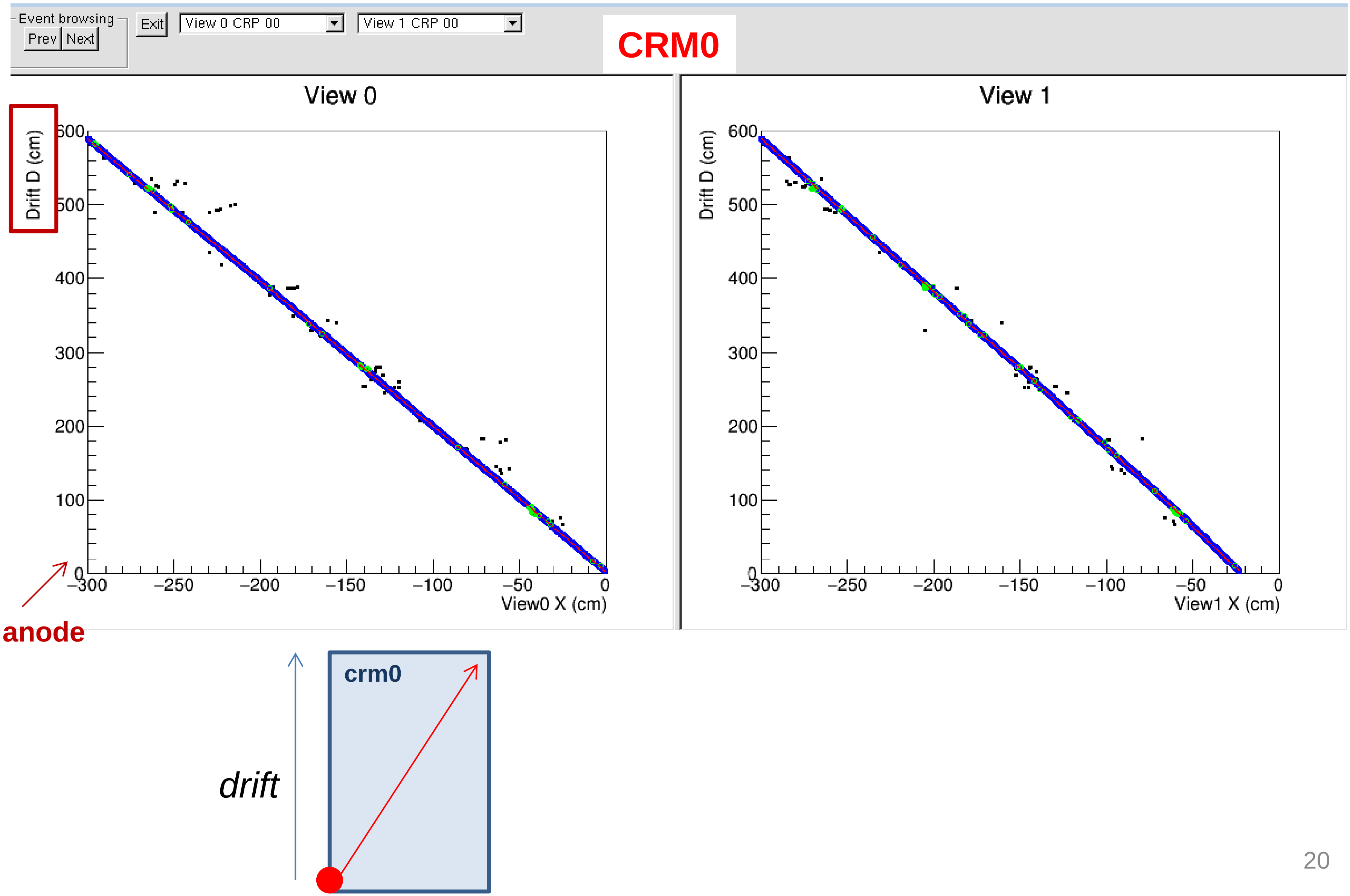
- To set up the method, the 6x6x6 geometry has been used, to exploit the full drift distance.

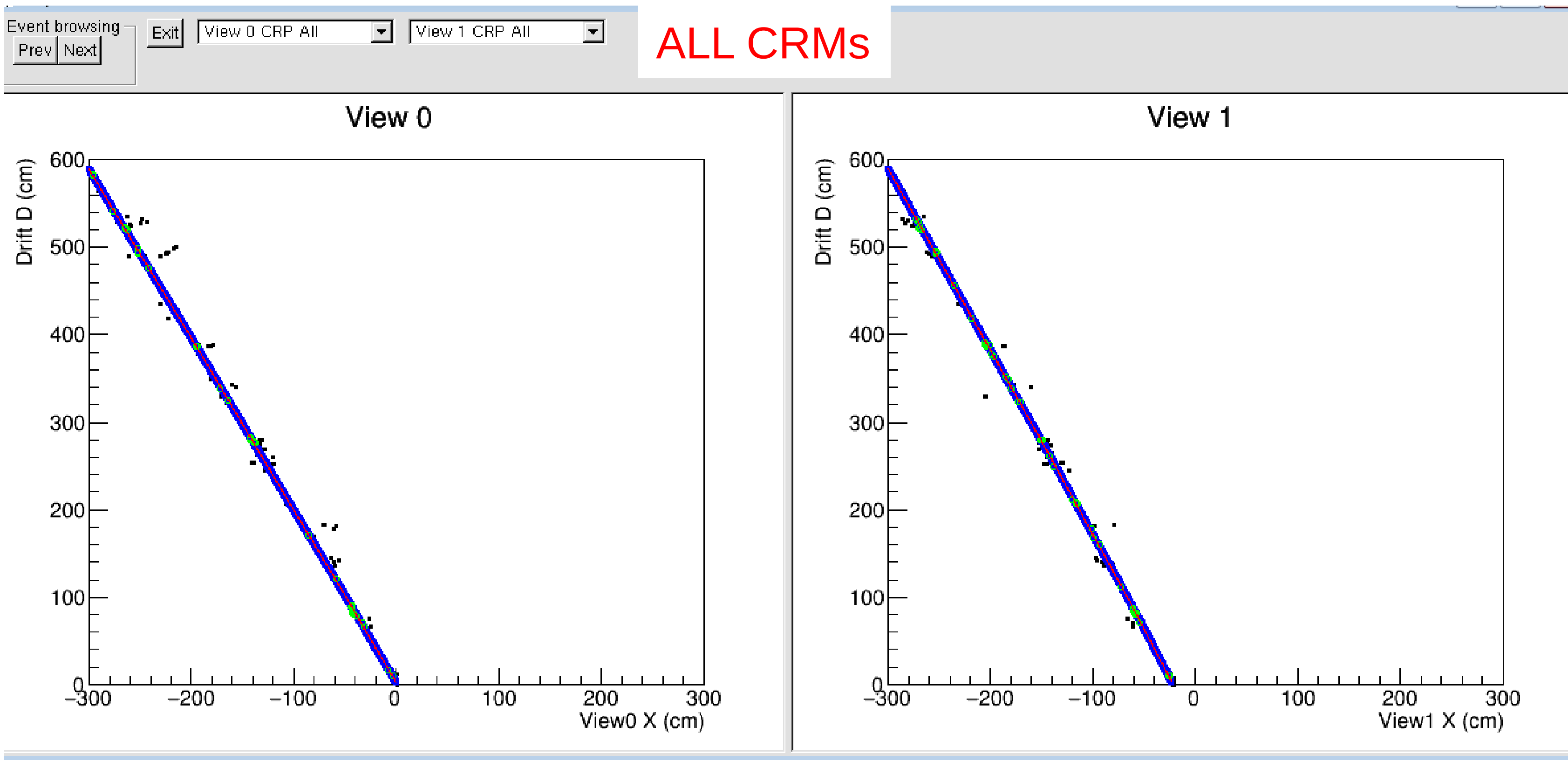
- Only one CRM will be taken into account (CRM0) : our priority is the data taking of September with the 3x1x1 prototype whose anode counts only one CRM

- The method has been tested on samples of muon at different generation angles

- Once the method has been set up, it has been applied to 3x1x1, to check the results

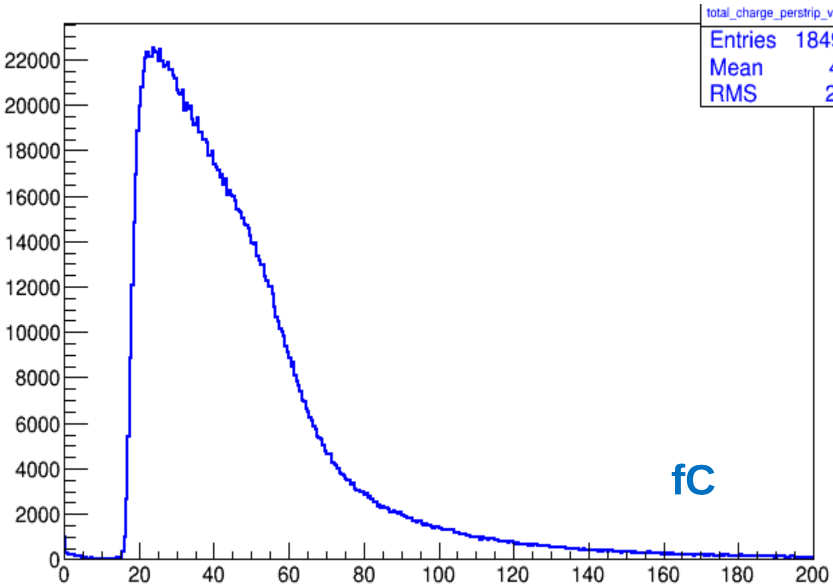
1st sample: muons at 4 Gev, 2K events $\varphi = 45^0$, crossing only CRM0, τ=3ms



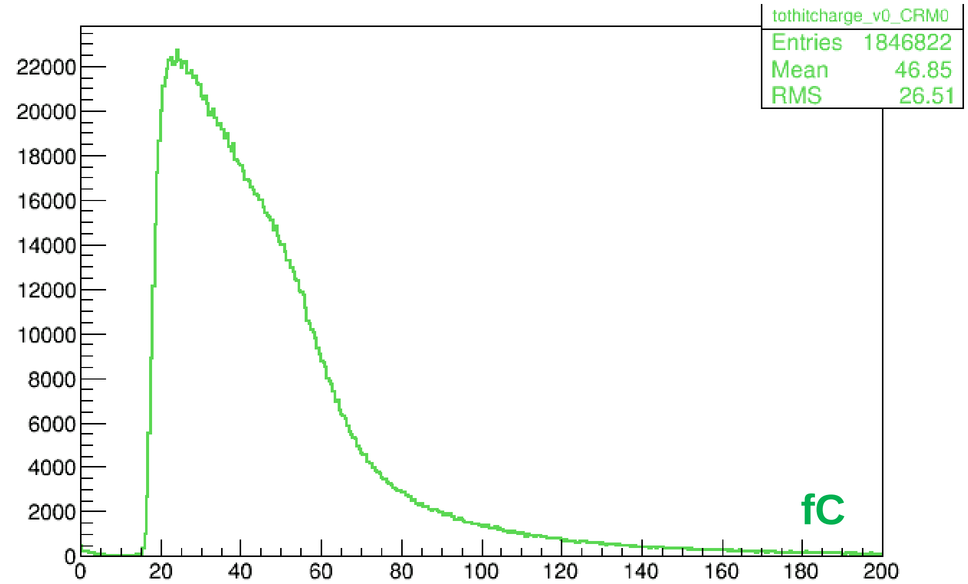

Event browsing
Prev
Next
Exit
View 0 CRP All
View 1 CRP All
ALL CRMs
View 0
Drift D (cm)
600
500
400
300
200
100
0
−300
−200
−100
0
100
200
300
View0 X (cm)
View 1
Drift D (cm)
View1 X (cm)

Some crosschecks on the total charge at raw level, hits and track ….

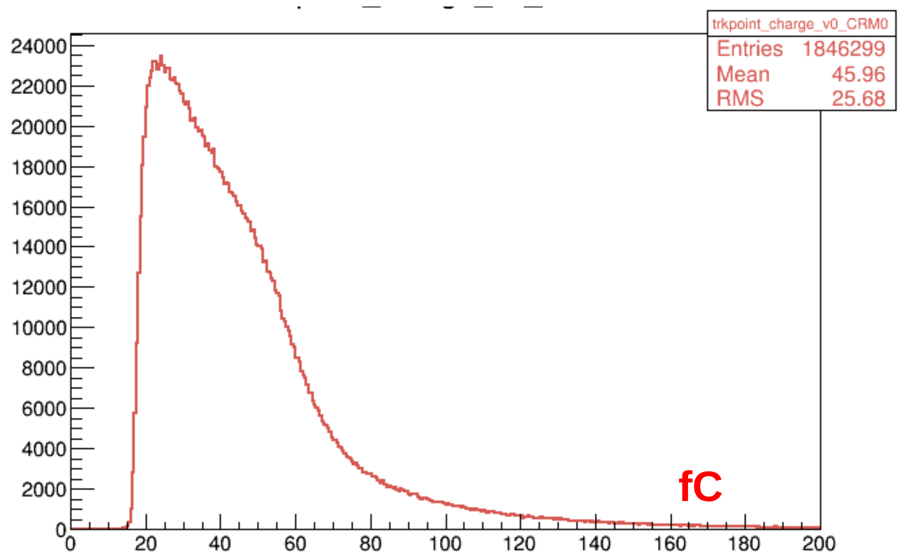
Total charge collected on strips (raw data)



Total charge collected from hit reconstructions

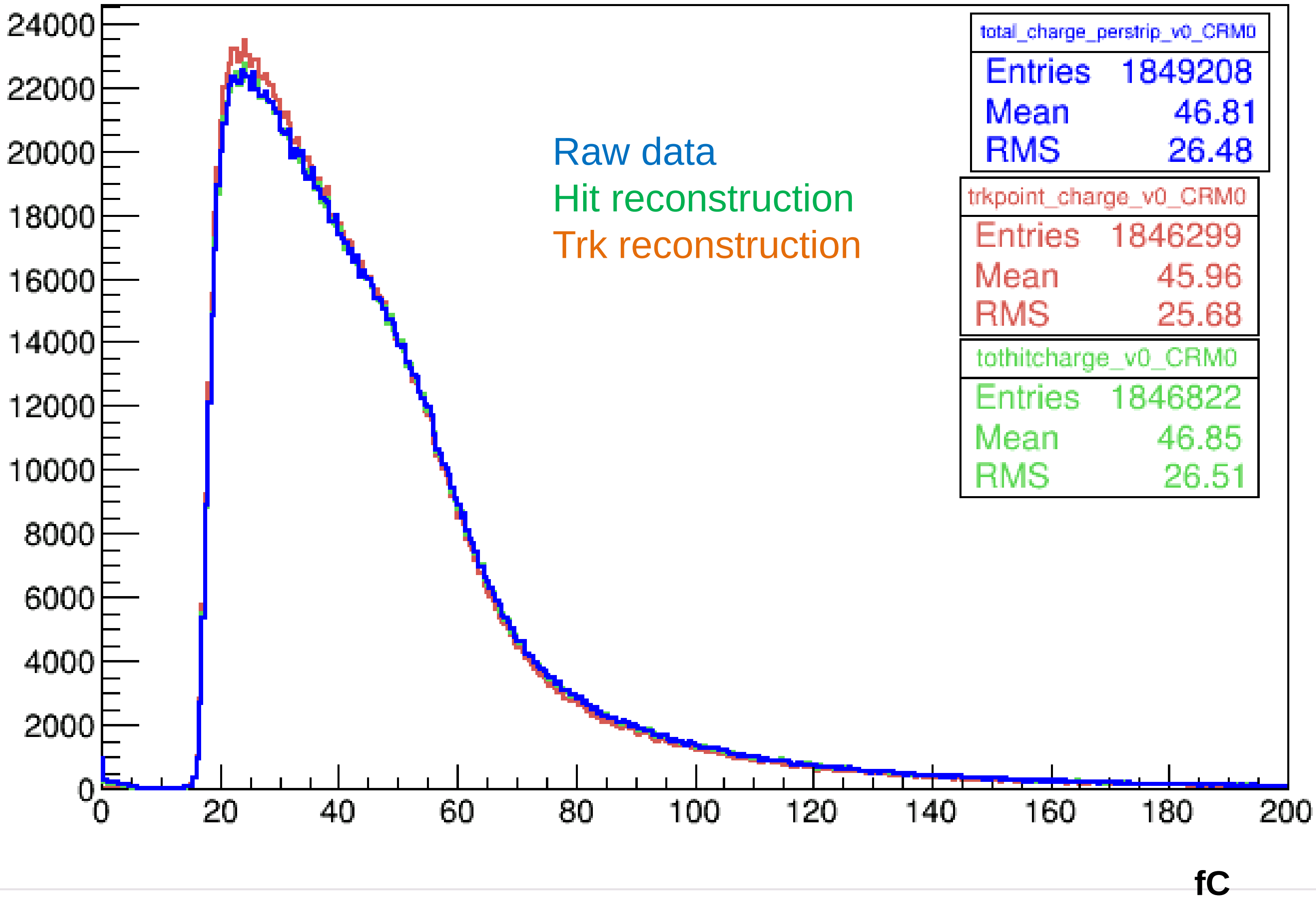


Total charge collected from track reconstruction

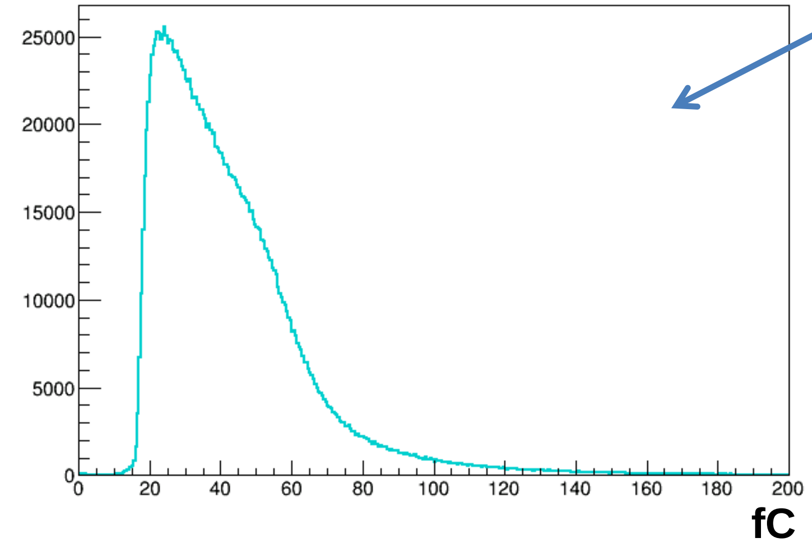


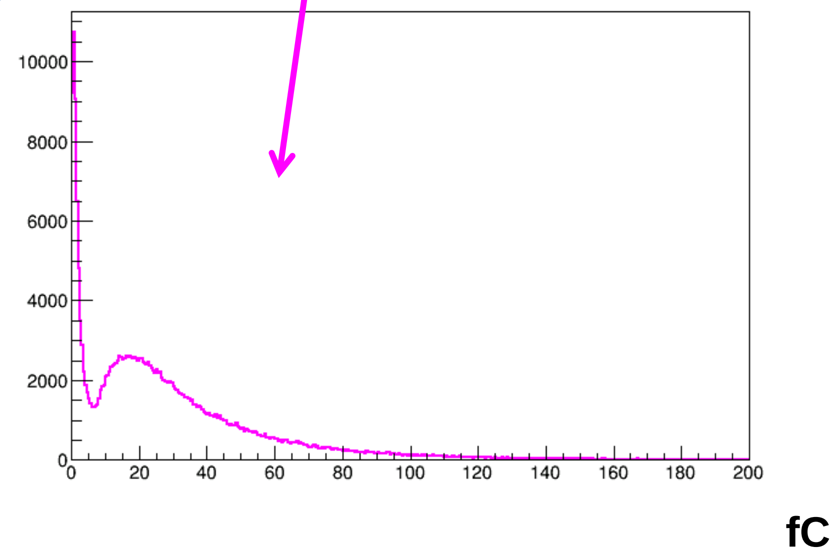
(see later)

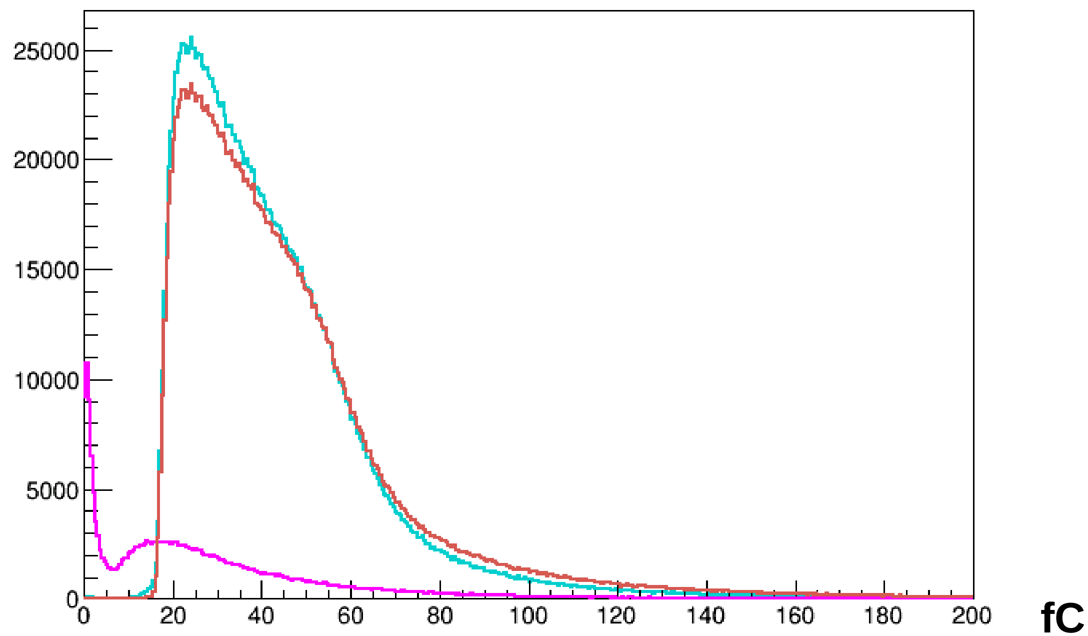
the 3 distributions are in good agreement



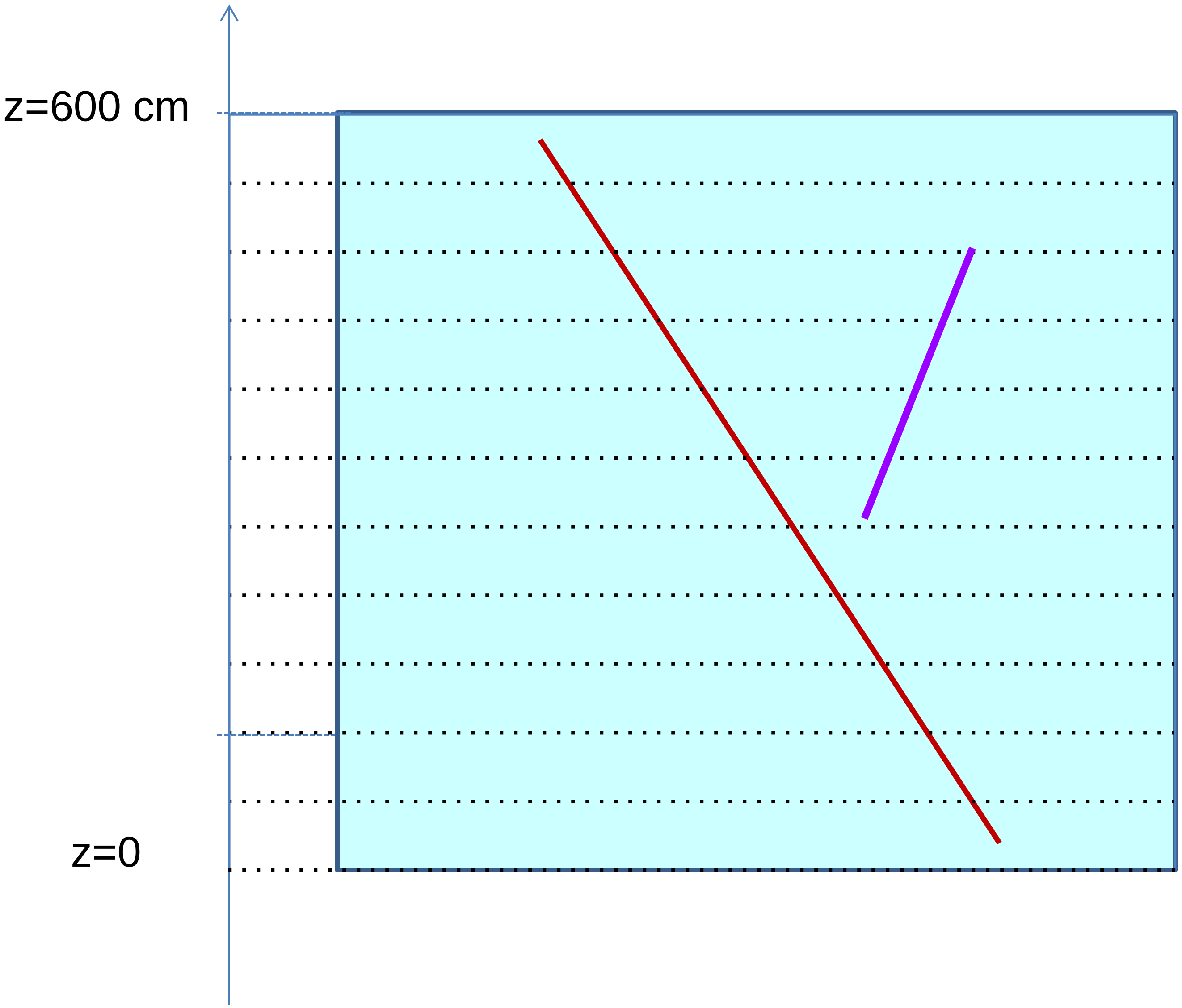
Before moving to the purity measurement, it is useful to remind that the charge collected using track reconstruction information is obtained from hits and delta rays associated to the track







- Due to impurities, the collected charge is a decreasing function of drift time $Q=f(t_{drift})$.
- Points belonging to this function can be represented as $P=(t_{drift}, Q)$;
- To build these points , the drift distance is divided into n equal bins



Let's assume now to divide the full drift distance (6m) in 60 bins

→ Expected loss on one bin ~2%:

1 bin= 10cm, 0,067 ms each → $e^{-(\frac{.067}{3})}$ ~ 98%

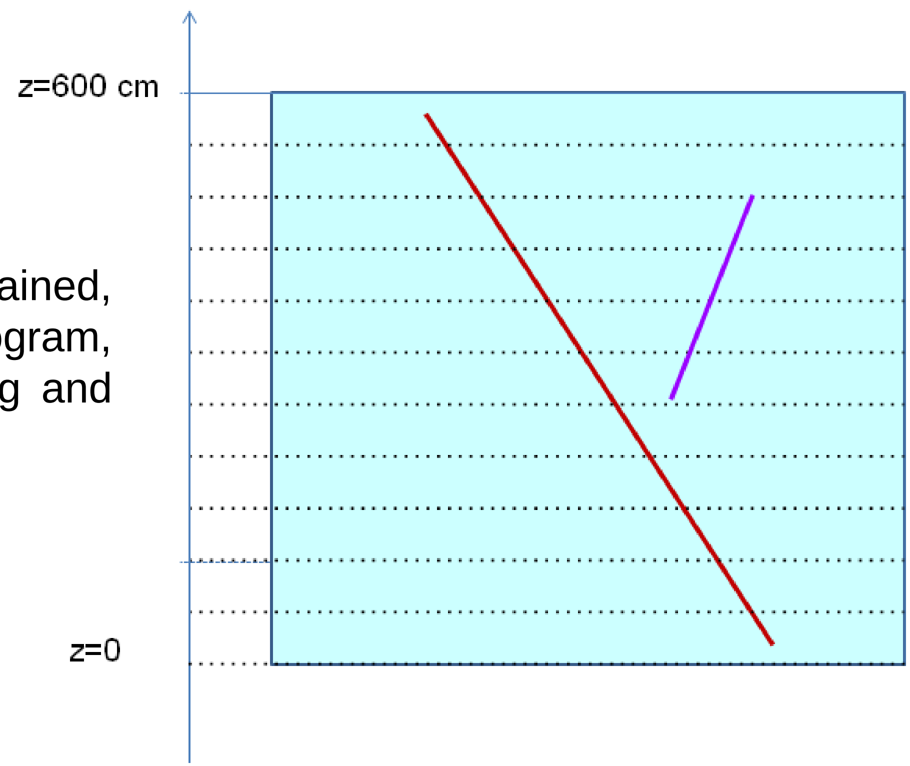
5 cm → $e^{-(\frac{.067}{3})}$ ~ 99%

20 cm → $e^{-(\frac{.134}{3})}$ ~ 95%

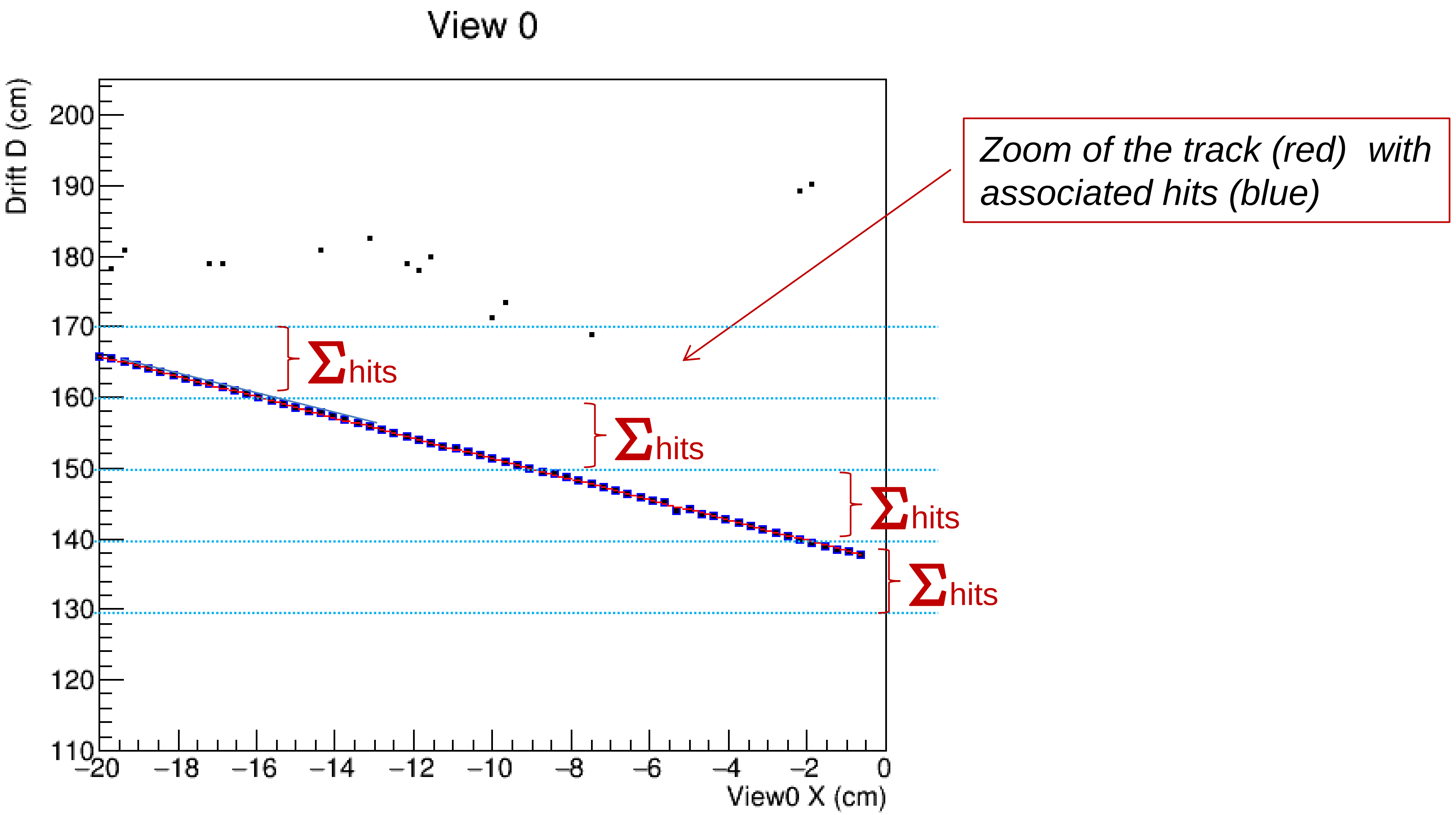
50 cm → $e^{-(\frac{.335}{3})}$ ~ 89%

1 m → $e^{-(\frac{.67}{3})}$ ~ 80%

A set of histograms (60) for each view is obtained, and each track enters in each histogram, depending on its length and on its starting and ending points



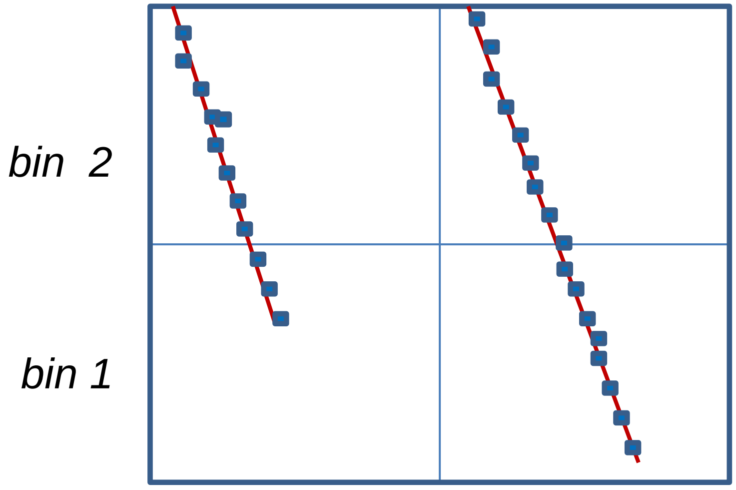
Sum of hits belonging to the same 10 cm bins:



the track is divided in n bins of 10cm in z, and, for each bin, the charge depositions of all hits belonging to the tracks are summed → a “vector” of charge depositions is built

Effect of bin quantization

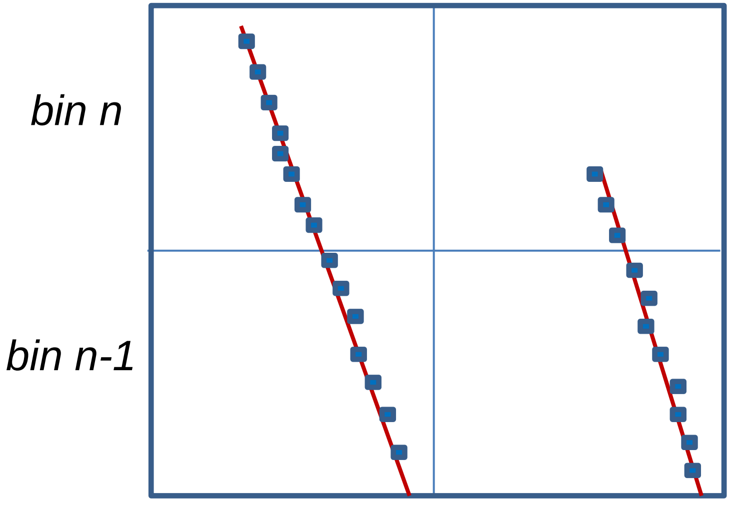
Considering 2 tracks with their first point in bin 1:



→ the charge deposition in the first bin depends on the starting point of the track inside the bin

and their last point in bin n :

The same effect is also present in the last bin.

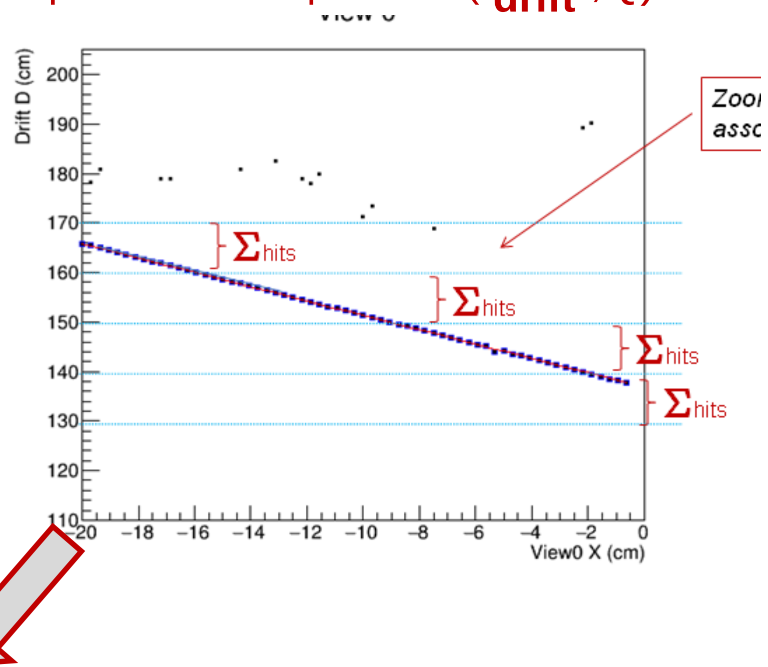


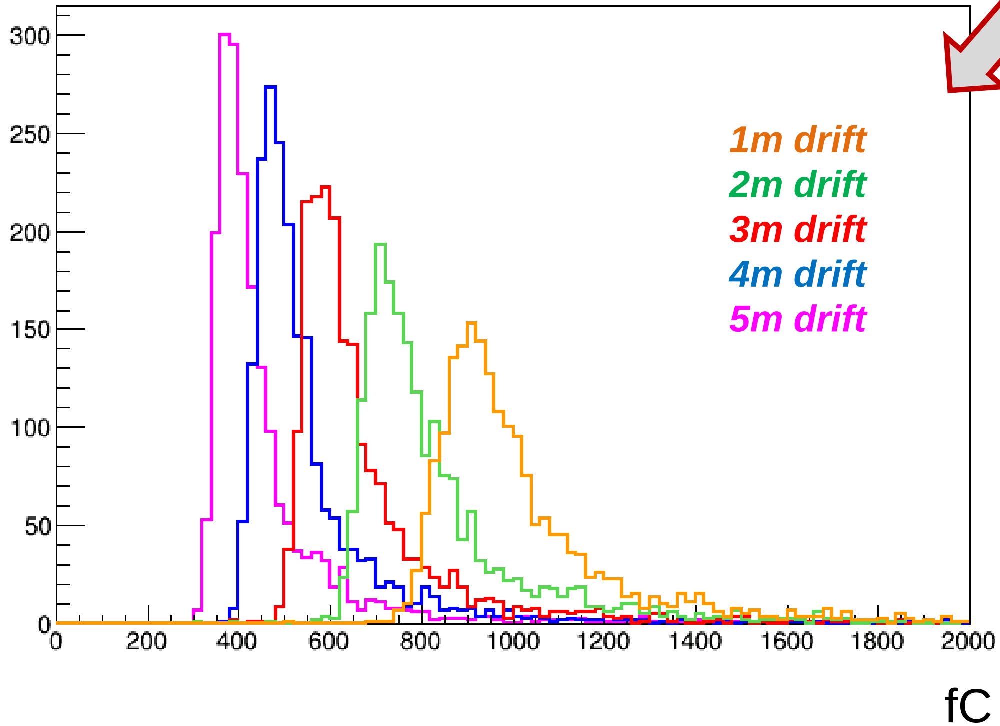
→To avoid disuniformities the first and the last bin of each track are not taken into account

Each component of the charge depositions vector corresponds to one point P=(t_{drift} ,Q) where

t_{drift} = center of the time bin,
135, 145, 155 cm in the example

Q= Σhits charge



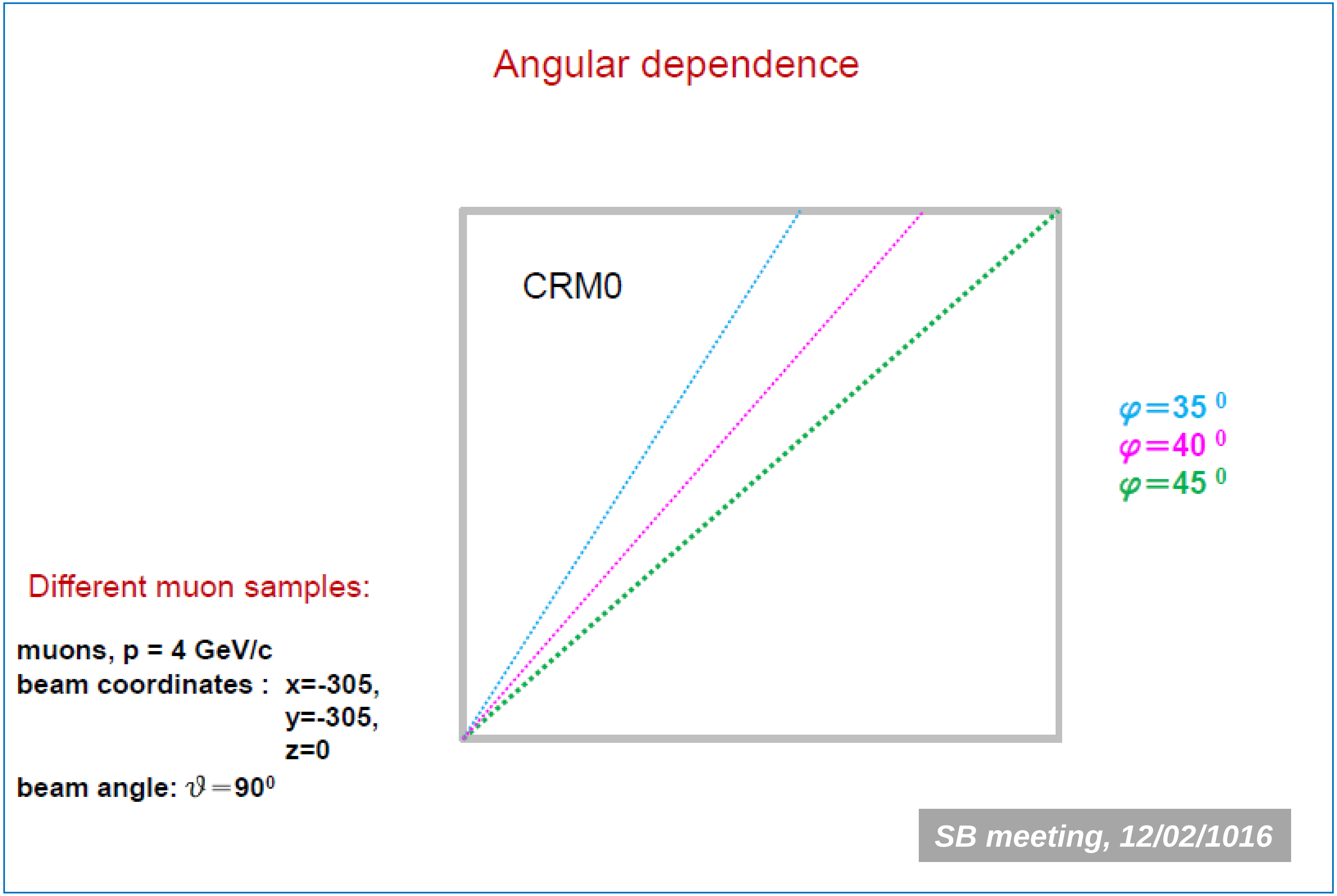


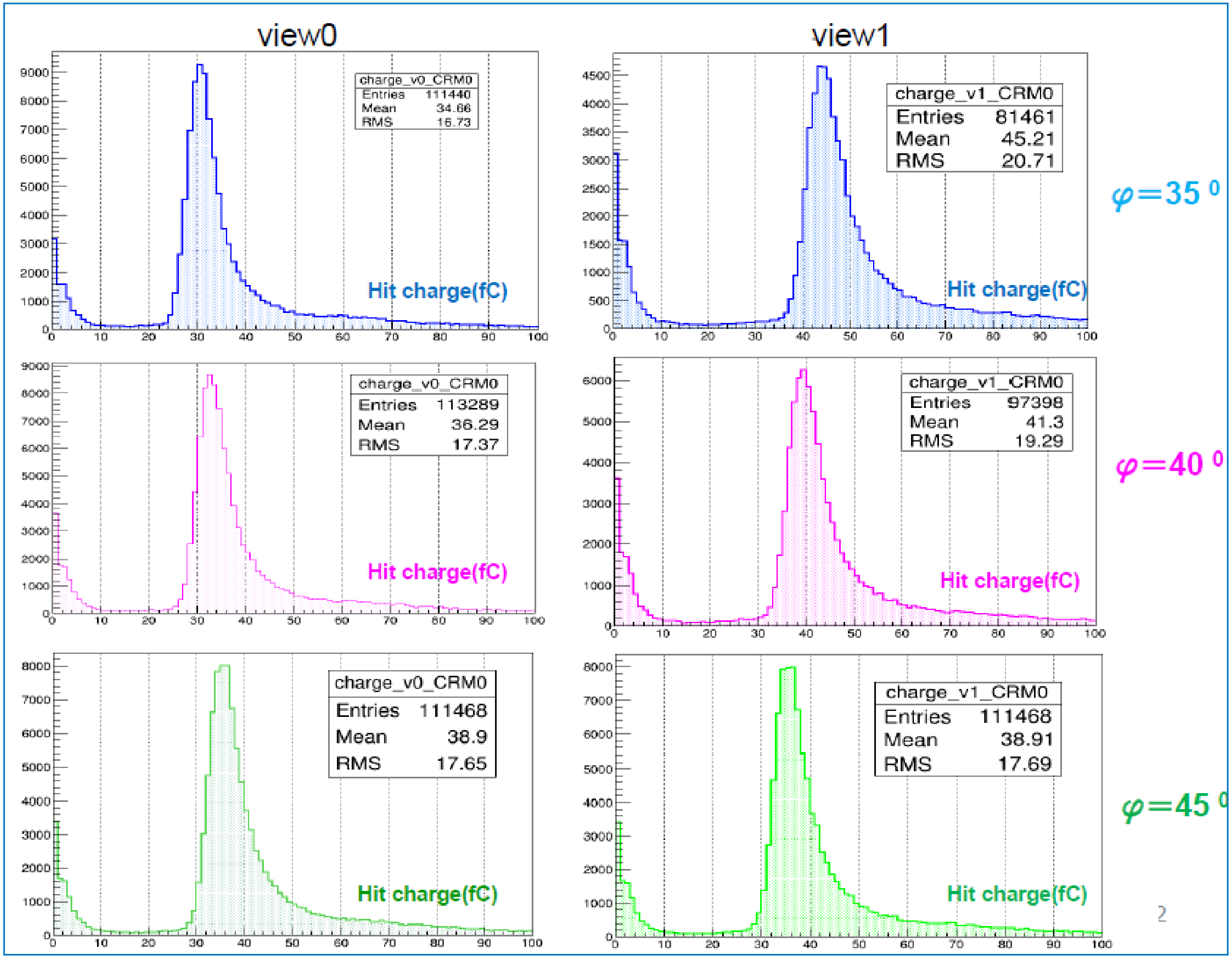
→ Charge depositions in bins of 10 cm corresponding at different drift distances

→moving to longer drift distances, the peak moves to lower values and the distribution becomes narrower as expected due to the charge reduction q=q0*exp(-t/τ)

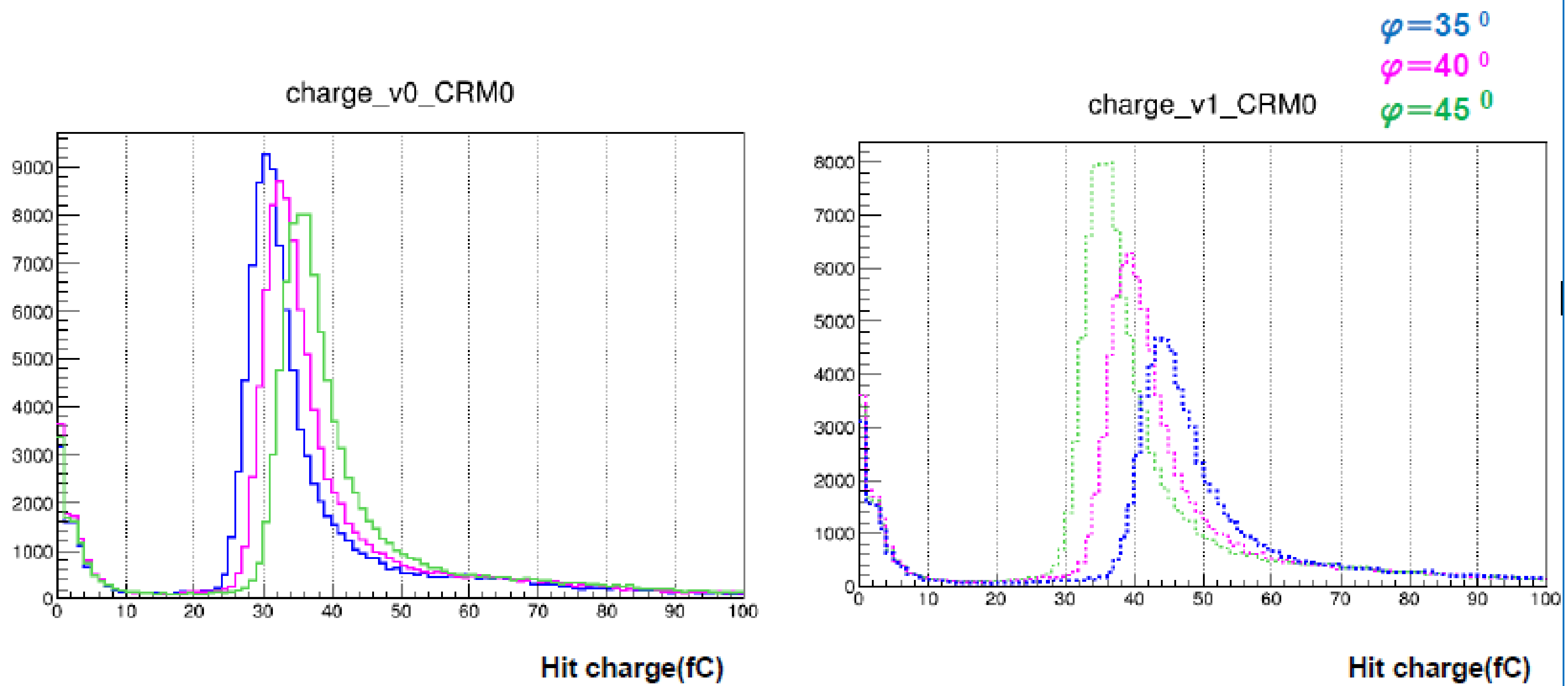
Before moving on in setting up a method to measure purity, it is necessary to come back to some slides shown in the SB meeting hold on December 2nd
https://laguna.ethz.ch/indico/conferenceDisplay.py?confId=177

The subject of these slides was the dependence of the mip position on track angle:




view0
view1
charge_v0_CRM0
Entries 111440
Mean 34.66
RMS 16.73
Hit charge(fC)
charge_v1_CRM0
Entries 81461
Mean 45.21
RMS 20.71
Hit charge(fC)
φ=35 0
charge_v0_CRM0
Entries 113289
Mean 36.29
RMS 17.37
Hit charge(fC)
charge_v1_CRM0
Entries 97398
Mean 41.3
RMS 19.29
Hit charge(fC)
φ=40 0
charge_v0_CRM0
Entries 111468
Mean 38.9
RMS 17.65
Hit charge(fC)
charge_v1_CRM0
Entries 111468
Mean 38.91
RMS 17.69
Hit charge(fC)
φ=45 0

The 2 views are no longer equivalent, and the charge hits varies with angle:

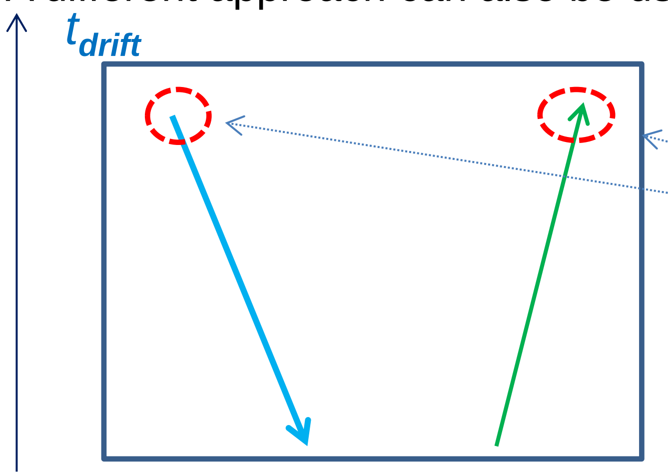


It is necessary to associate to each hit the "effective pitch", depending from angles,
It represents the *width of the LAr slice* seen by the hit

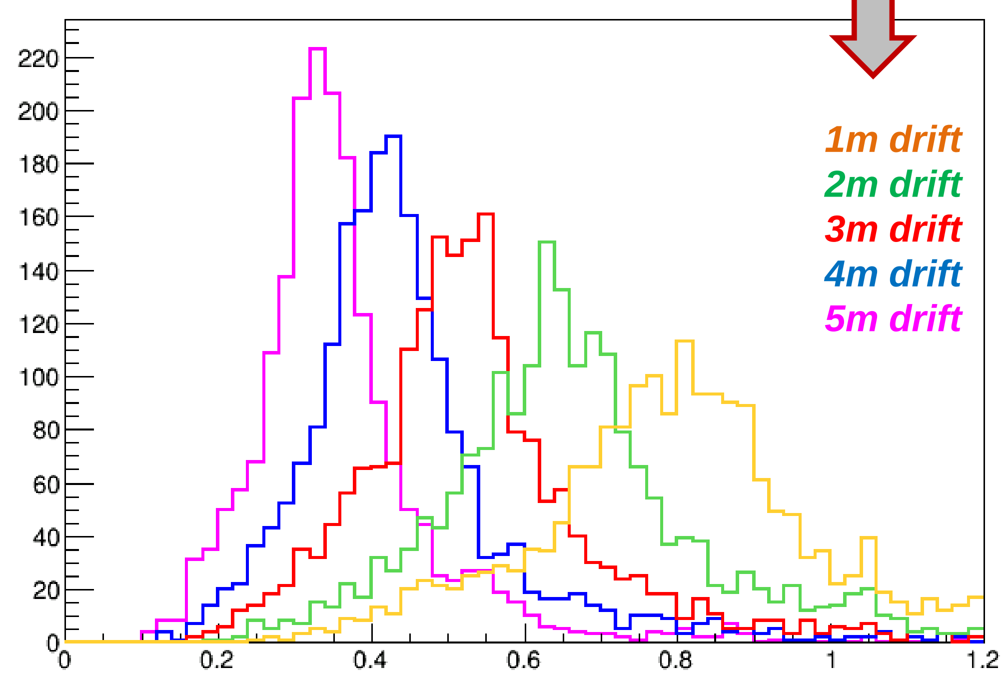
SB meeting, 12/02/1016

To take into account this angular effect the value of the charge deposition has to be "normalized" w.r.t the track angles, using angles provided by reconstruction.

A different approach can also be used → normalize the bins to the one with shortest drift :



- Example of tracks going **upwards** or **downwards** (with respect to drift coordinate)
- The first bin of the vector is defined to be the one corresponding to the minimum drift time
- The charge value of different bin is normalized to the one of the first bin (all are, on average, less than 1)



These histograms are then fitted with a gaussian , to get the peak value:

```
    b_inf=(h->GetMean())-(h->GetRMS());
    b_sup=(h->GetMean())+(h->GetRMS());

1  if (nent>99) {
     h->Fit("gaus","Q","",b_inf,b_sup);    2
     TString fitresult=gMinuit->fCstatu;
     if (fitresult.BeginsWith("CONVE")) itowrite=1;
   }
```

1 to get meaningful results from the fit it is required to have at least 100 entries, otherwise the fit is not done

2 the fit is performed in an interval defined starting from histogram mean value and rms

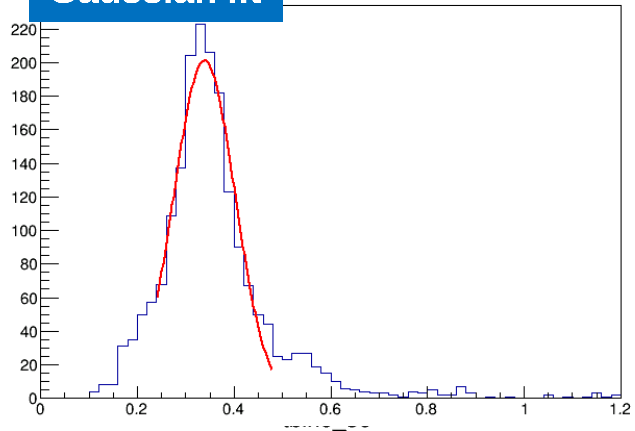
Results are written to an external file, and then a fit to measure lifetime is performed→

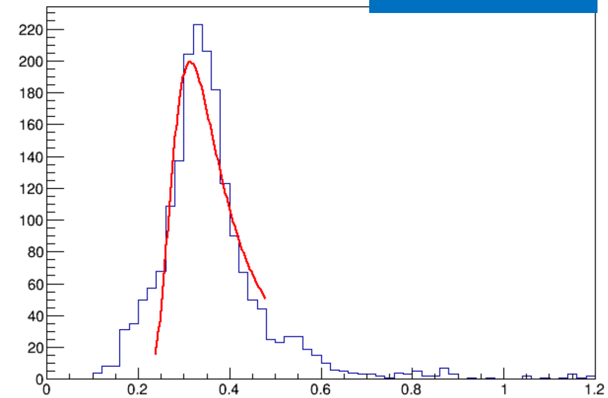
Fit examples for different drift lengths

Gaussian fit

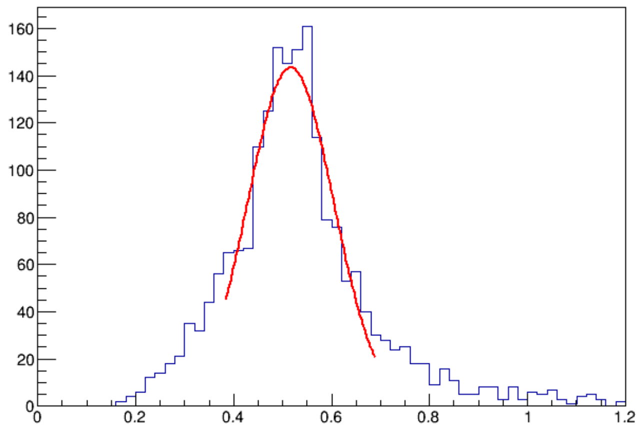
Landau fit

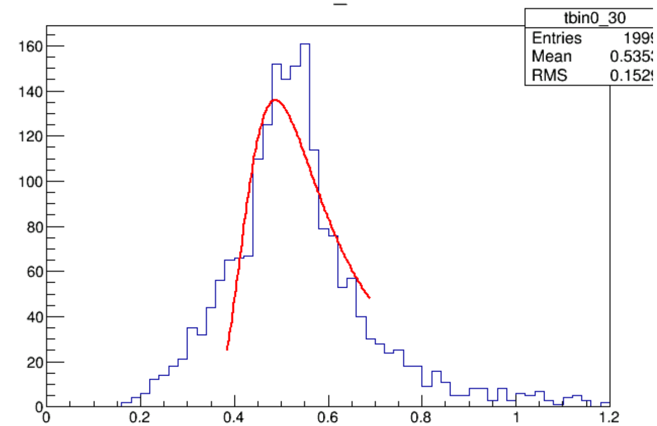
5m drift



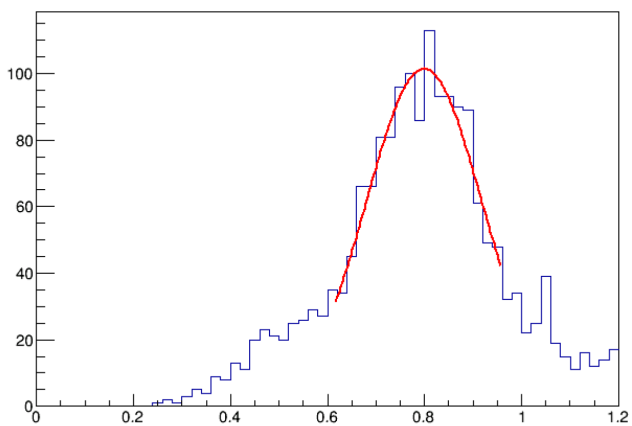


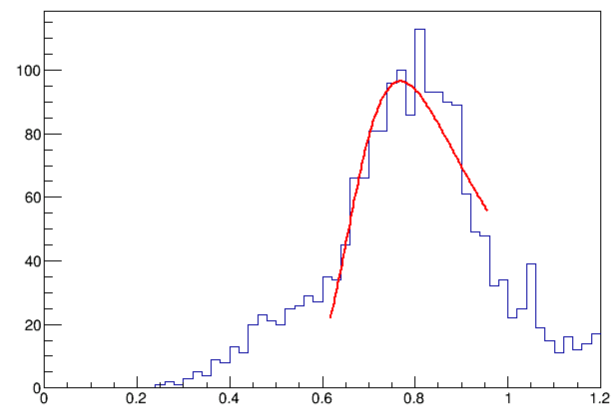
3m drift



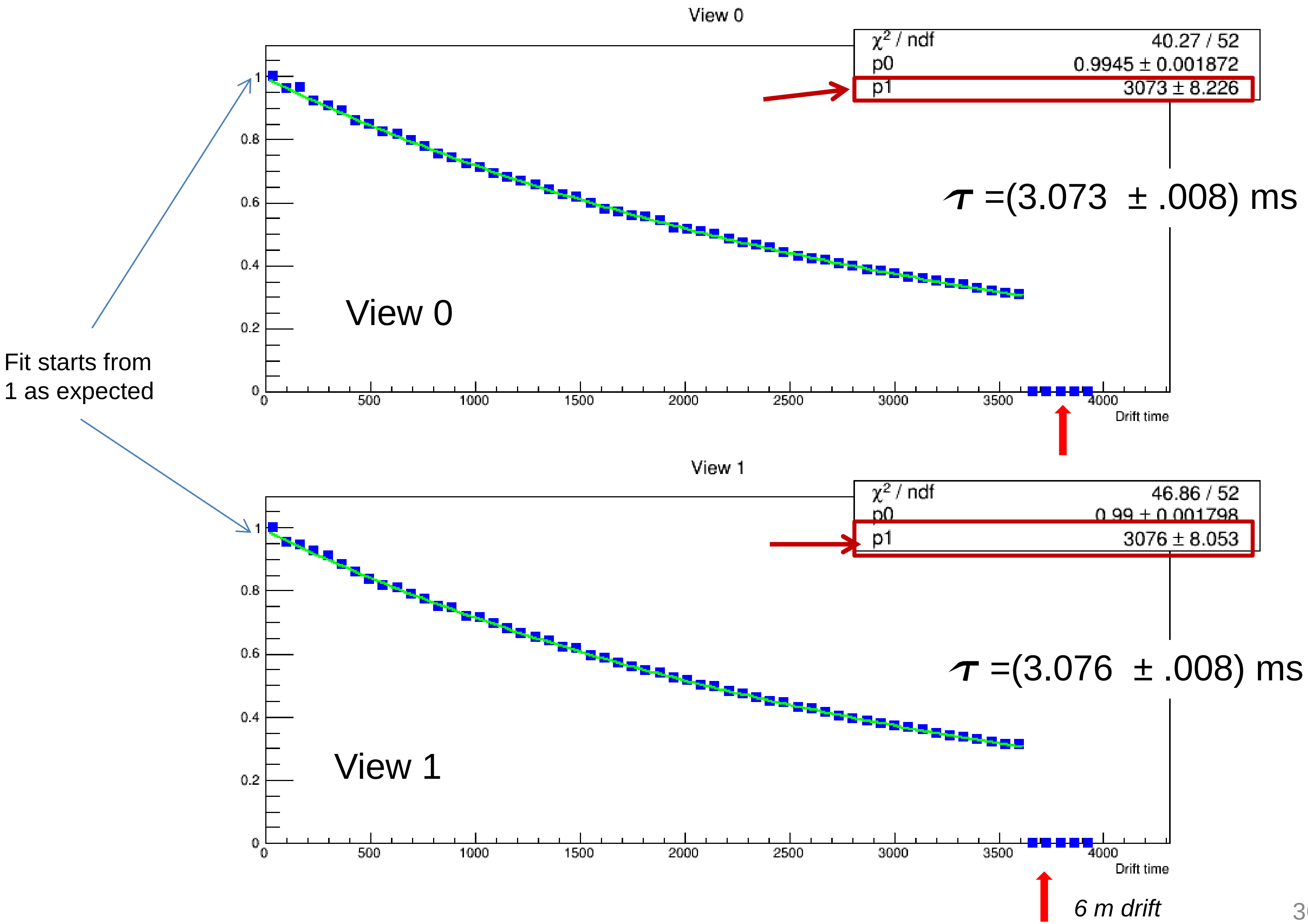


1m drift



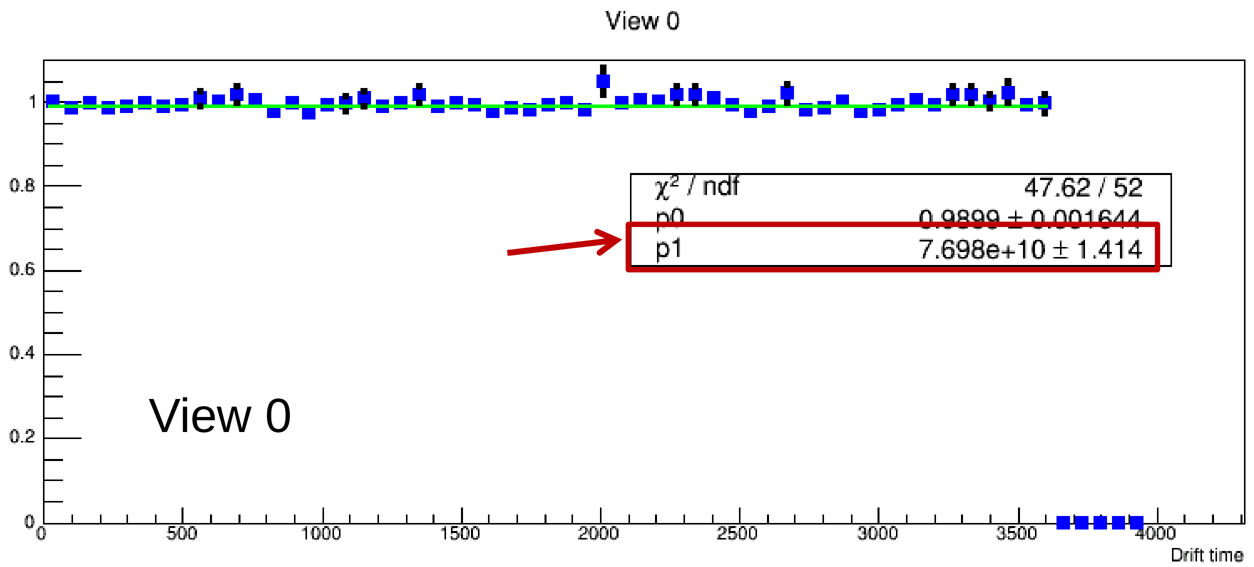


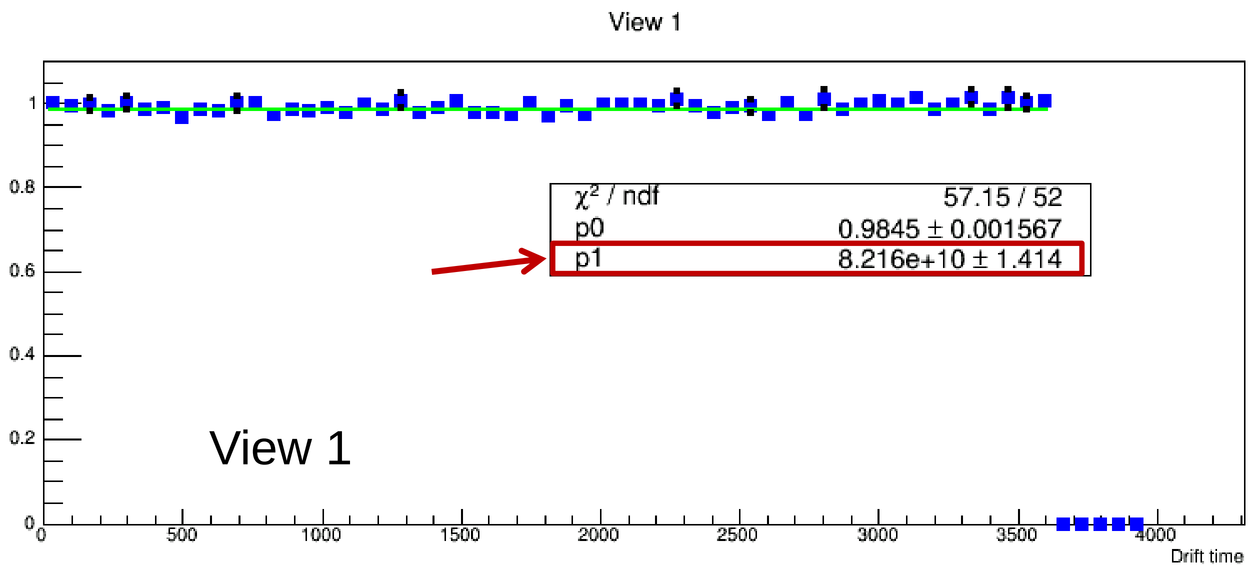
Fit results:



View 0
χ² / ndf 40.27 / 52
p0 0.9945 ± 0.001872
p1 3073 ± 8.226
τ =(3.073 ± .008) ms
View 0
Fit starts from
1 as expected
Drift time
View 1
χ² / ndf 46.86 / 52
p0 0.99 ± 0.001798
p1 3076 ± 8.053
τ =(3.076 ± .008) ms
View 1
Drift time
6 m drift

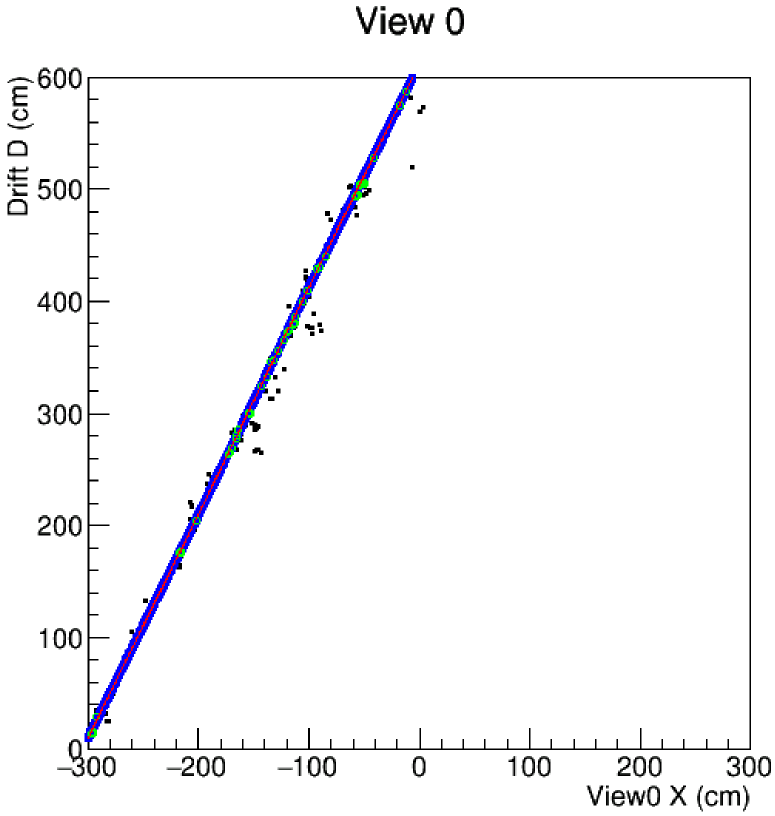
The method has been tested on a sample of muon generated without lifetime effect…

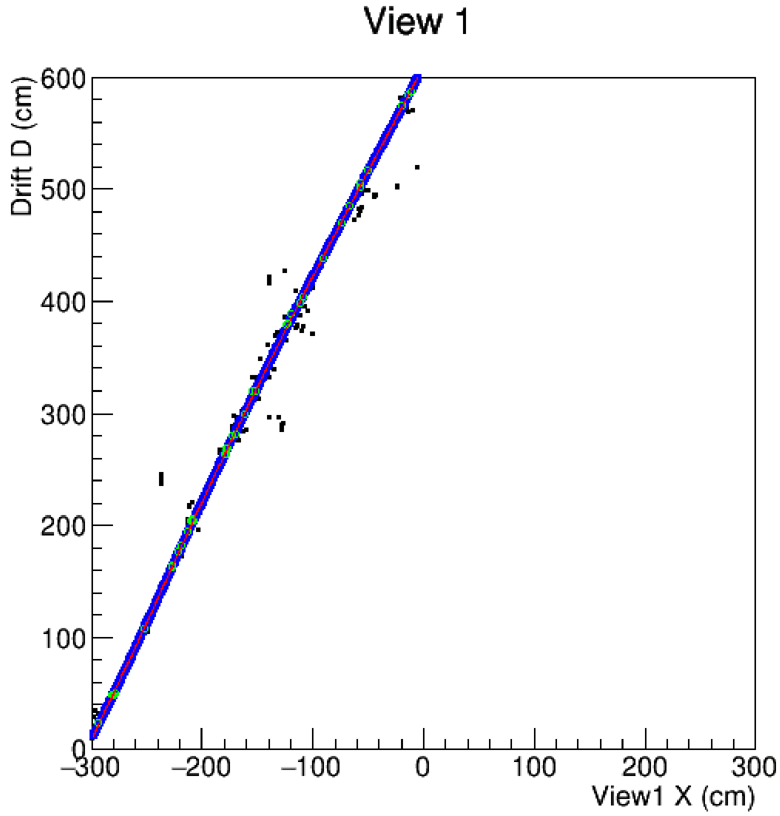


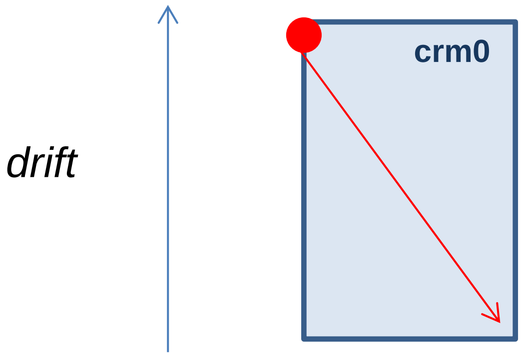


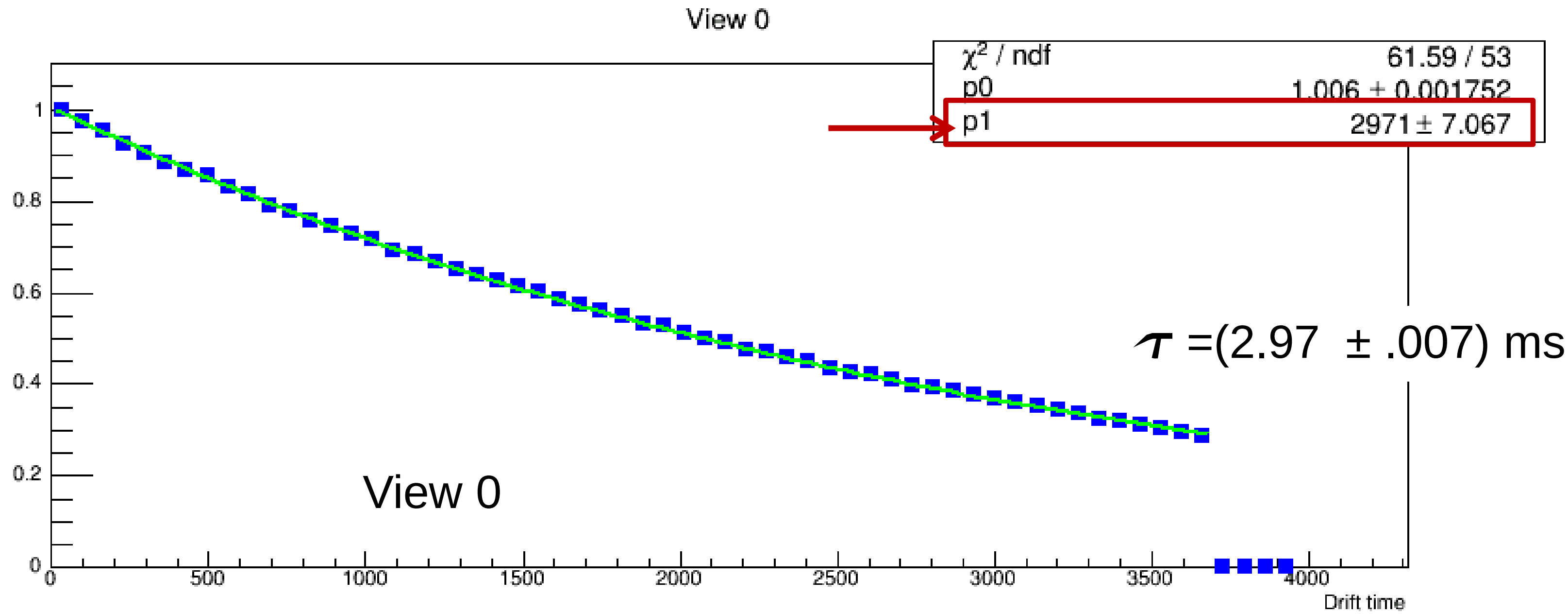
The distributions are flat, as expected

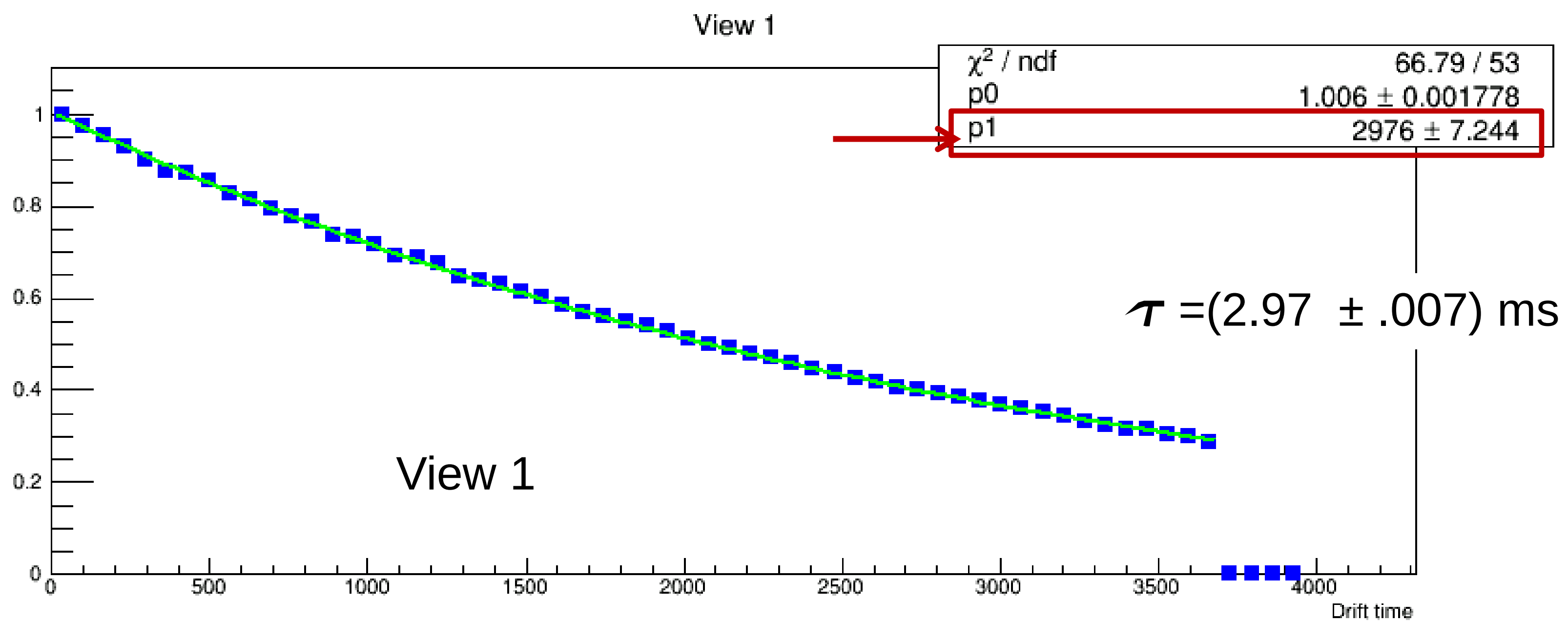
..and on a sample of muons crossing the detector from top to bottom:



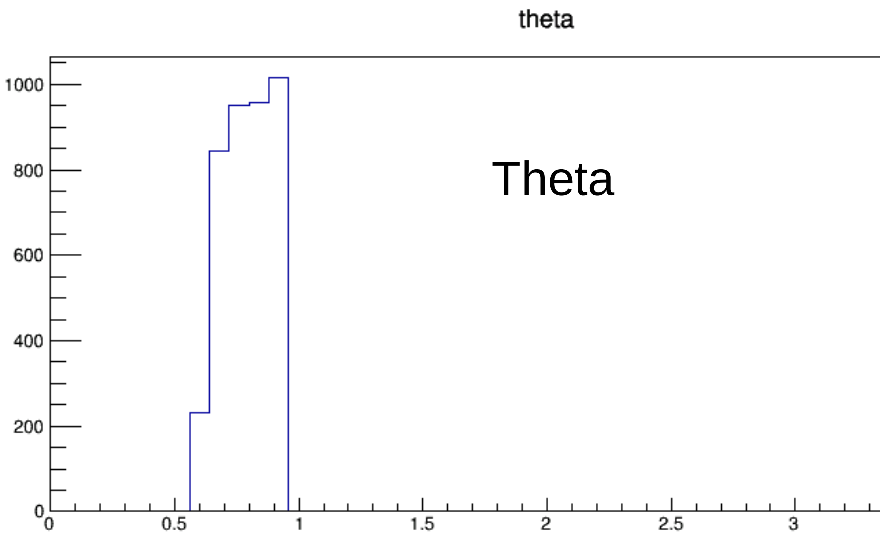


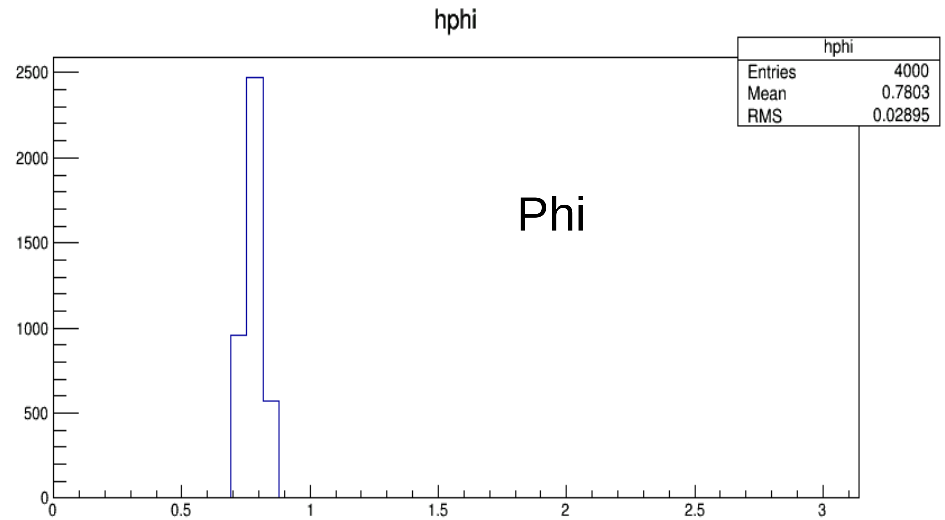



View 0
χ² / ndf 61.59 / 53
p0 1.006 ± 0.001752
p1 2971 ± 7.067
τ =(2.97 ± .007) ms
View 0
0
0.2
0.4
0.6
0.8
1
0
500
1000
1500
2000
2500
3000
3500
4000
Drift time

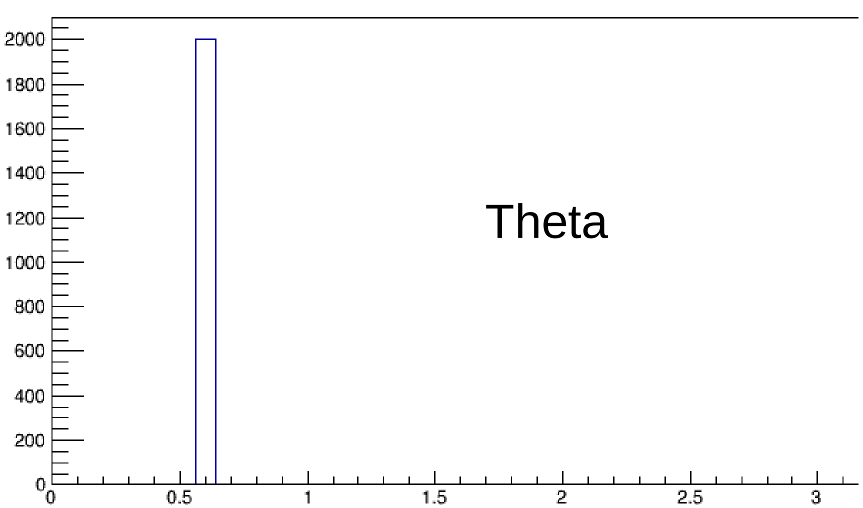

View 1
χ² / ndf 66.79 / 53
p0 1.006 ± 0.001778
p1 2976 ± 7.244
τ =(2.97 ± .007) ms
View 1
0
0.2
0.4
0.6
0.8
1
0
500
1000
1500
2000
2500
3000
3500
4000
Drift time

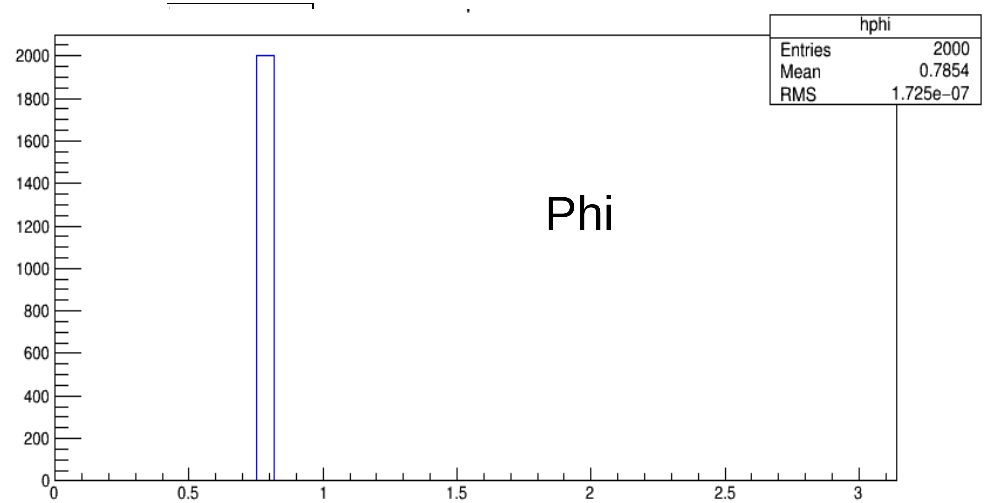
The method has been tested on a sample of tracks at different angles:



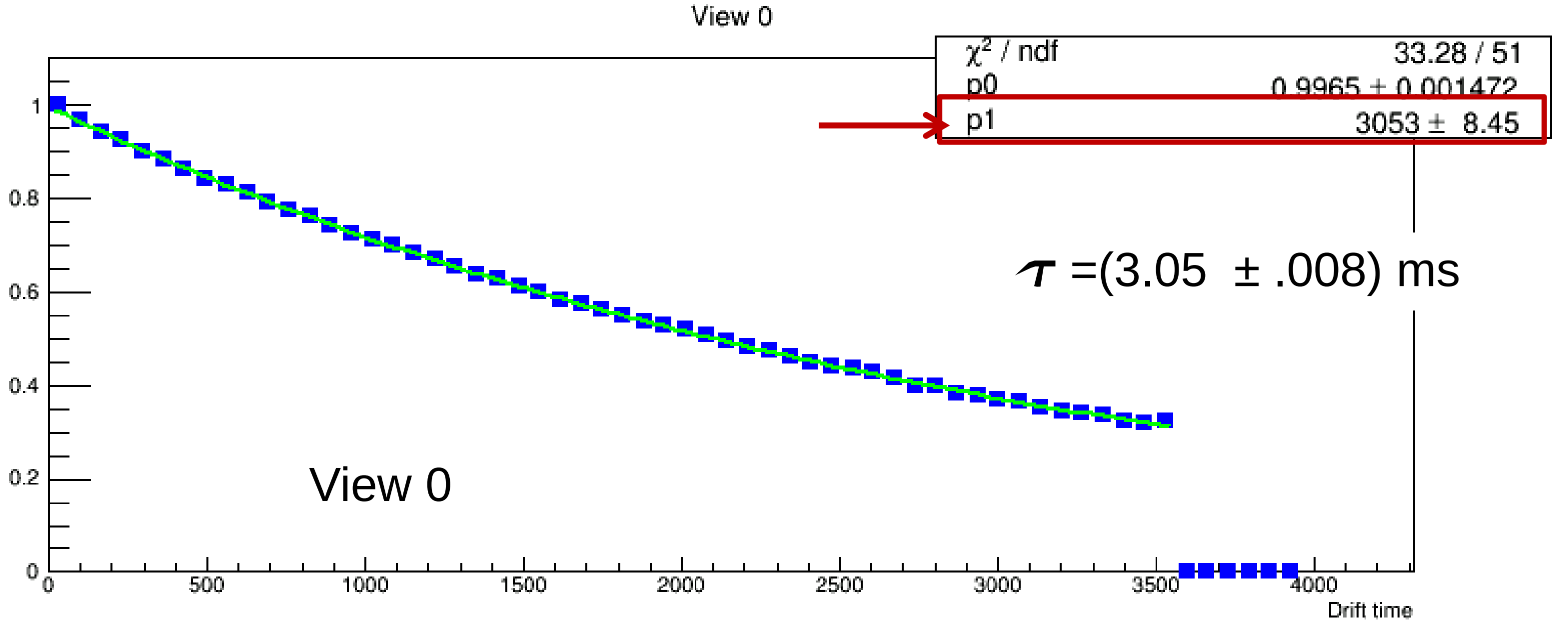


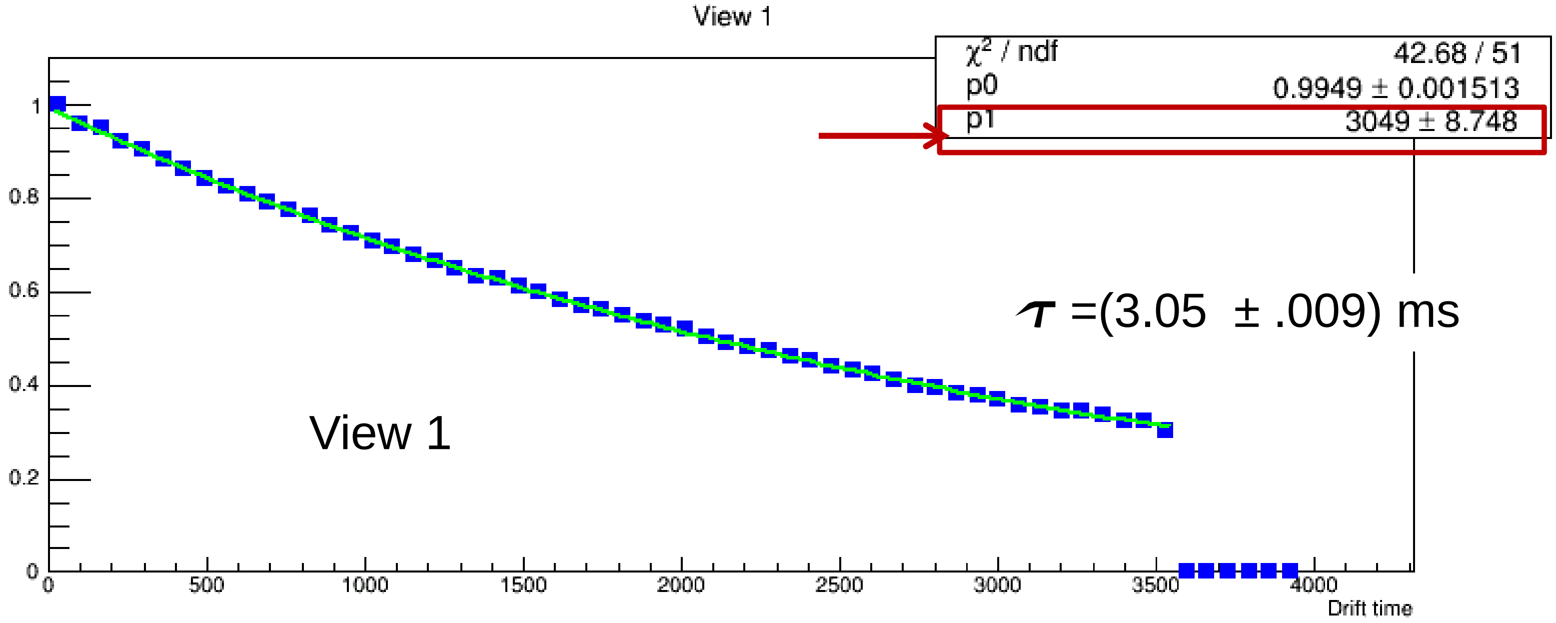
For comparison: values for previous sample:



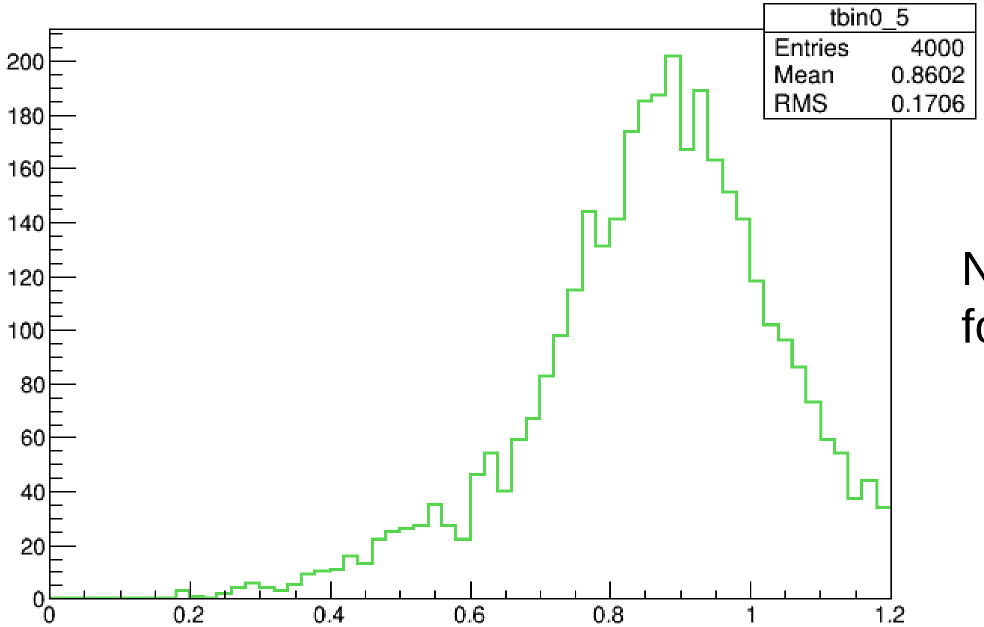


Fit results on the sample with random angular directions:

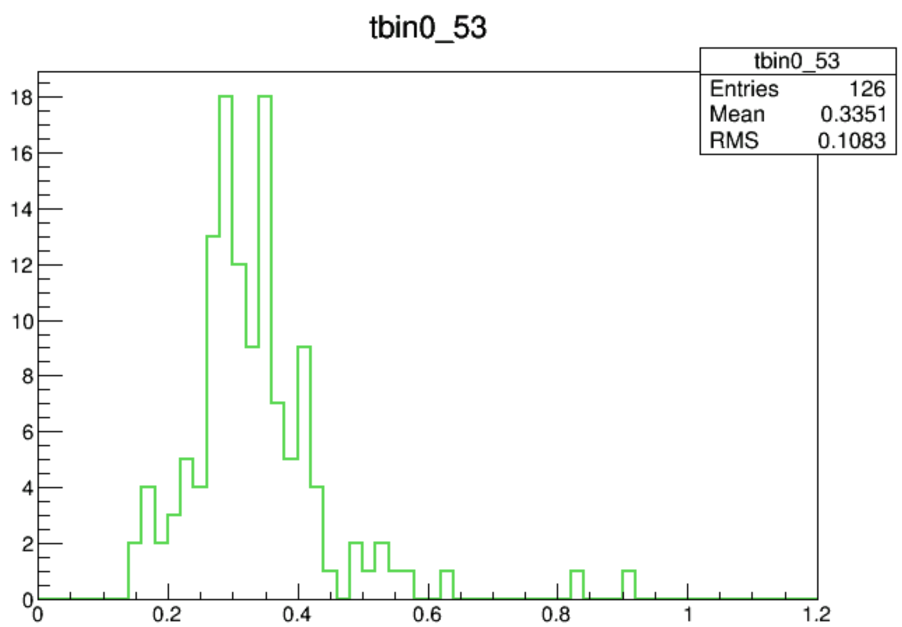




The sample with random directions used for the fit corresponds to 4K tracks:

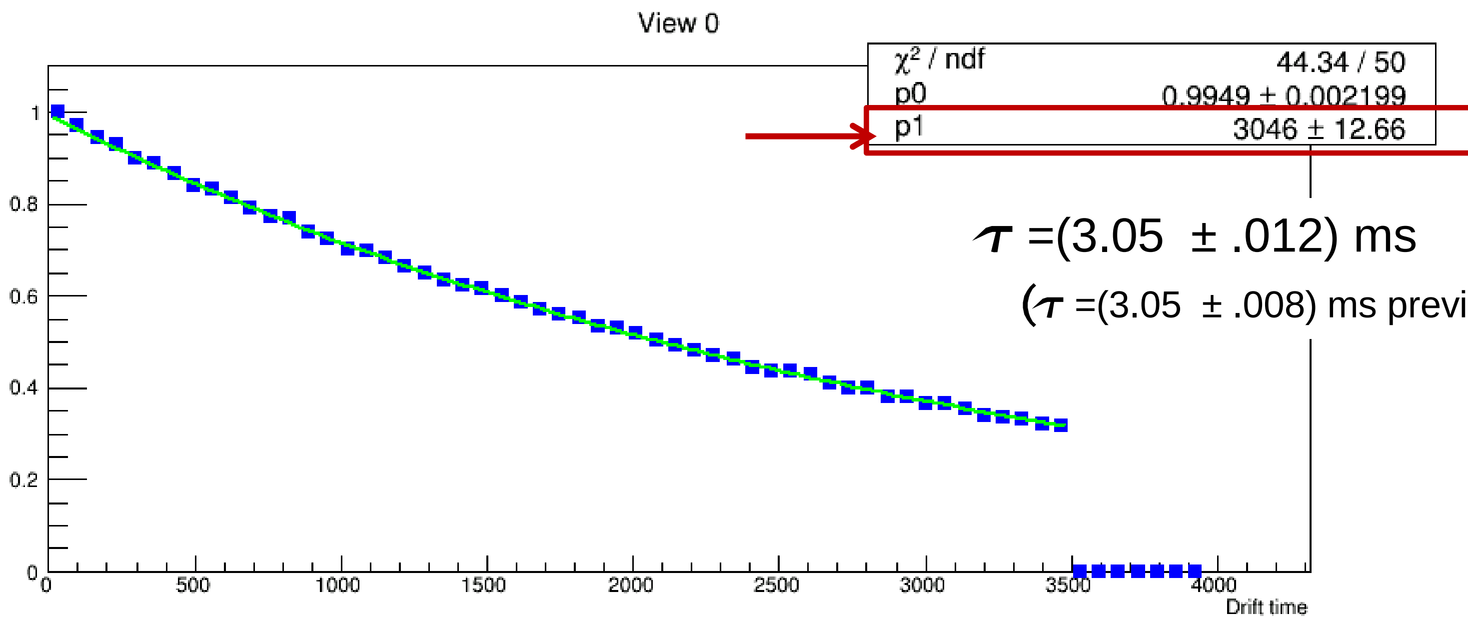


Normalized charge deposition for a drift distance of 50 cm



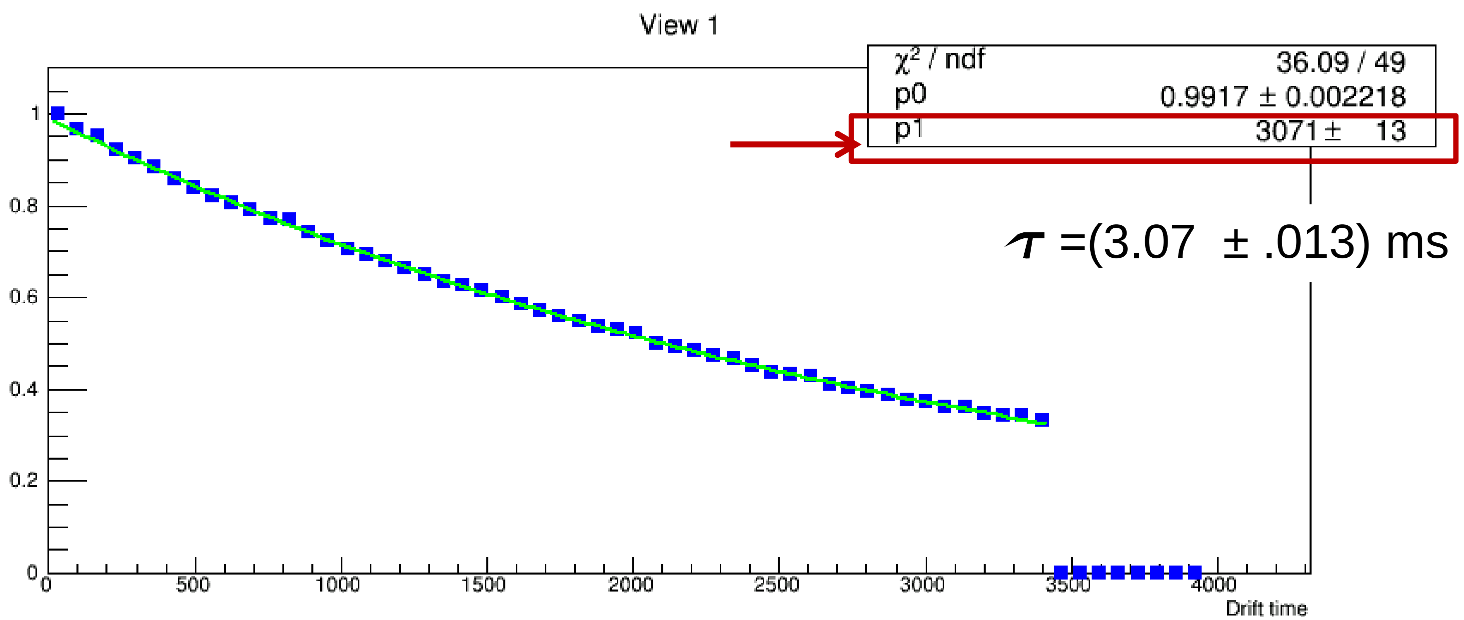
The last bin used in the fit shown at page 43 corresponds to a drift distance of ~5.3 m with a population of 126 tracks

Fit results with lower statistics (a subsample of 2K tracks):

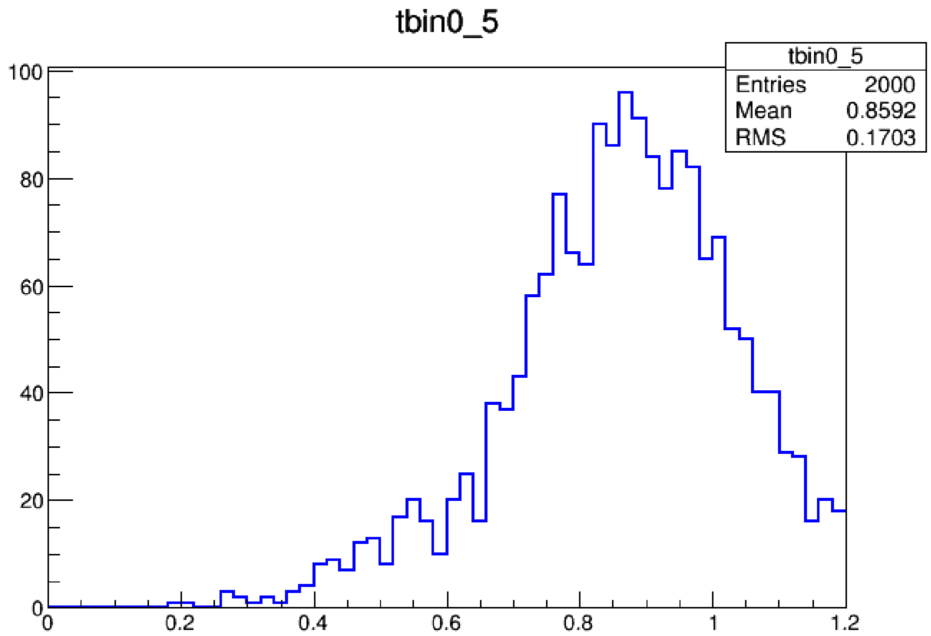


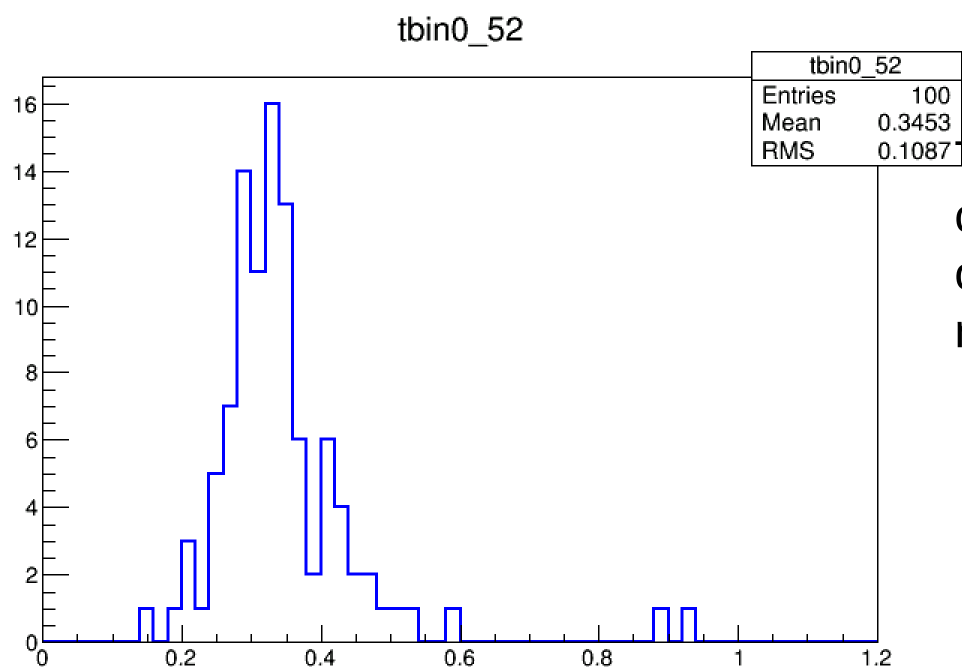
τ =(3.05 ± .012) ms

(τ =(3.05 ± .008) ms previous value)



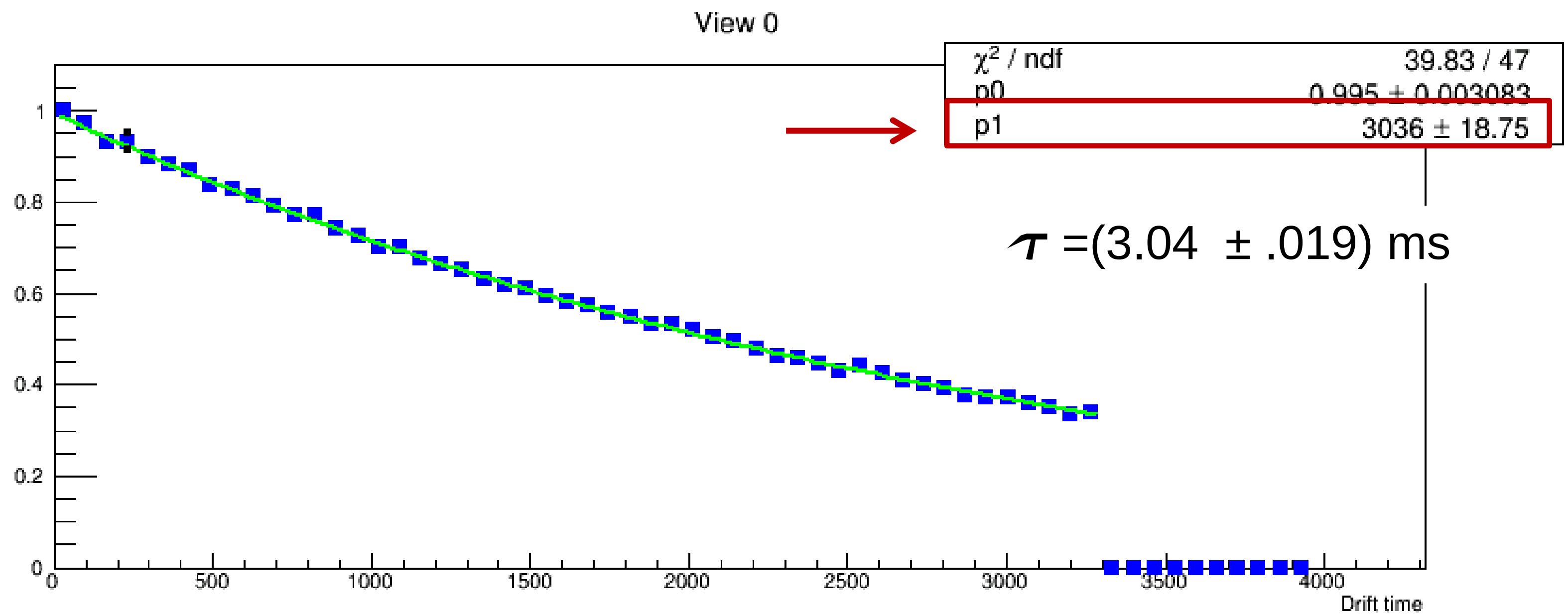
τ =(3.07 ± .013) ms



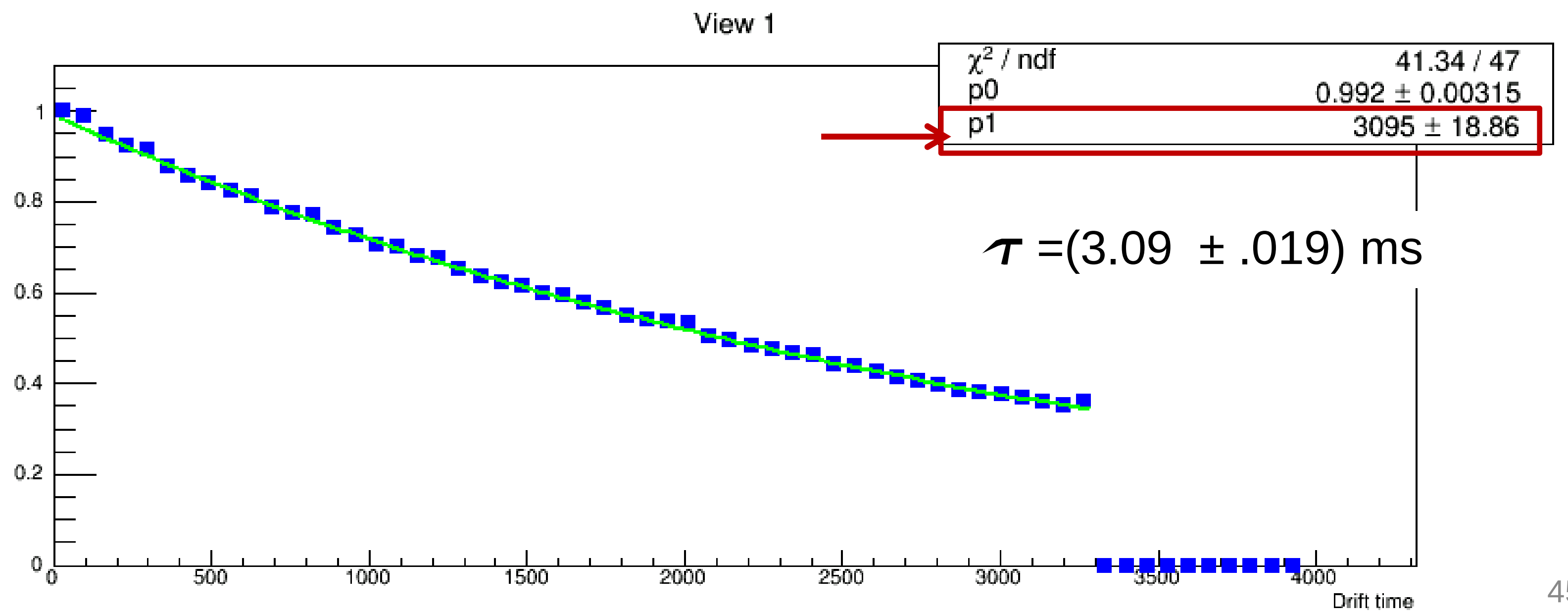


The last bin used in the fit corresponds to a drift distance of ~5.2 m with 100 tracks, the next bins do not have statistics

Fit results with lower statistics (a subsample of 1K tracks):

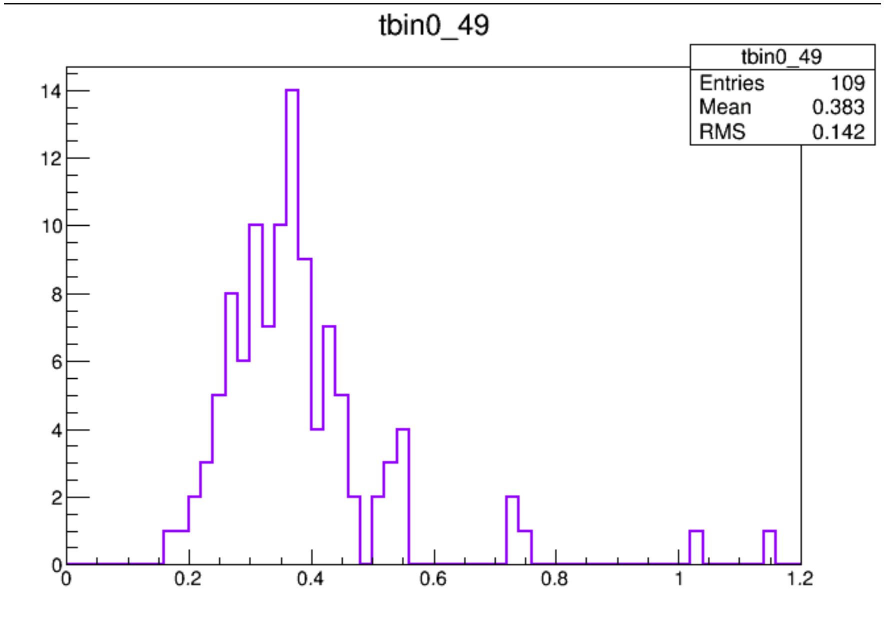


τ =(3.04 ± .019) ms



τ =(3.09 ± .019) ms

Last useful bin: 100 track, drift distance ~5m



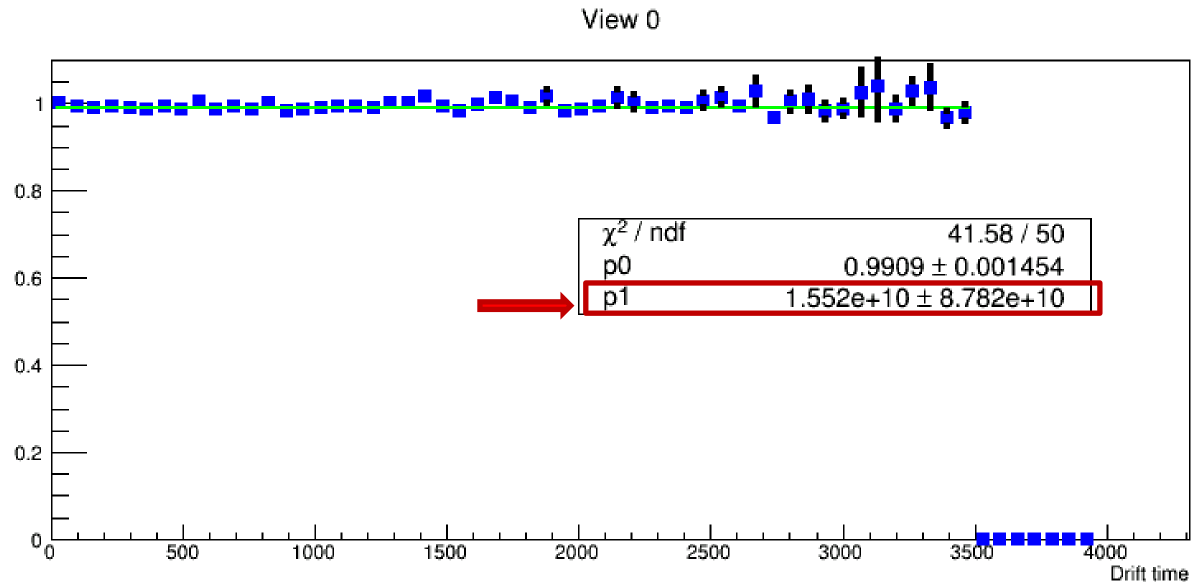
Summary:

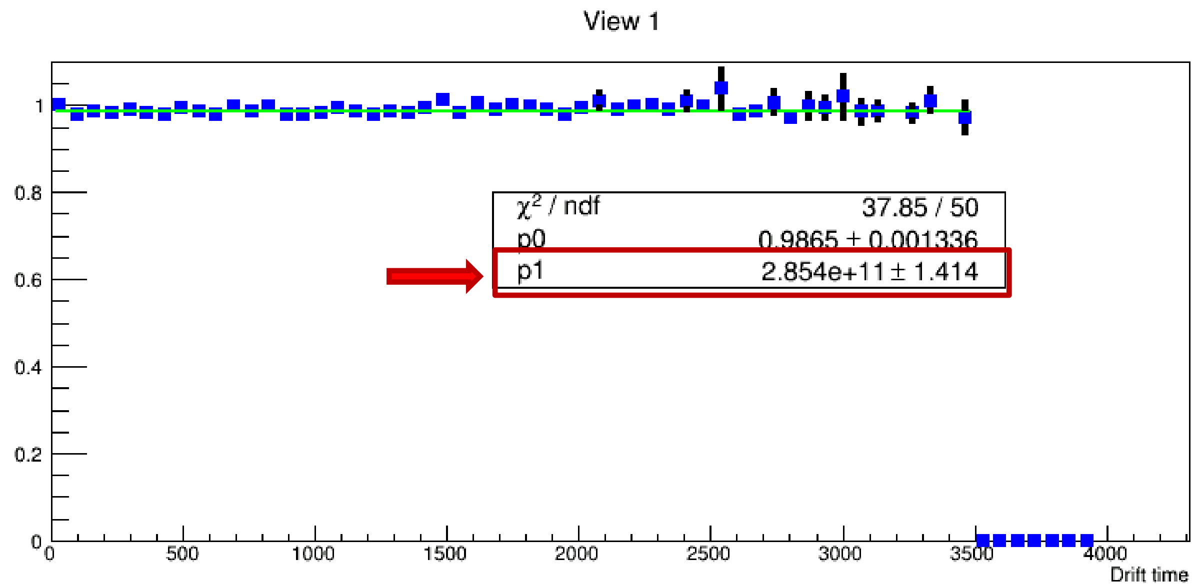
Real lifetime τ =3.00 ms

4k tracks τ =(3.05 ± .008) ms

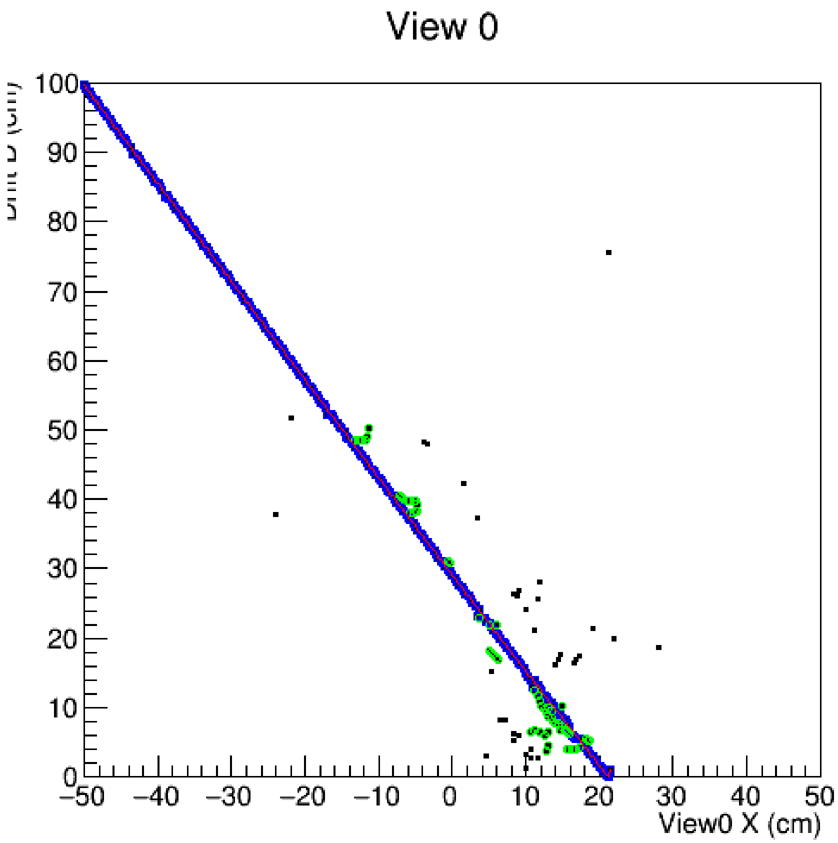
2k tracks (subsample) τ =(3.05 ± .012) ms

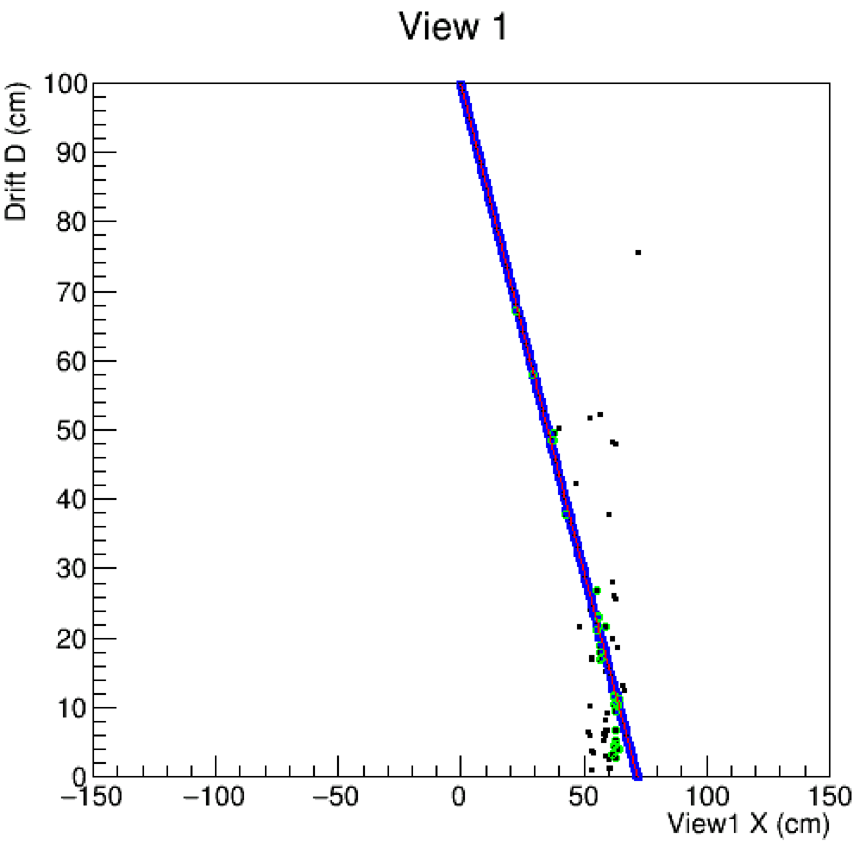
1k tracks (subsample) τ =(3.04 ± .019) ms just for check→ same sample of muon generated without lifetime effect:

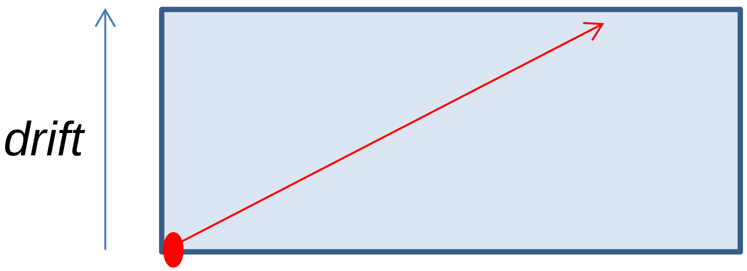




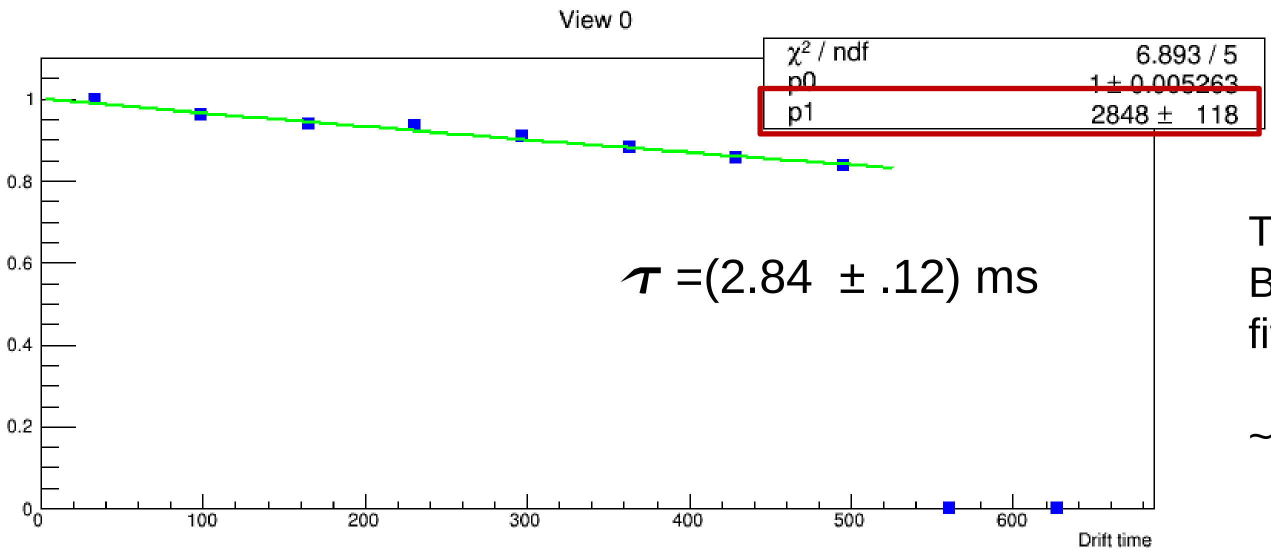
The same method has now been applied to muon generated assuming the 3x1x1 configuration:





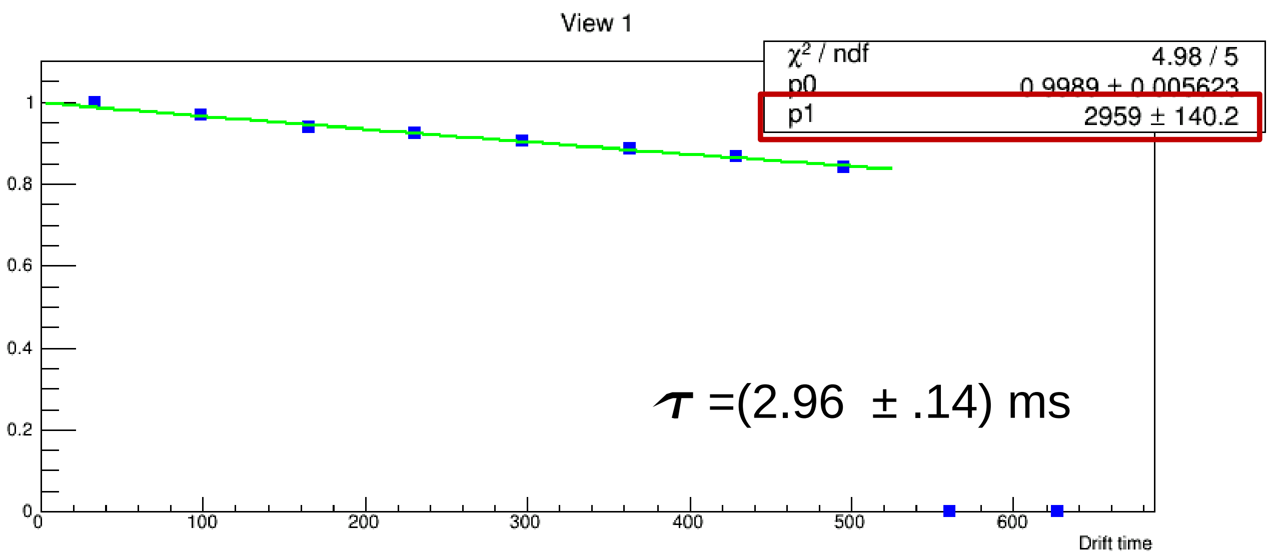


Fit results:

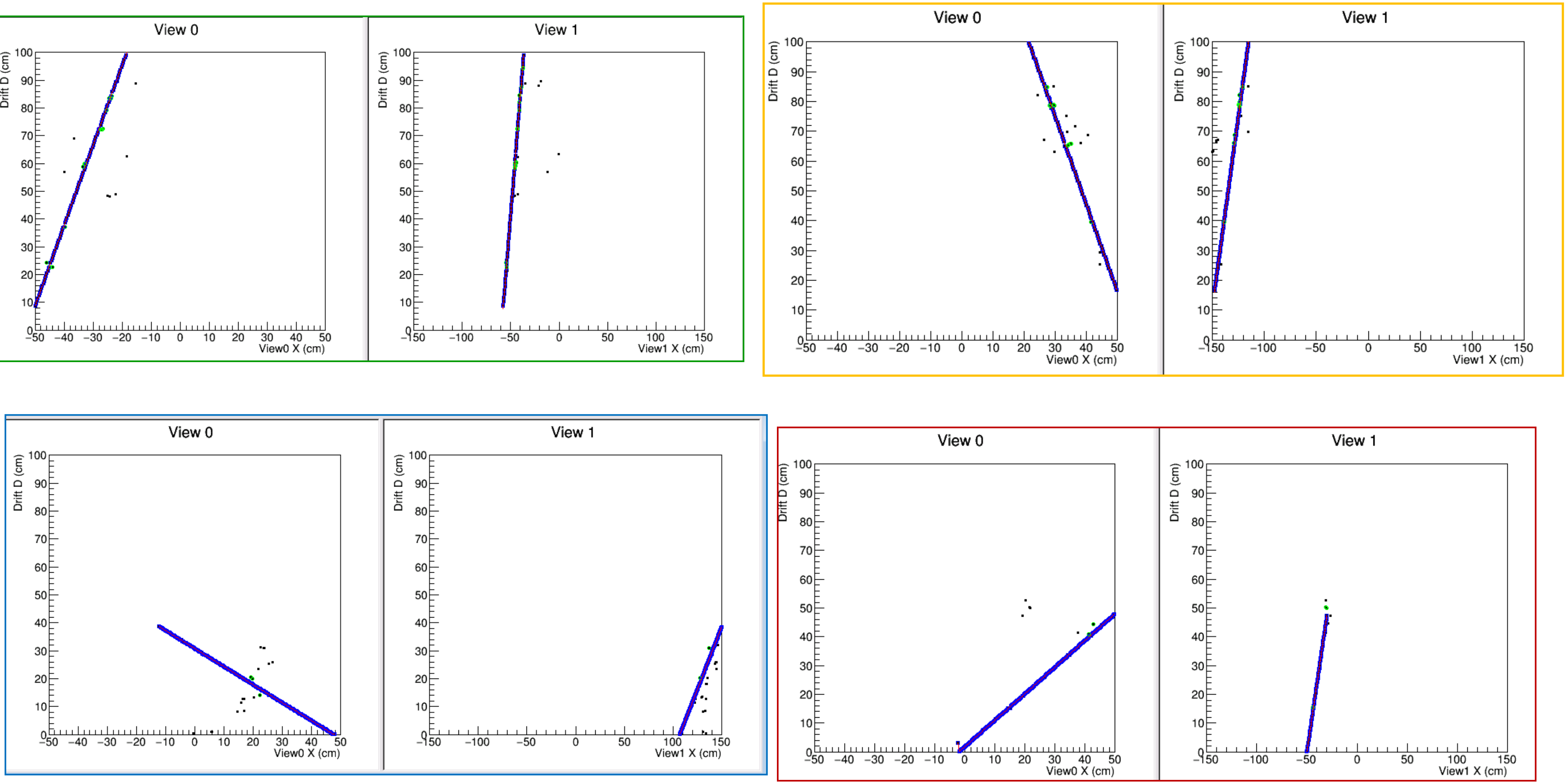


Tracks are shorter→
Bigger error and worst
fit result w.r.t. 6x6x6

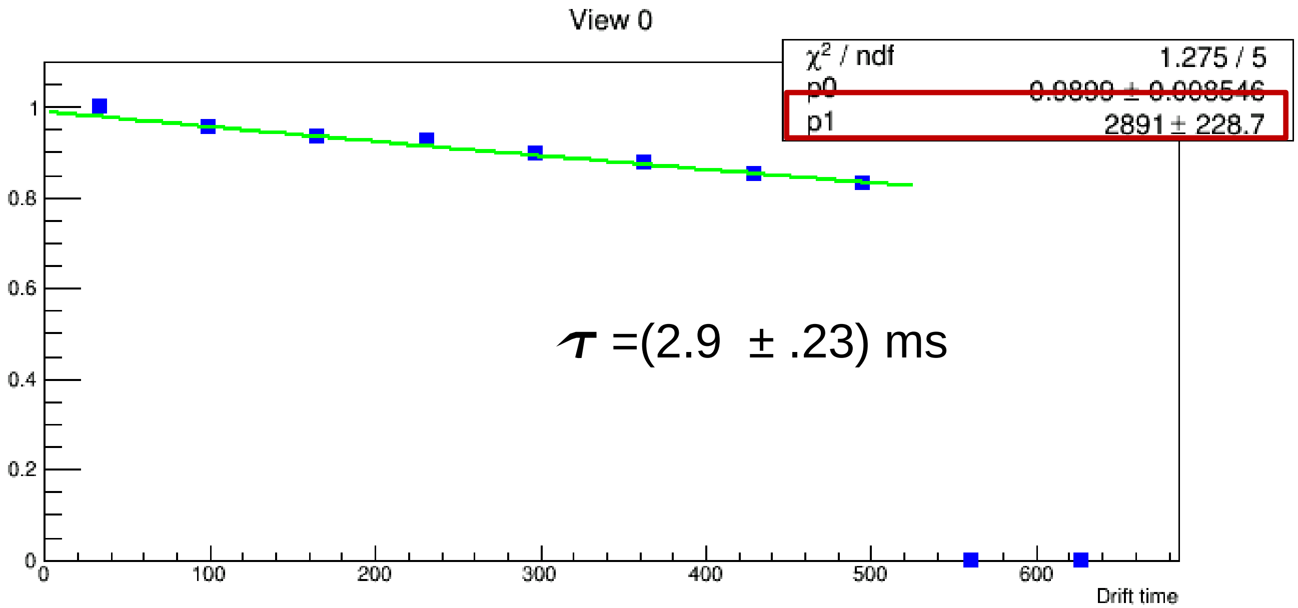
~4K tracks, 70 cm drift length

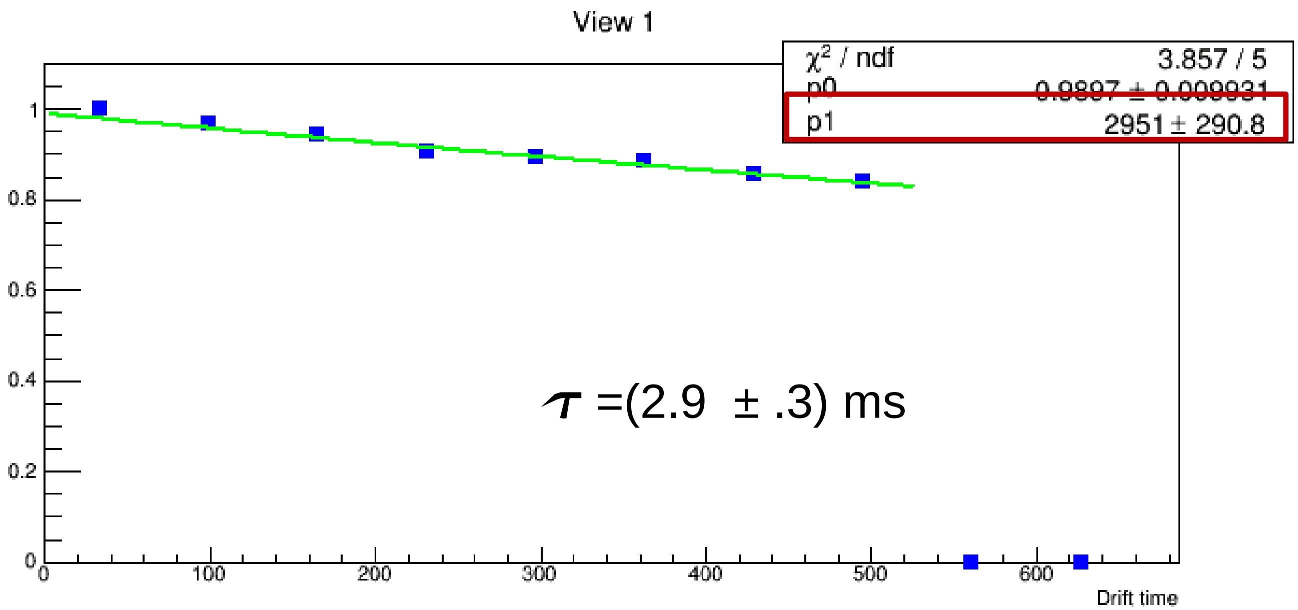


To test the method in a realistic situation, a sample of cosmic has been generated.
Some examples:



tracks with different slopes and different starting point results of purity measurement:





~1500 tracks

Conclusions:

- A method to measure purity from muon tracks has been developed.
  It is independent on track direction, on track starting point, and it is based on 2D track reconstruction. Two measurements of purity are obtained, one for view 0 and one for view 1 *(this a cross check, since the 2 values have to be equal)*

- It is based on a script which requires as input the root file from reconstruction. For each analyzed file (filename) the script produces a directory (purity_filename) containing all results.

- This script which runs an executable, generates and runs a root macro to analyze the output of the executable in order to determine the purity values.

- The code still needs some cleaning up, it is not yet committed.

- To be done:

- Since for the 3x1x1 the trigger counters will select nearly horizontal tracks, it is probably needed an additional method to evaluate the purity using horizontal tracks at different drifts