ν_τ analysis

Introduction

MC data for IC86 made by Claudio. Total size of the set is 100,000 ν_τ events with energy range from 1TeV up to 10PeV. The spectrum is injected as E^{-1} at earth surface.

Double cascade sample: ν_τ charge current event

τ must decay to e or hadrons

Both cascades should be inside IceCube

Single cascade sample: ν_τ neutral current events

ν_τ charge current events with small track length

Use data to check offline fitter (log likelihood)

- Test double seed (two cascades) selection
- Test the fitter itself compare reconstructed results with MC truth

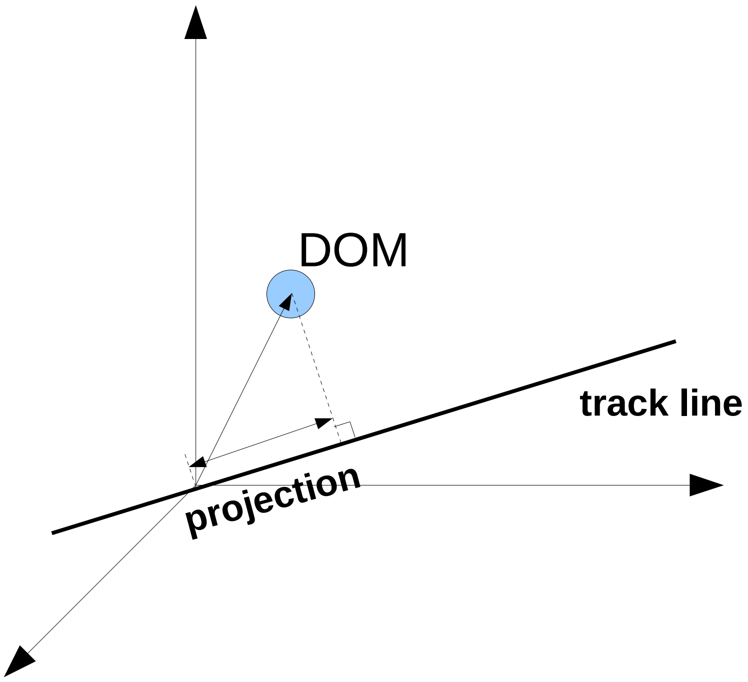
Seed selection

ν_τ event have two cascades, so we need 2 I3Particles as a seed for the fitter. The seeds are made by TauDPSeedMaker module. It has several modes, depending on the input information (clast cascade, MC truth seed,...).

Seed selection:

- MC seed – uses MC truth positions of the cascades; for NC events TauDPSeedMaker creates two overlapping cascades, one of them has energy 0
- clast seed – uses clast result to find position and direction of the first cascade; TauDPSeedMaker estimates how energy is split between τ and ν interaction cascade; then τ track length is estimated and the second cascade position is found using track length and the first cascade direction
- split seed – split pulse map in two, finding pulses from the first and the second cascades; use clast for each map separately to find two cascades (energies and positions); the cascade directions are set as direction of the line connecting the first cascade with the second

Map splitting



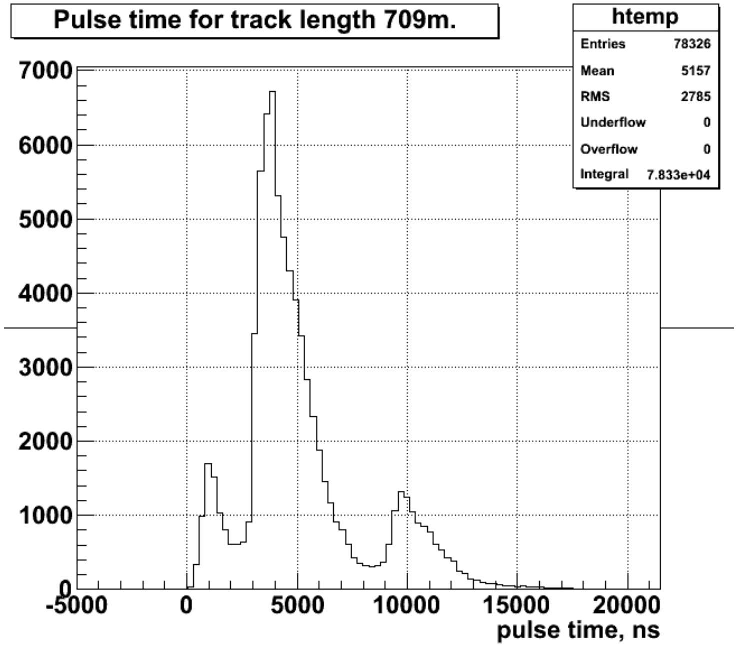
Pulse time splitting:

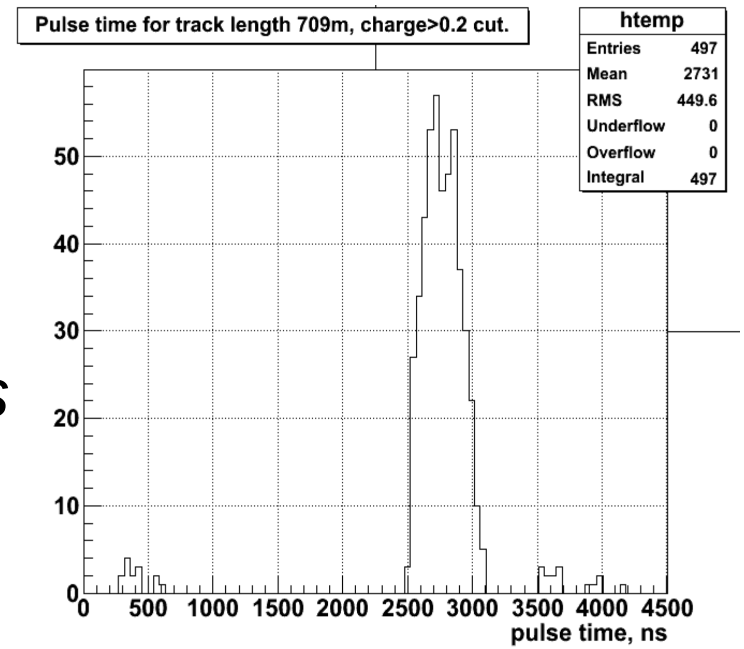
- *find start and end time of the event*
- *calculate split time of the event (more on slide 8)*
- *all pulses before the split time belongs to the first cascade, all pulses after it belong to the second one*
- *all pulse times are shifted in such a way that the start time is 0*

Pulse position projection on the tau track line:

- *find track line using linefit algorithm*
- *create projection distribution*
- *find the first and the last pulse projection*
- *calculate the middle of the projection segment*
- *all pulses that are projected to the first half of the segment belong to the first cascade, all pulses that are projected on the second half belong to the second cascade*
- *all distribution plots will be shifted that middle of the projection segment is at 0*

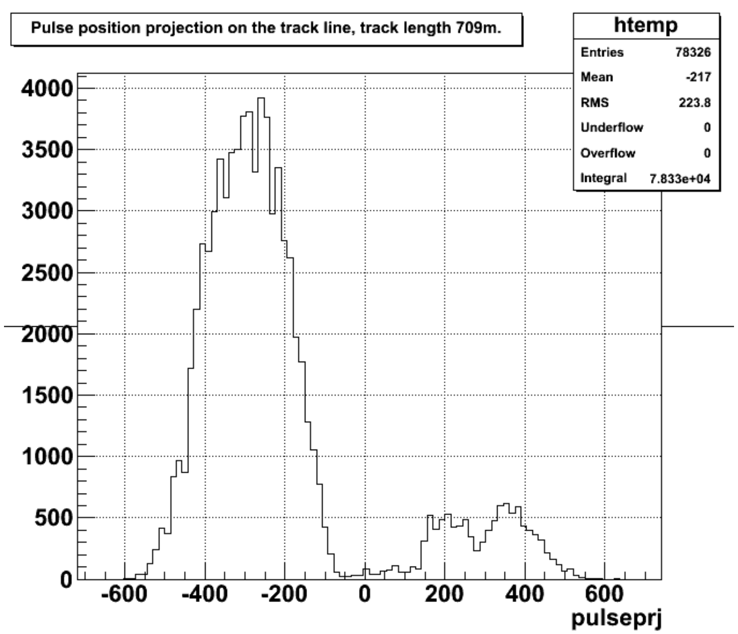
Map splitting example ($l_{\tau\ track}$=709m)



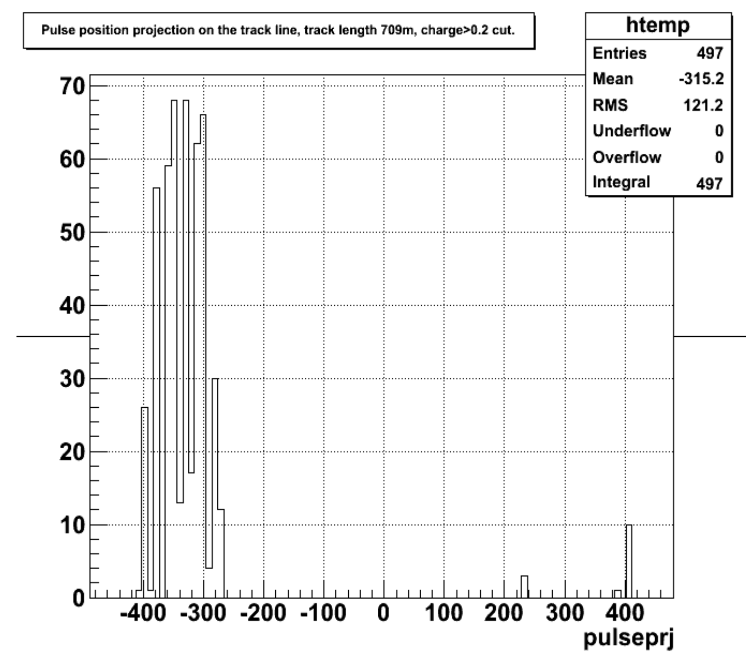


time splitting

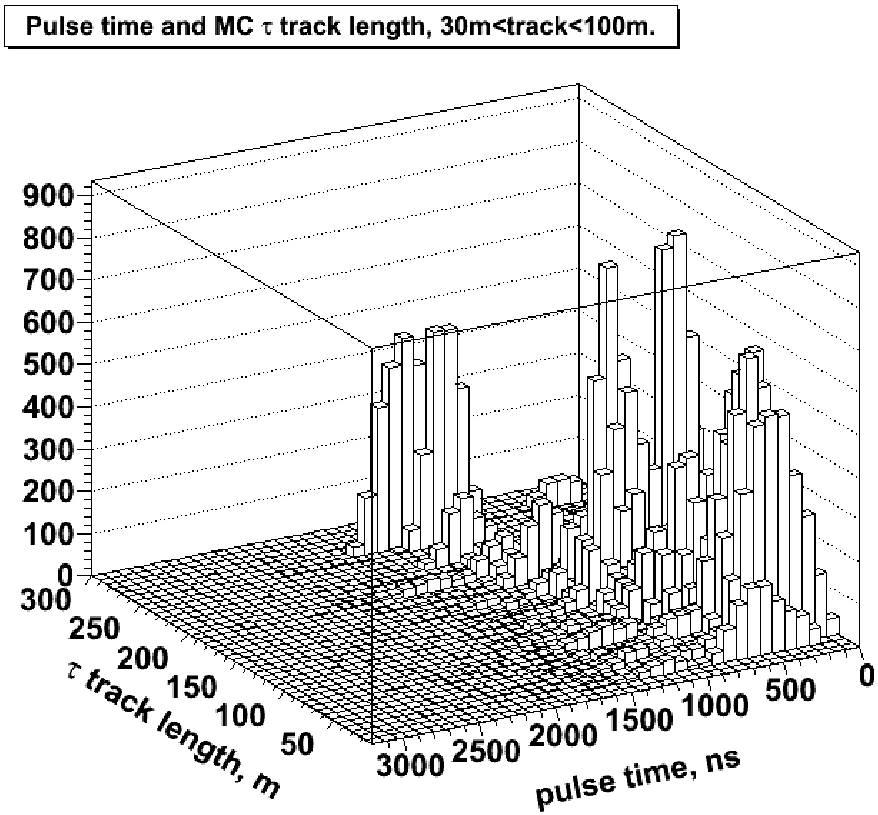
$dt_{estim.}$=2000ns

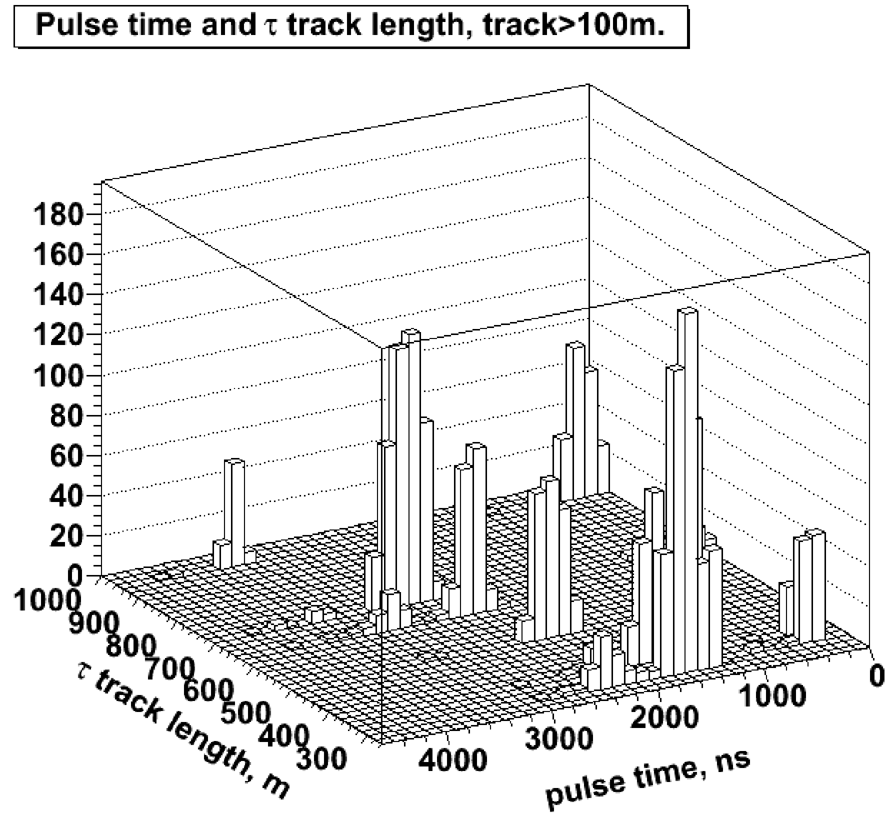


track projection splitting



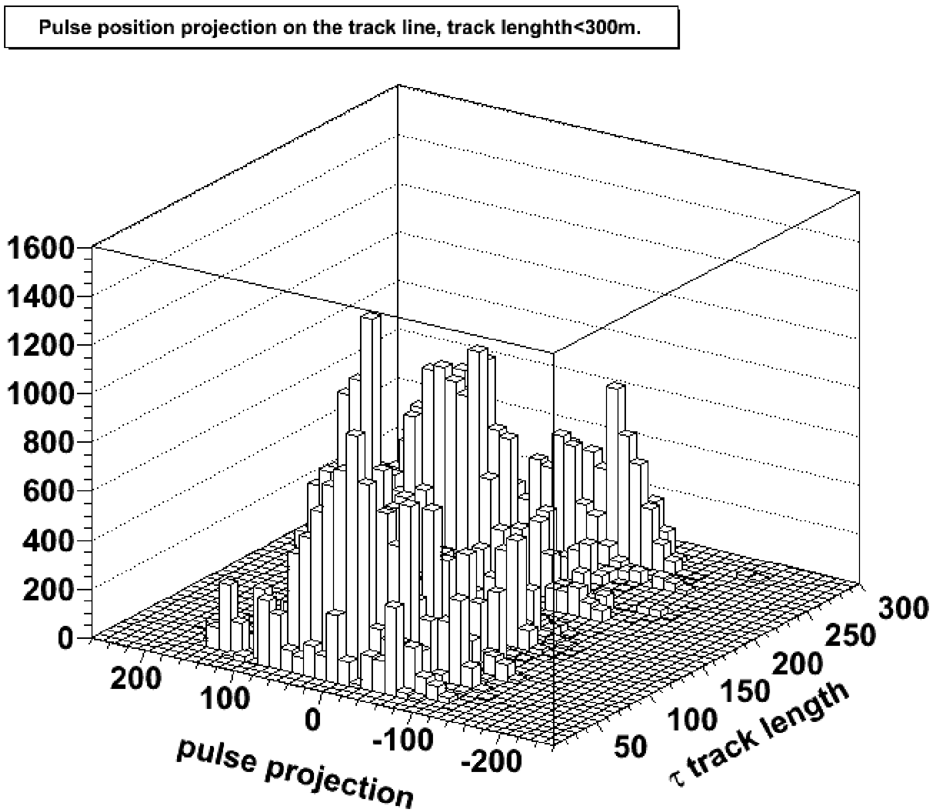
Pulse time versus track length

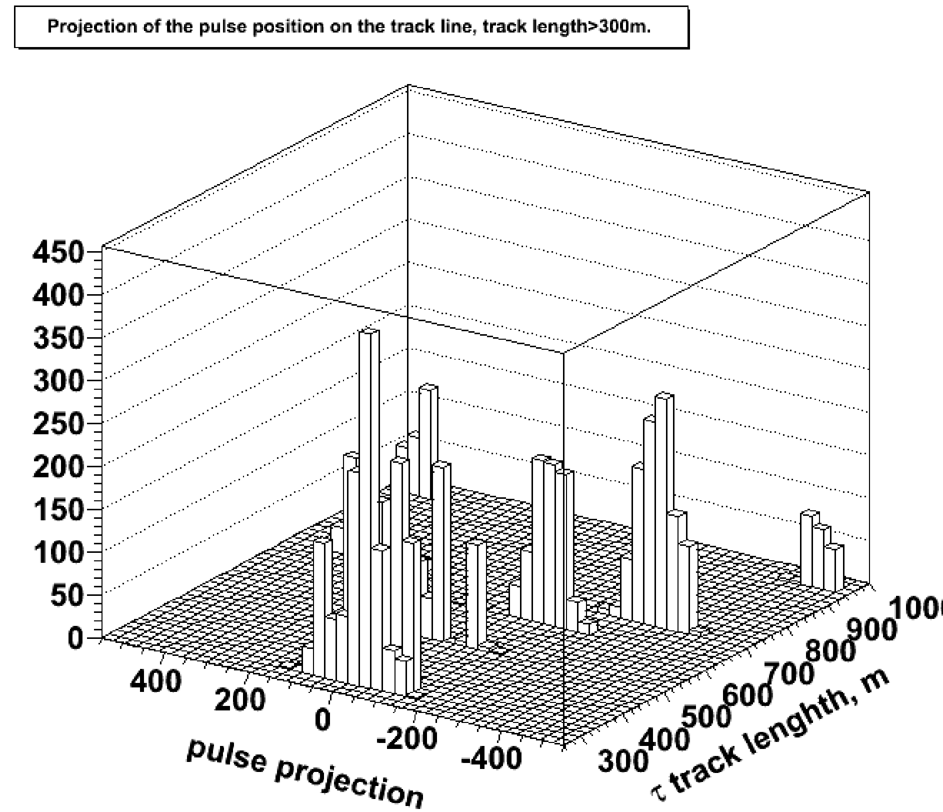




all pulses passed cut $charge_{pulse}>0.2*charge_{max}$.

Track projection versus track length



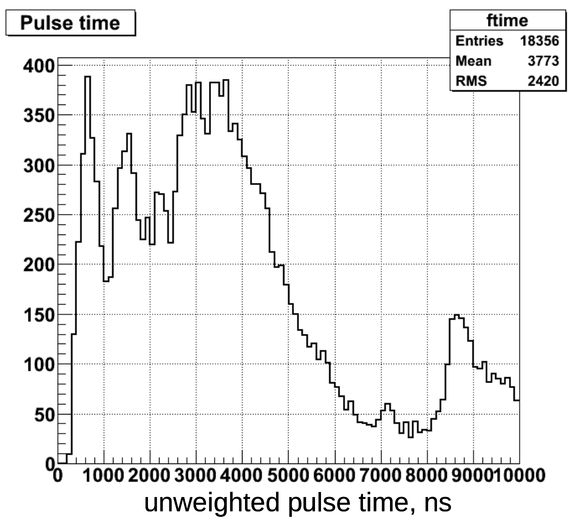


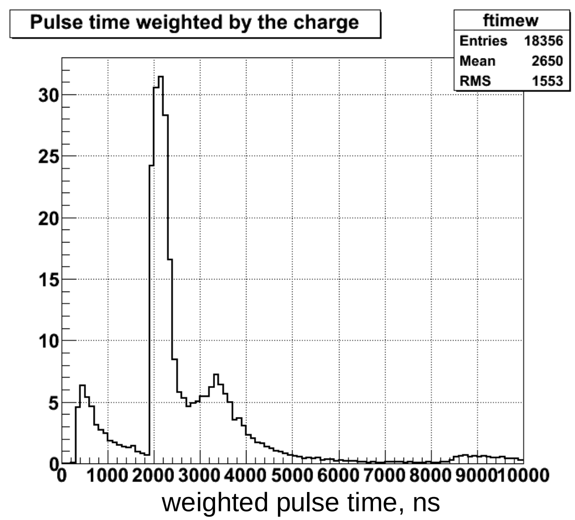
all pulses passed cut $charge_{pulse}>0.1*charge_{max}$.

Map splitting conditions

Time splitting:

Make a 100 bins function ftime, each bin 100ns. Fill it with charge weighted pulse time.





Use ftime find event time range:

- ftime>5 for the first time marks start event time.
- ftime<1 after the start time marks end event time. If there are more peaks that are higher then 5, end time is shifted to the end of the last high enough peaks.

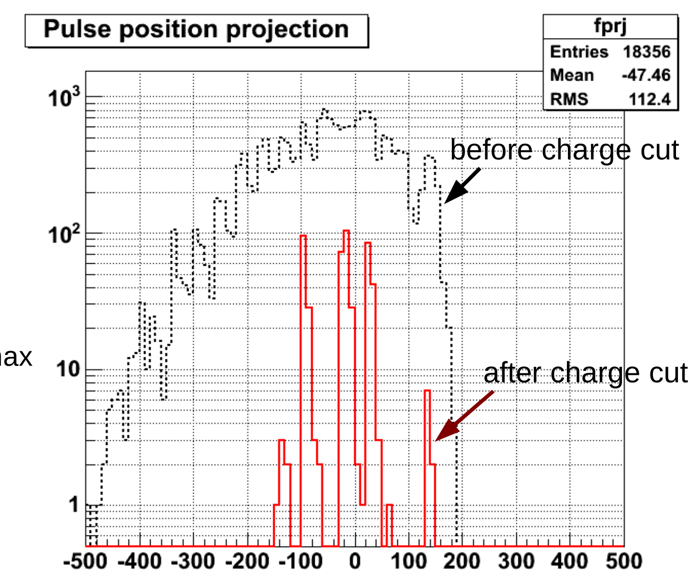
This example corresponds to event with L_{track}=510m, estimated time delay dt=1700ns

Track projection splitting:

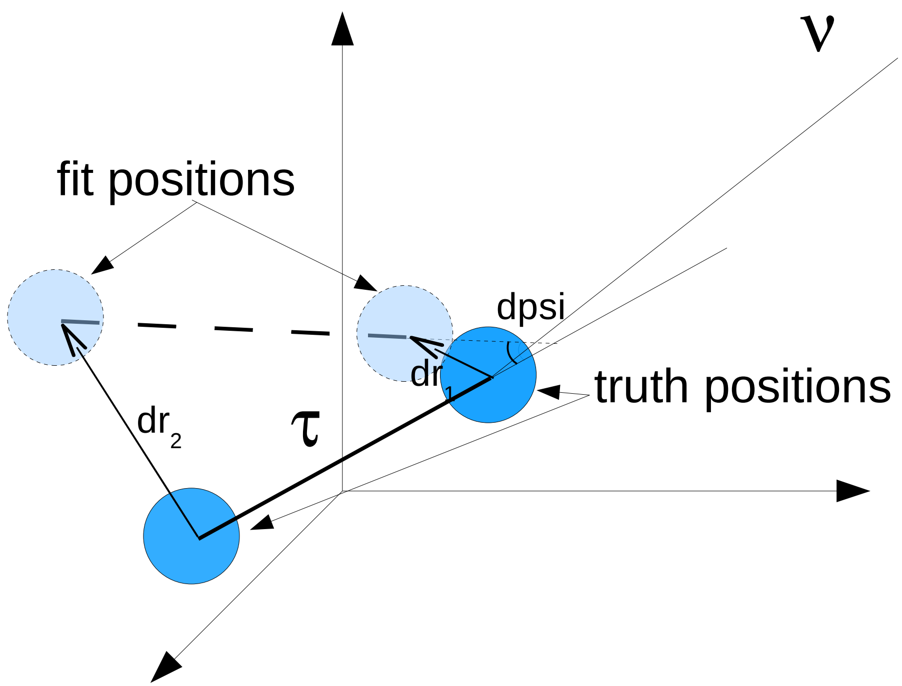
To avoid low charge pulses we apply cut $charge_{pulse}>0.1*charge_{max}$

The distribution is shifted, so its center is at 0.

All pulses with track projection>0 are in the first cascade, all pulses with track projection<0 are in the second one.



Fit introduction



Seed and result of the double pulse fit are two particles. Transition between 2 I3Particles and fit variables is done by I3TauDPParametrization module.

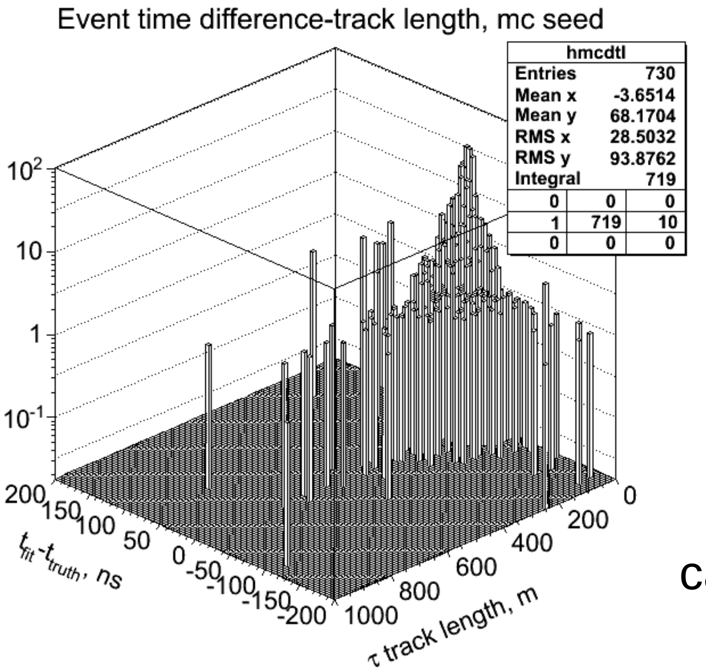
Single cascade fit uses 7 variables:

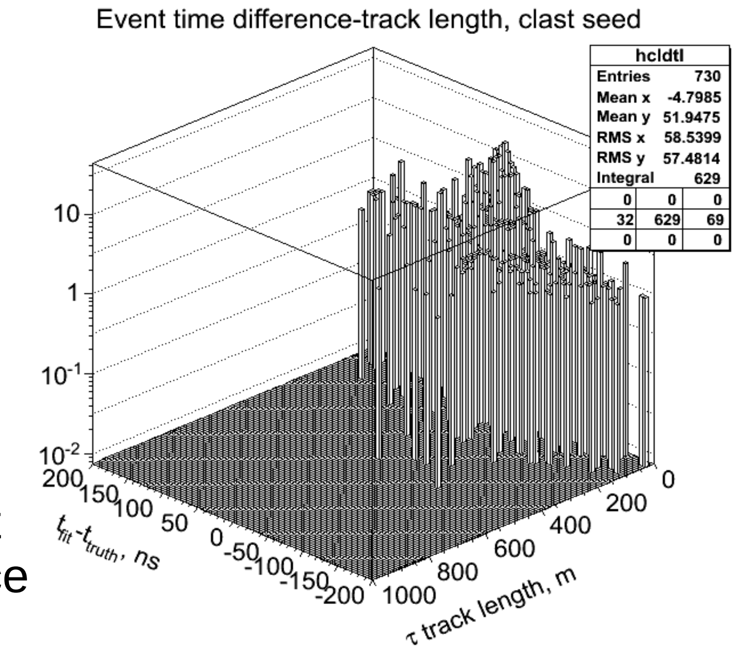
event time, 3 coordinates, 2 direction angles and energy.

Double pulse fit needs 9 variables:

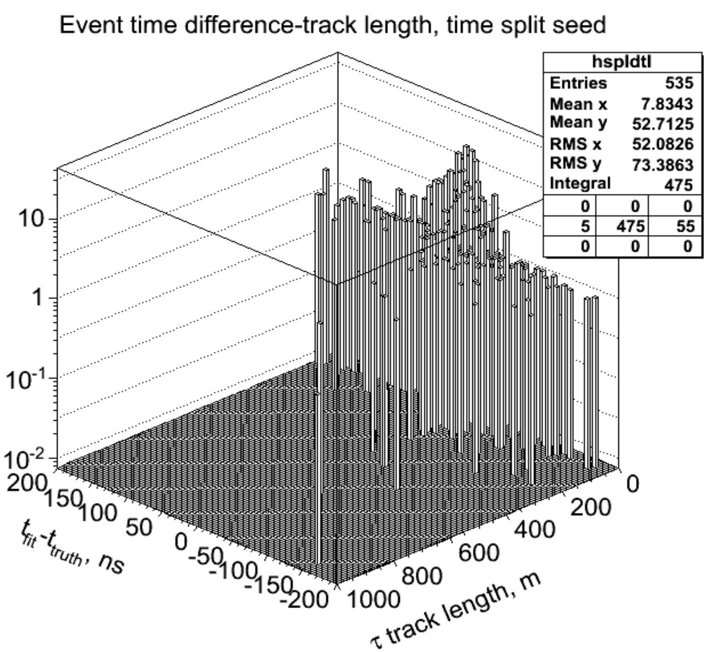
the first cascade time and its 3 coordinates, 2 direction angles (the same for both cascades), $\log_{10}$ of total energy, $\log_{10}(E1/E2)$ and distance between cascades (track length).

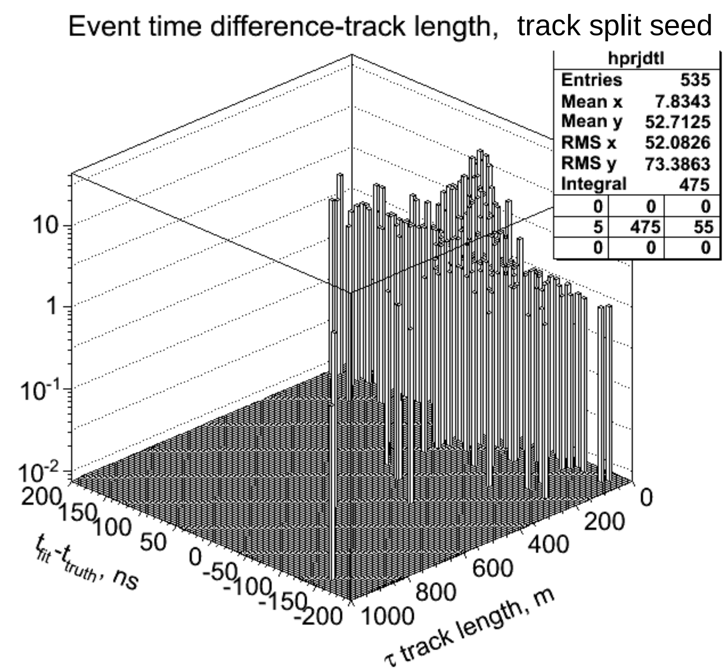
Time difference



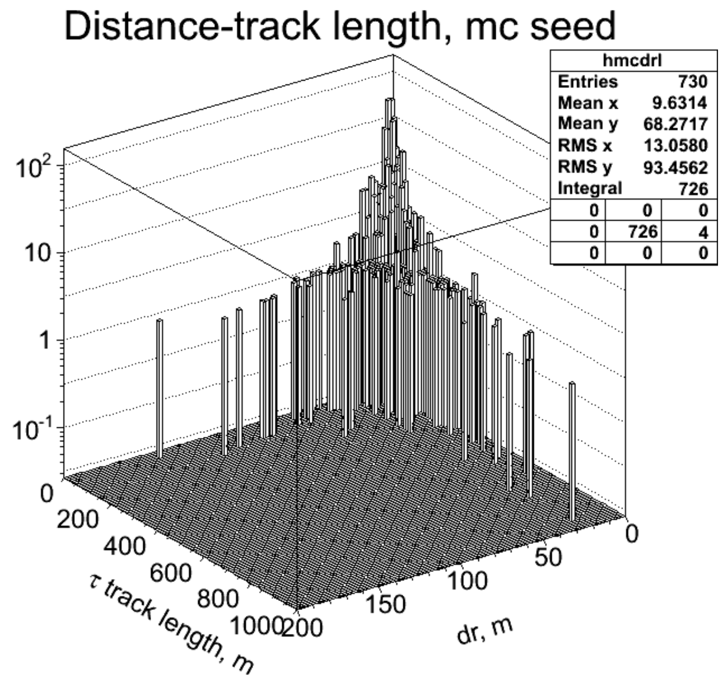


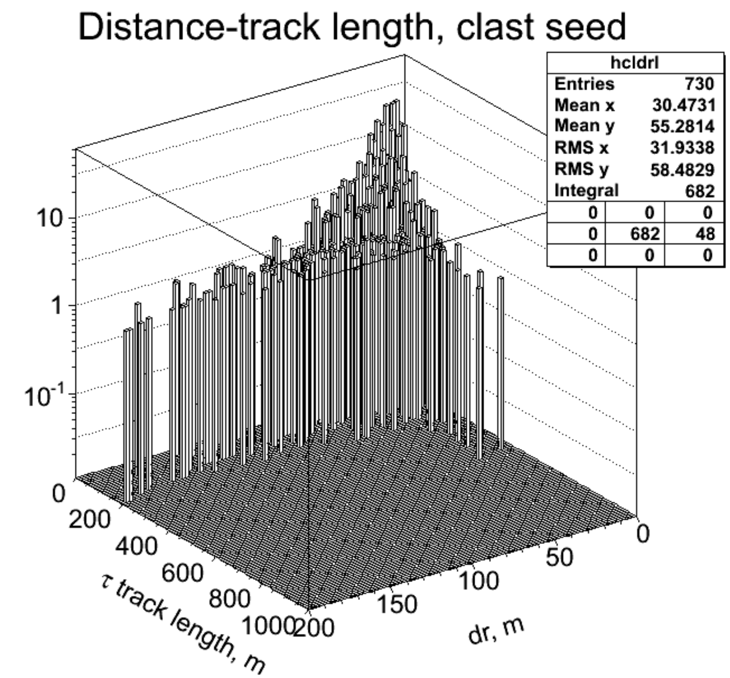
x-axis is event (or first cascade) time difference
y-axis is track length
L_{track}>10m



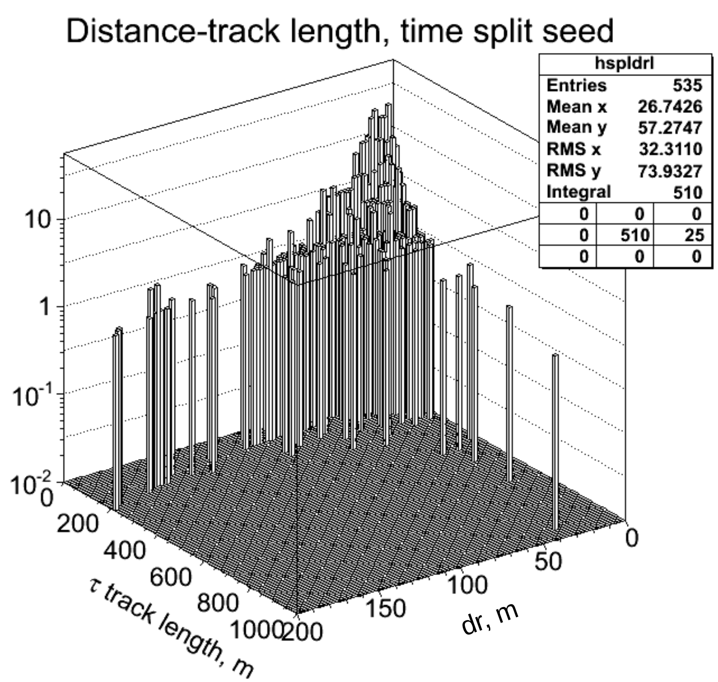


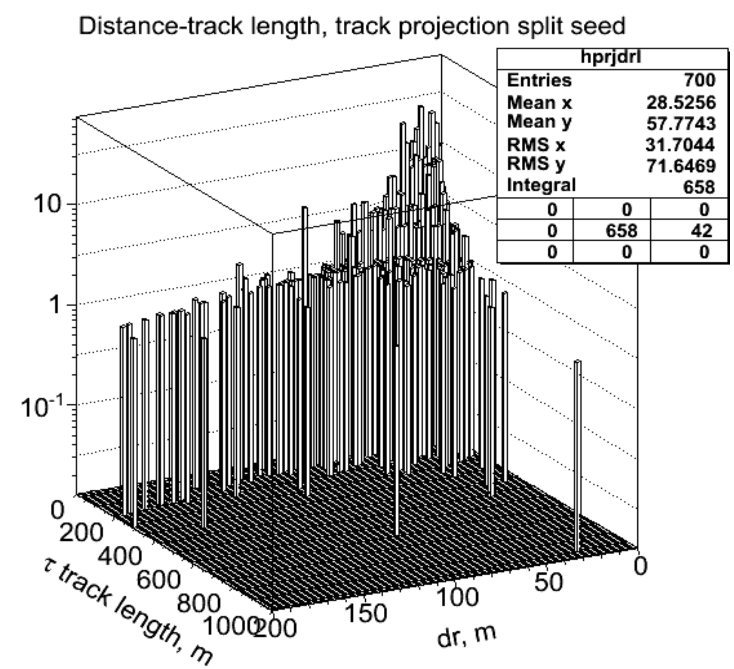
Position difference, first cascade



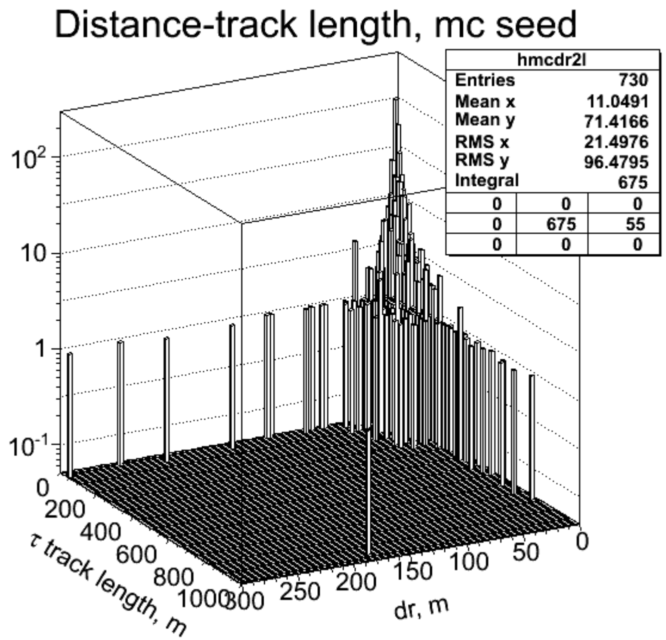


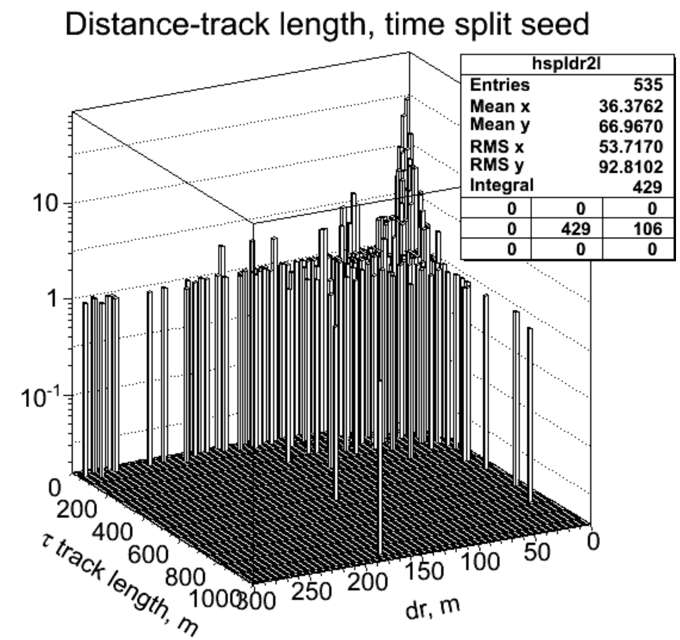
x-axis is position difference
y-axis is track length



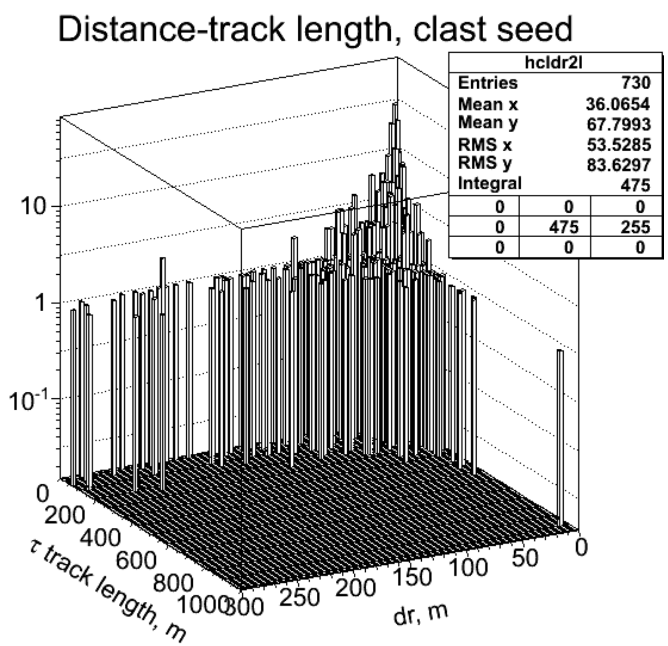


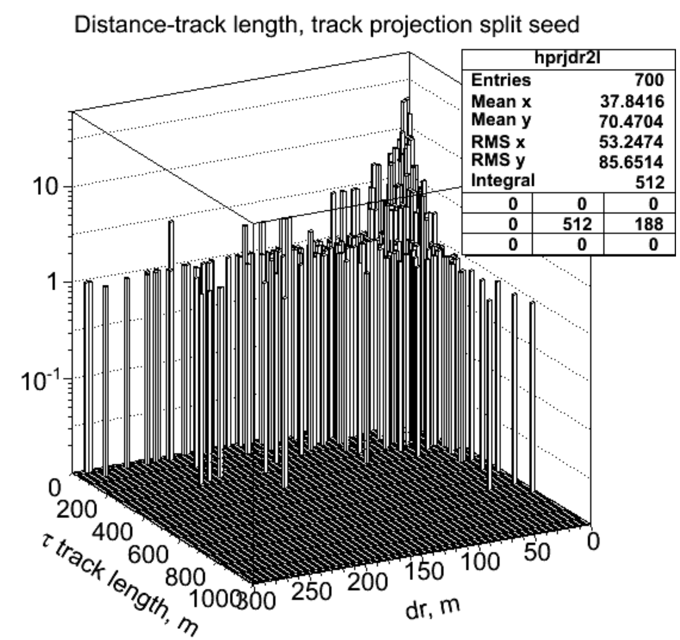
Position difference, second cascade



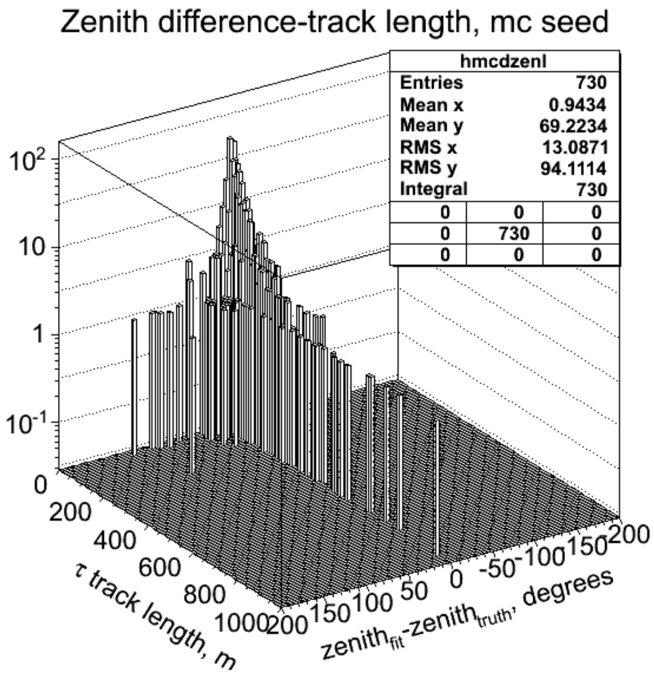


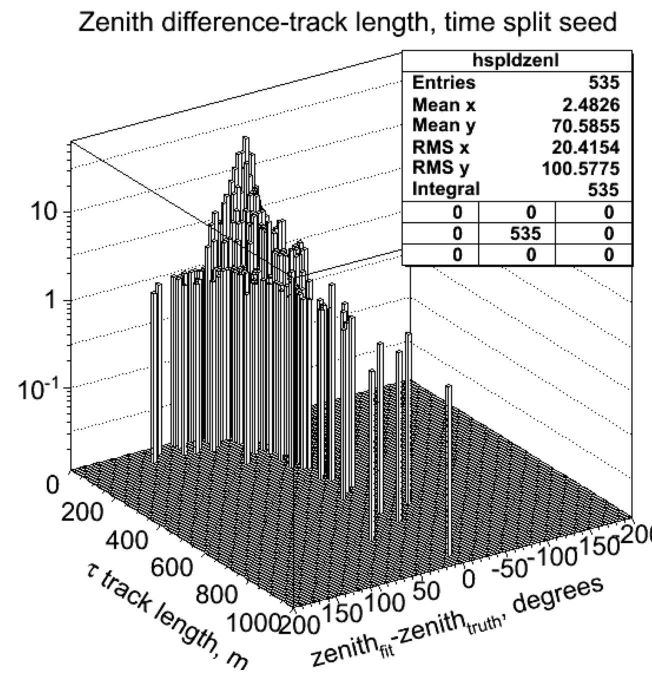
x-axis is position difference
y-axis is track length
L_{track}>10m





Zenith difference

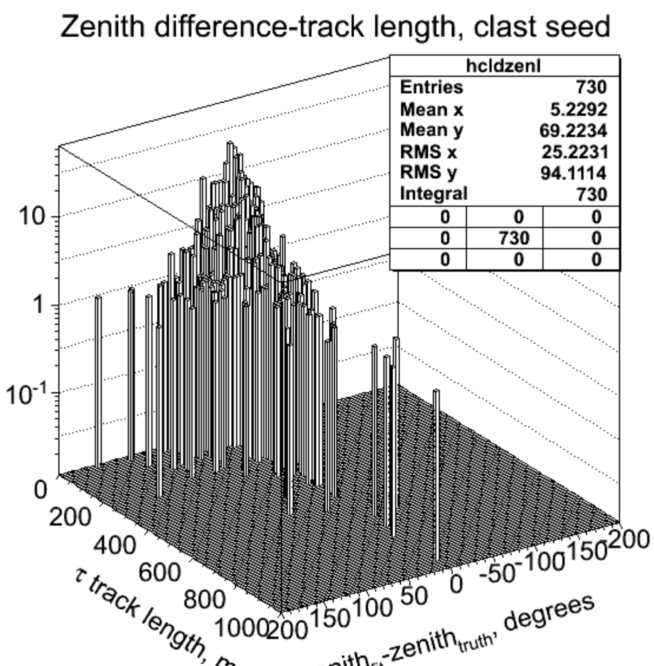


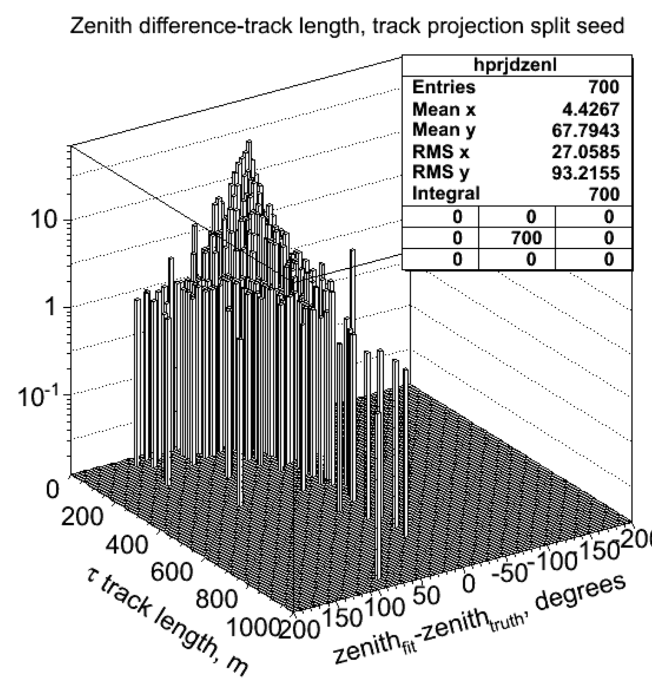


x-axis is zenith difference

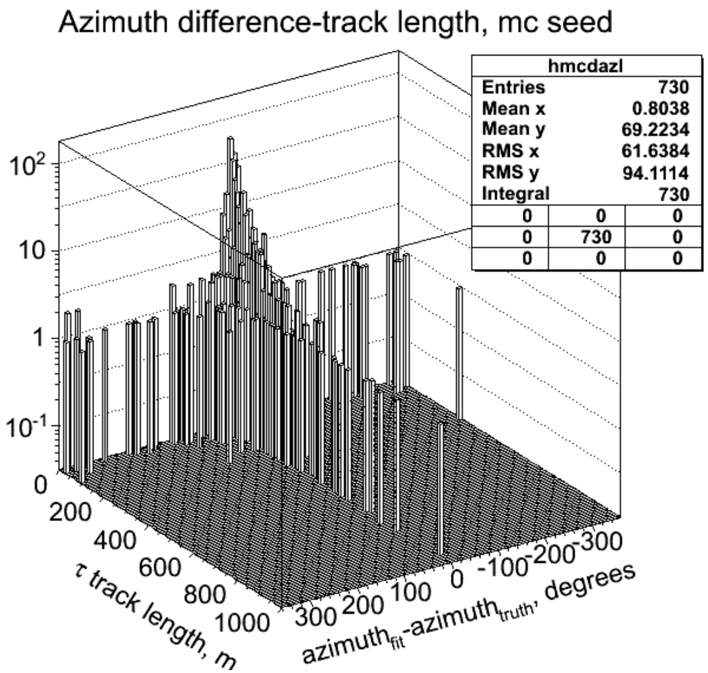
y-axis is track length

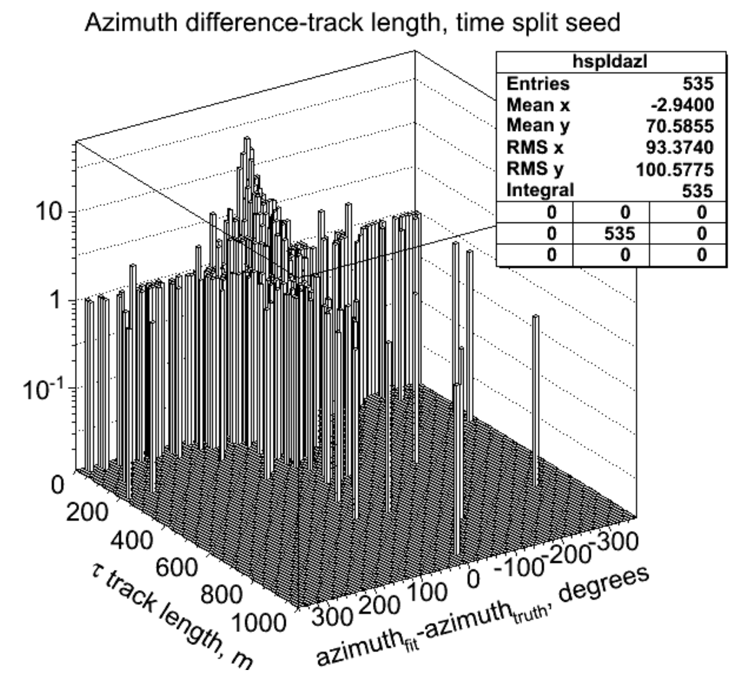
L_{track}>10m



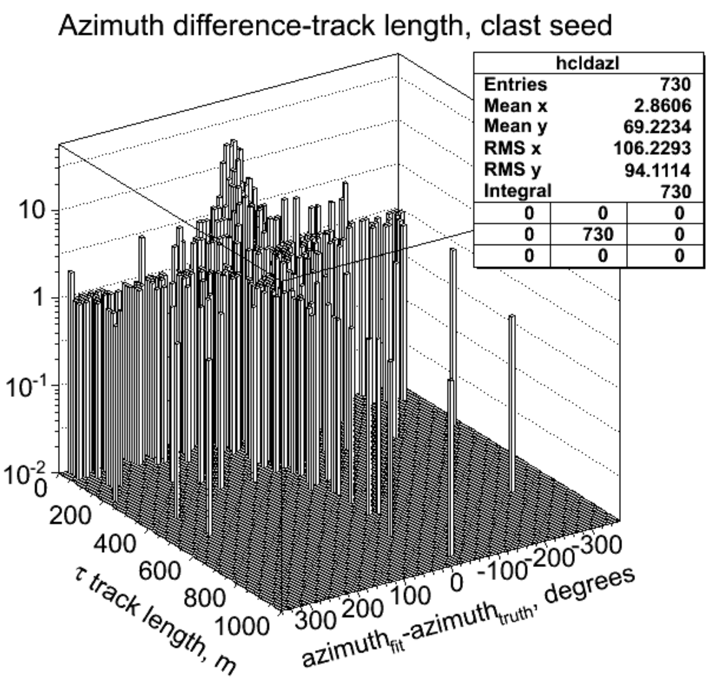


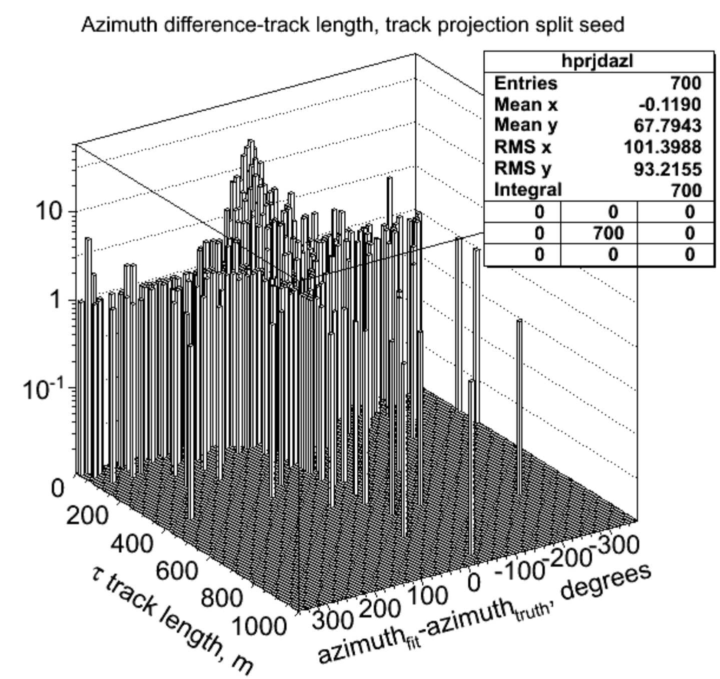
Azimuth difference



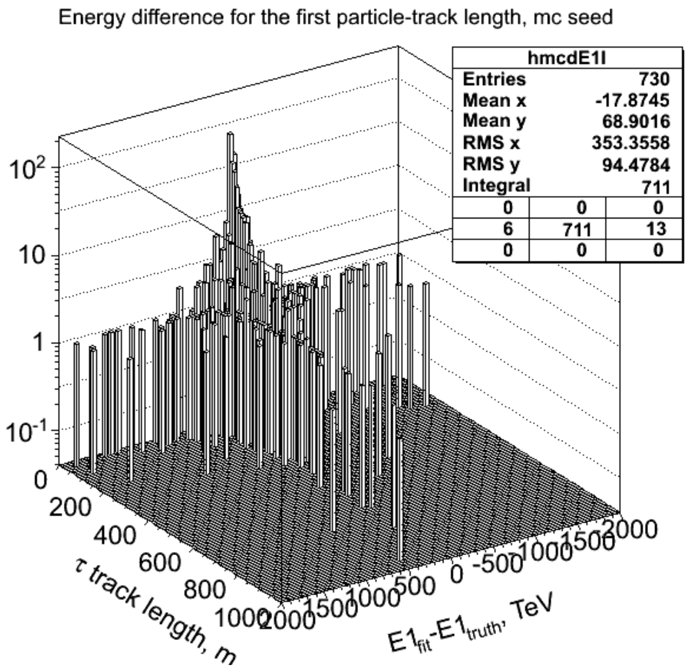


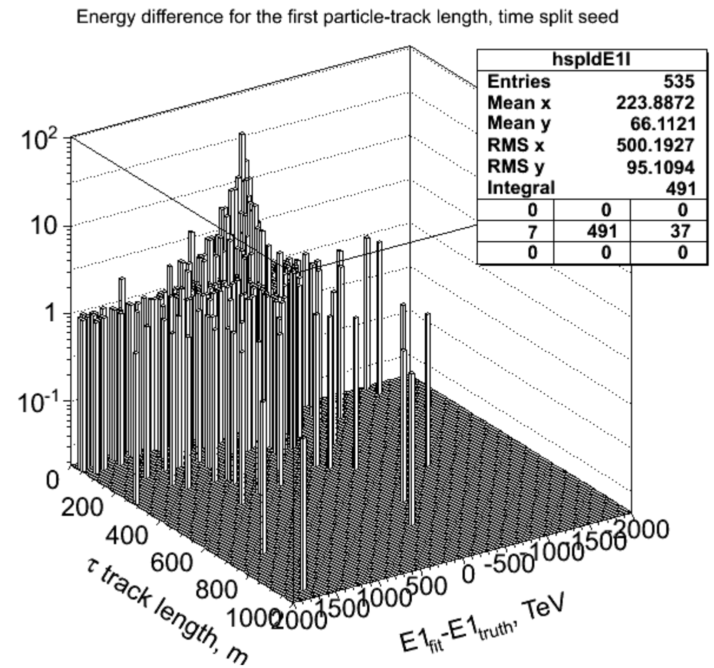
x-axis is azimuth difference
y-axis is track length
L_{track}>10m



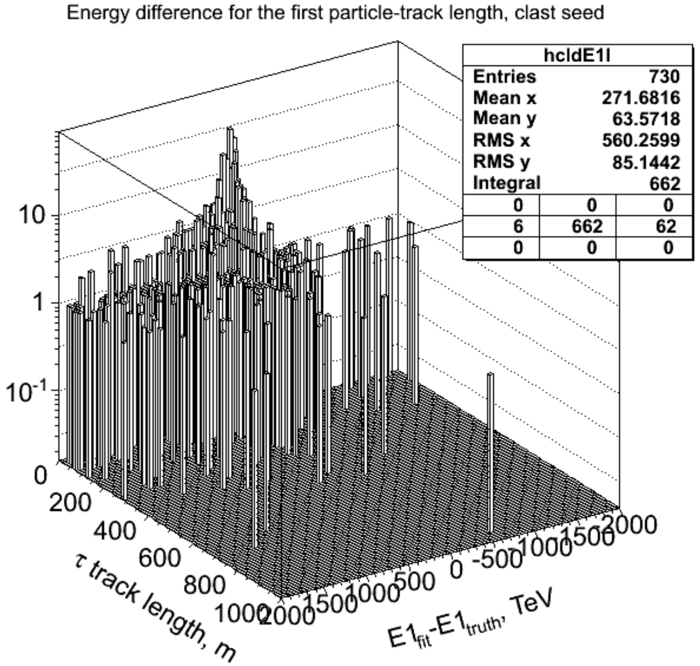


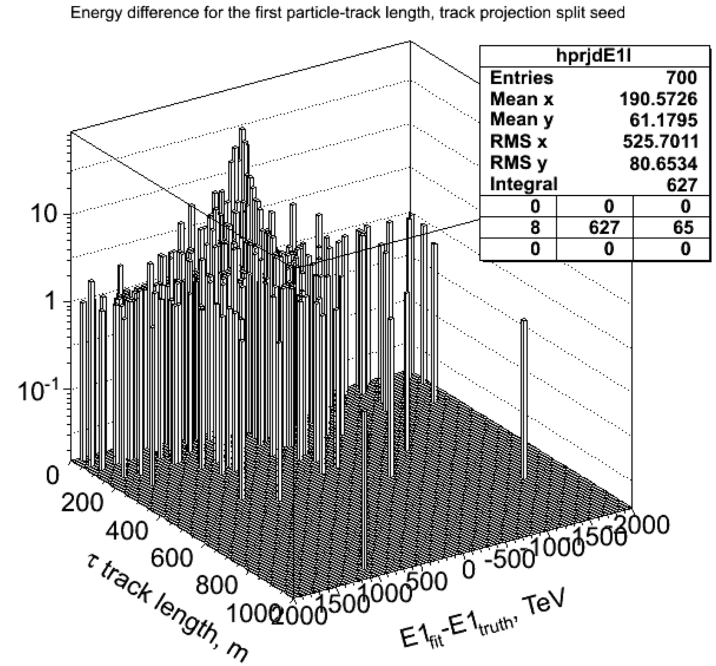
Energy difference



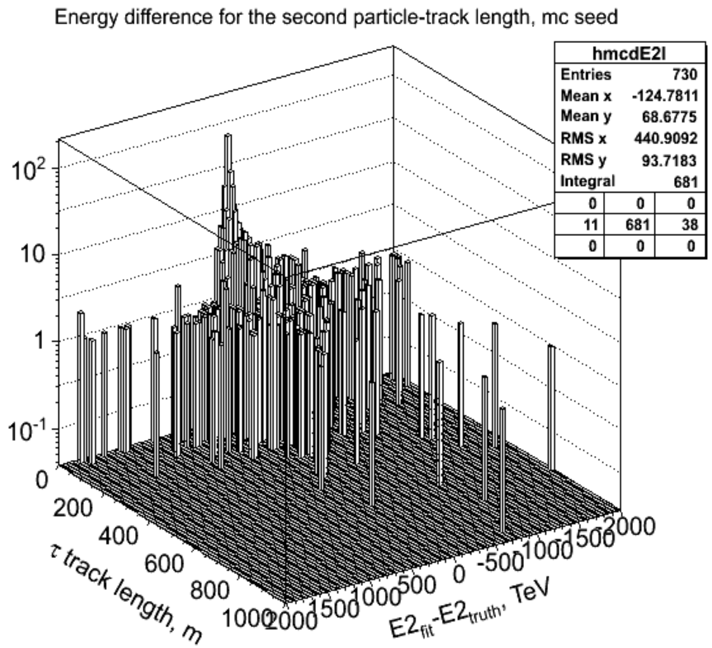


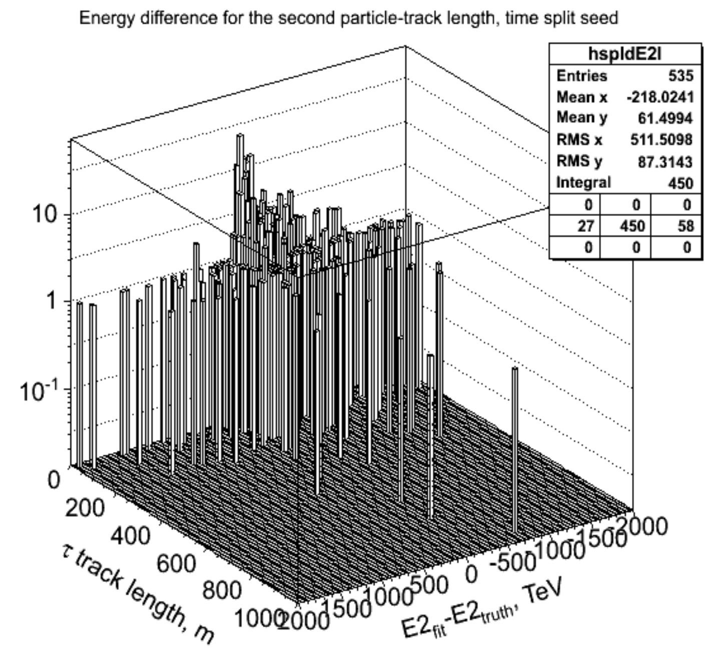
x-axis is energy difference
y-axis is track length
L_{track}>10m



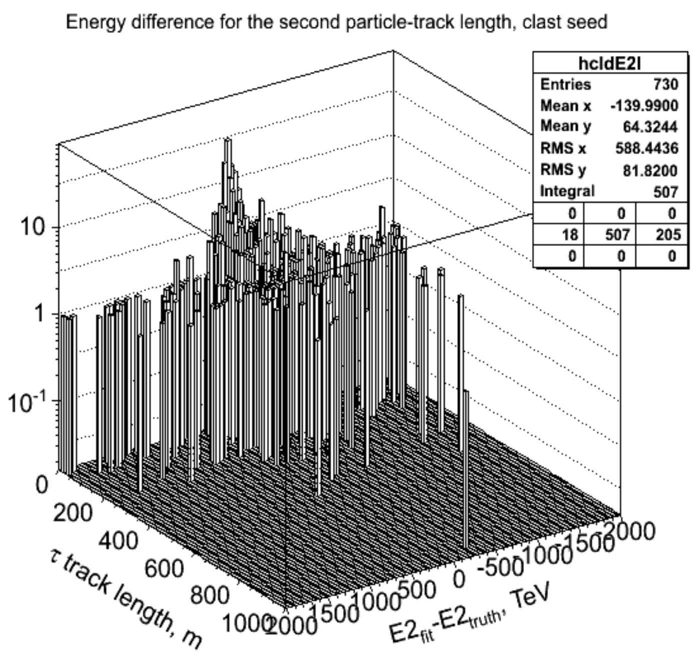


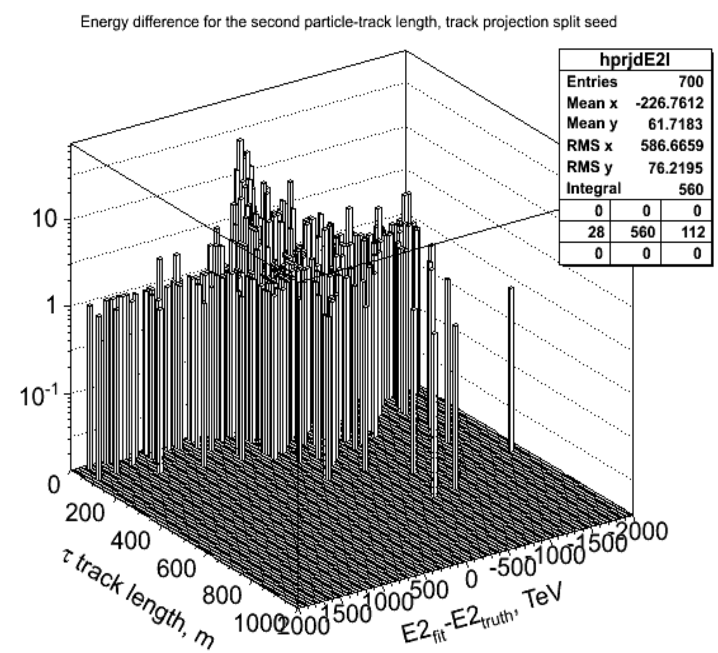
Energy difference



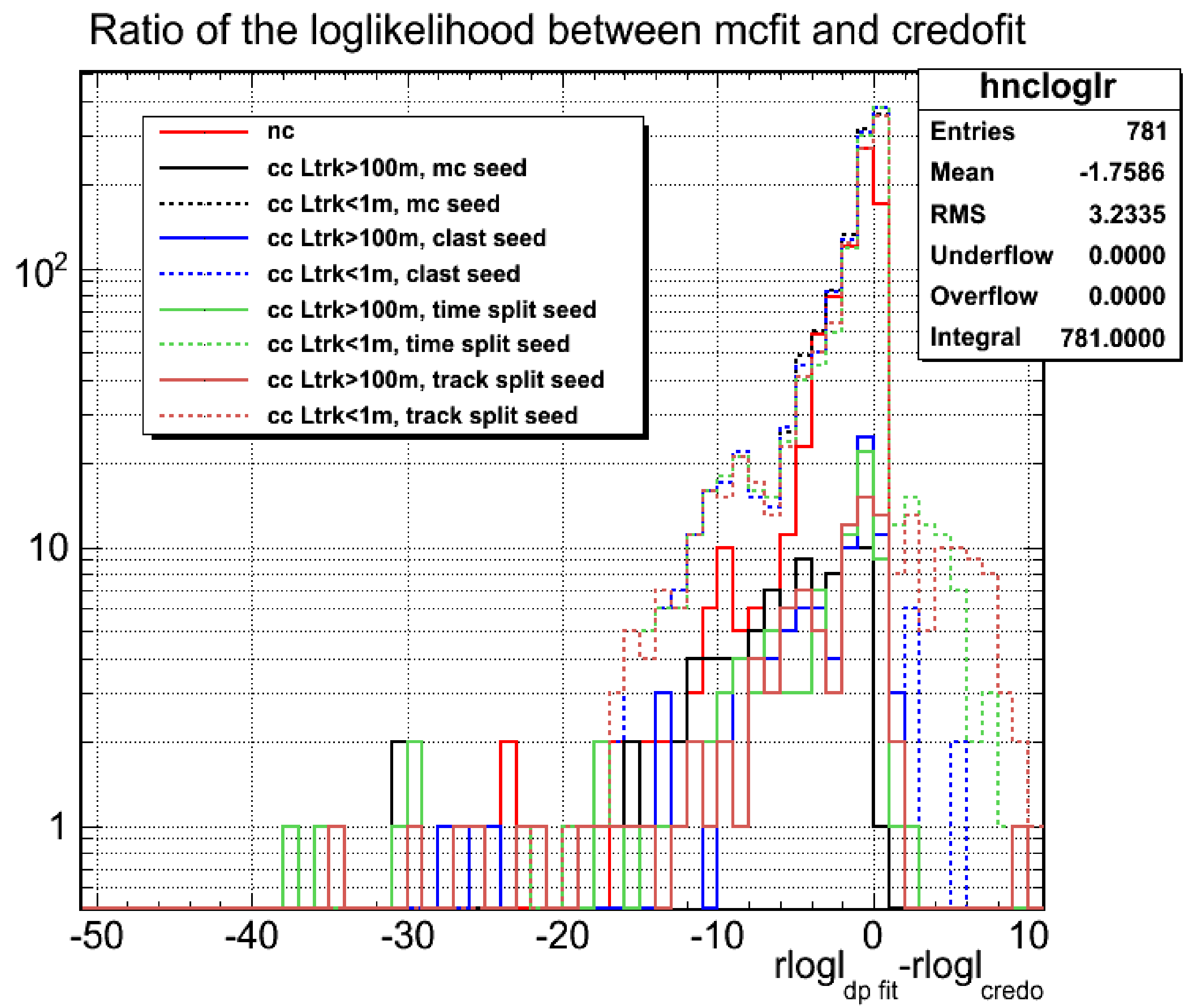


x-axis is energy difference
y-axis is track length
L_{track}>10m

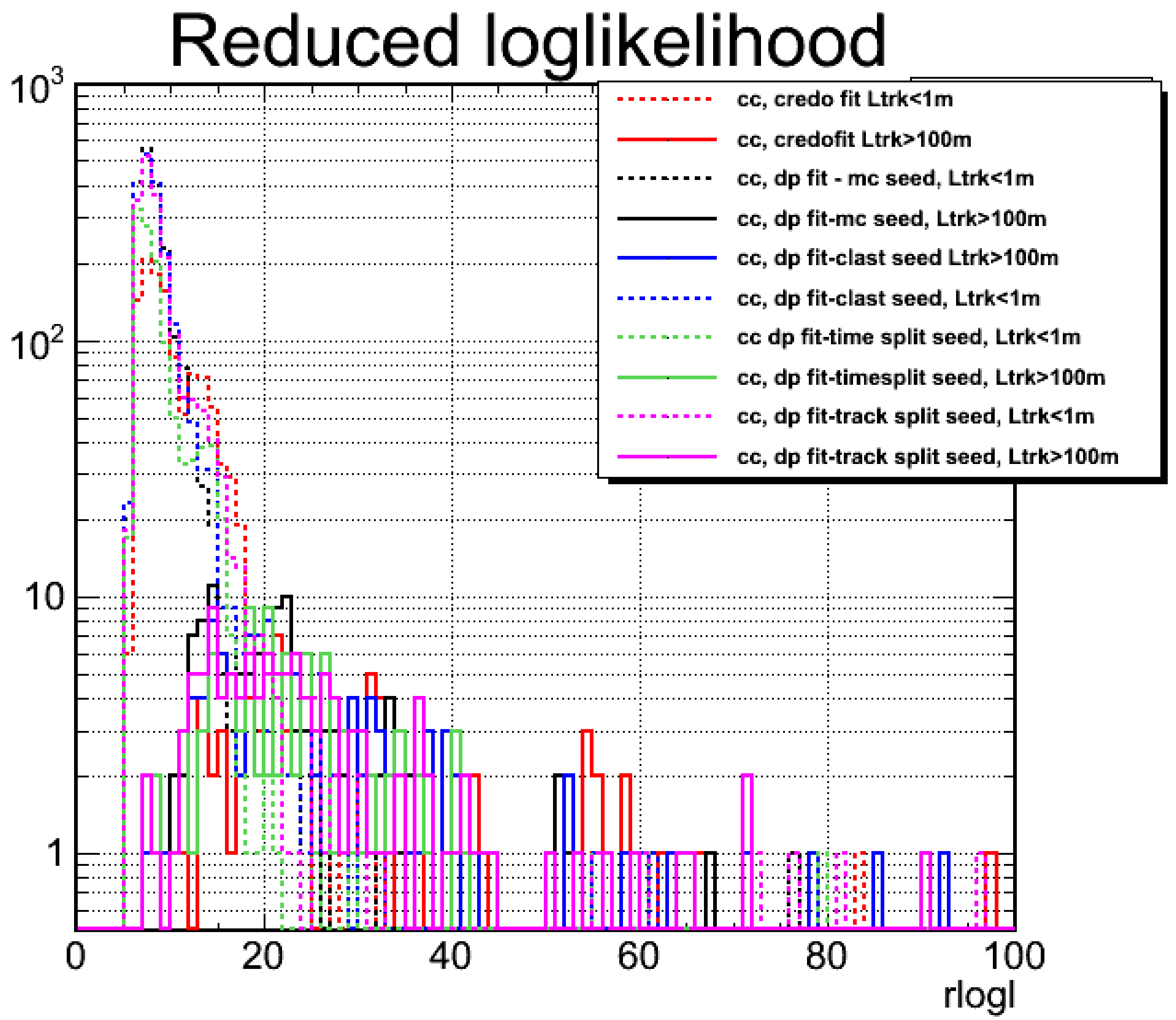




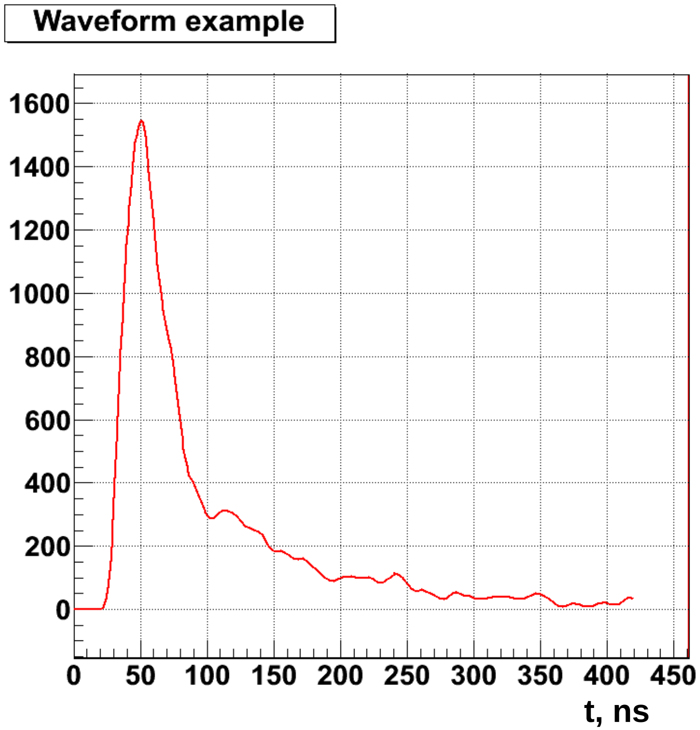
Reduced loglikelihood distributions



Reduced loglikelihood distributions

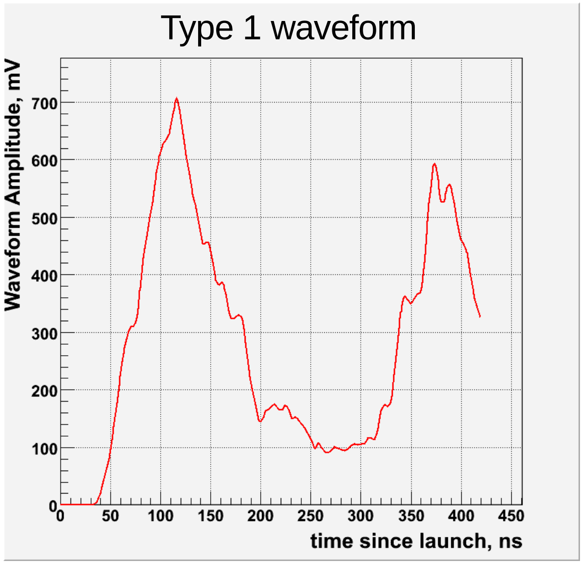


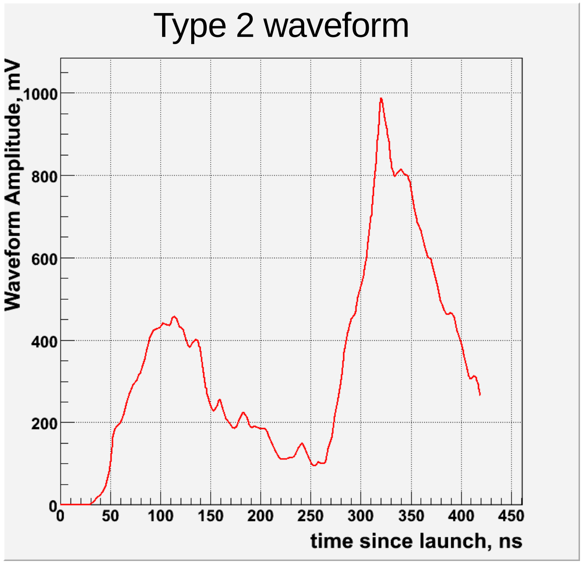
Type 2 waveforms



Normal IceCube waveform has one peak at the start of the event.

In case of charged current ν_τ event two cascades are produced. In this case double pulse waveform might be made when light from both cascades is detected in one waveform time window.





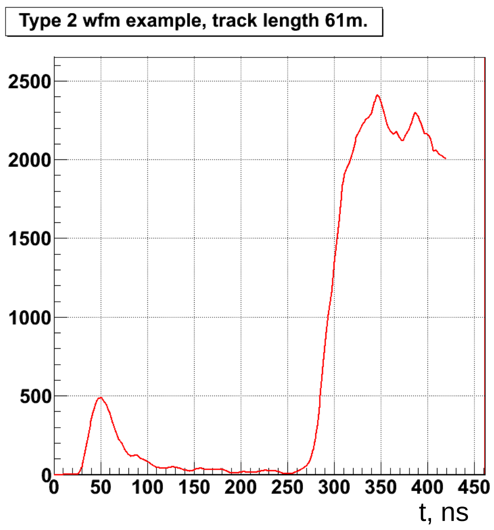
If the first peak of the waveform is higher than the second this is a type 1 waveform.

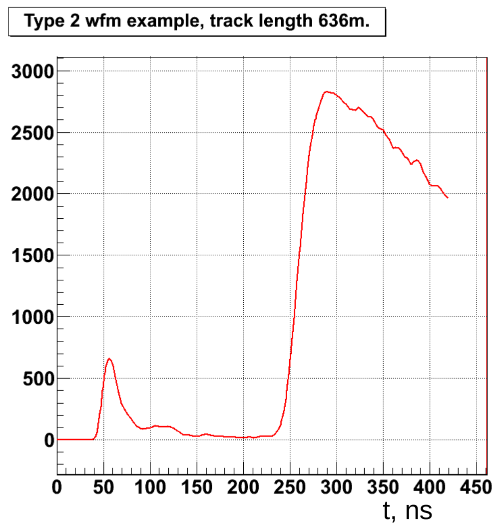
If the second peak is higher – this is a type 2 waveform.

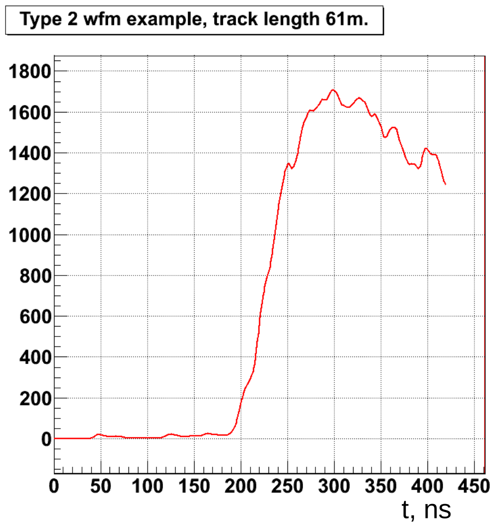
Examples here are from flasher run 118503 (simulating double pulse in Deep Core).

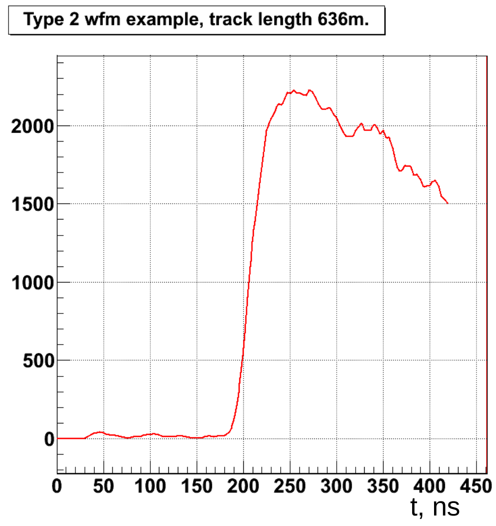
Type 2 waveforms

Waveform examples from our data set



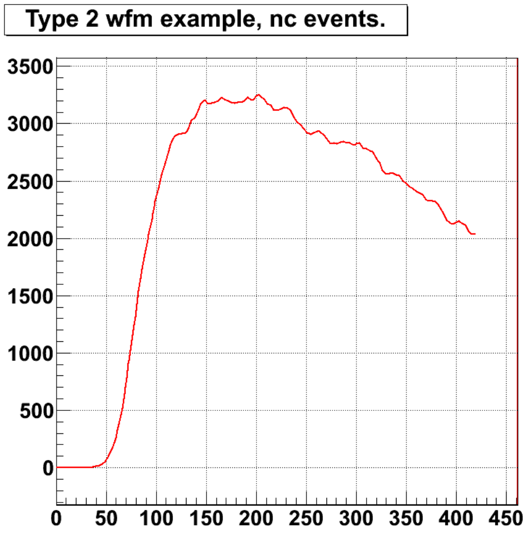






The first pulse can be relatively low or even almost negligible, producing waveform that looks like it has a delayed pulse.

Here are several examples of type 2 waveform for different track lengths (charge current events) and a type 2 waveform produced by neutral current event.



Type 2 waveforms

Part of the events that has at least one type 2 waveform.

	t_{max}>33ns	t_{max}>66ns	t_{max}>99ns	t_{max}>132ns	t_{max}>165ns	t_{max}>198ns	t_{max}>231ns	t_{max}>264ns	t_{max}>297ns
NC	**0.228**	**0.192**	**0.096**	**0.053**	**0.027**	**0.012**	**0.003**	**0.001**	**0.000**
CC, L<10m	**0.098**	**0.060**	**0.012**	**0.005**	**0.002**	**0.001**	**0.001**	**0.000**	**0.000**
CC, 10m<L <300m	**0.724**	**0.710**	**0.577**	**0.448**	**0.343**	**0.252**	**0.194**	**0.147**	**0.104**
CC, L>300m	**0.968**	**0.968**	**0.937**	**0.884**	**0.800**	**0.695**	**0.579**	**0.516**	**0.442**

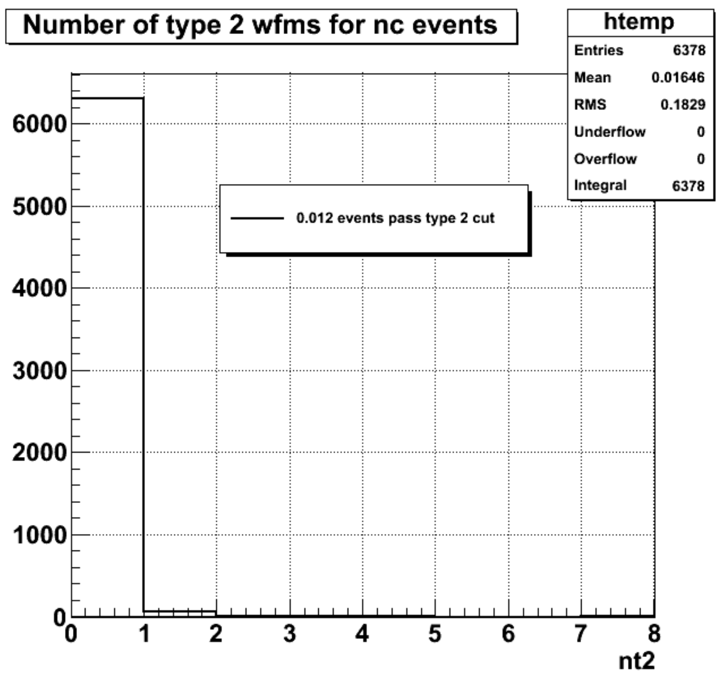
Type 2 waveform selection:

- waveform maximum should be after 198 ns
- maximum itself should be higher then 1000 mV

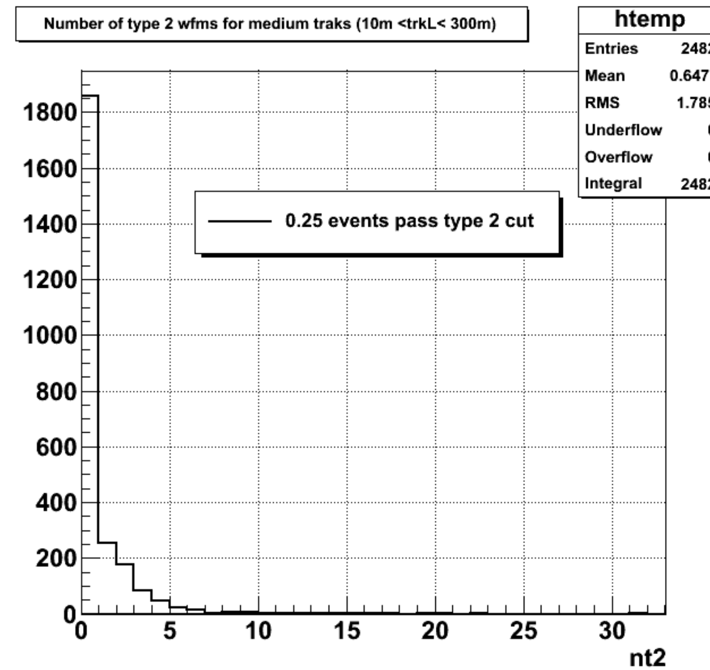
Type 2 waveforms can be produced by neutral current events as well, but there is much less of them and they look different.

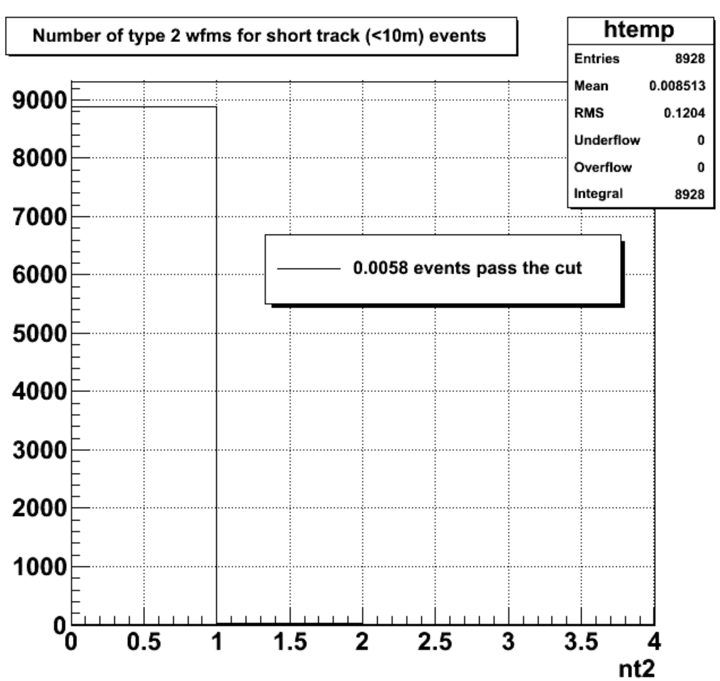
Large amount of low charge waveforms is cut down by the maximum cut

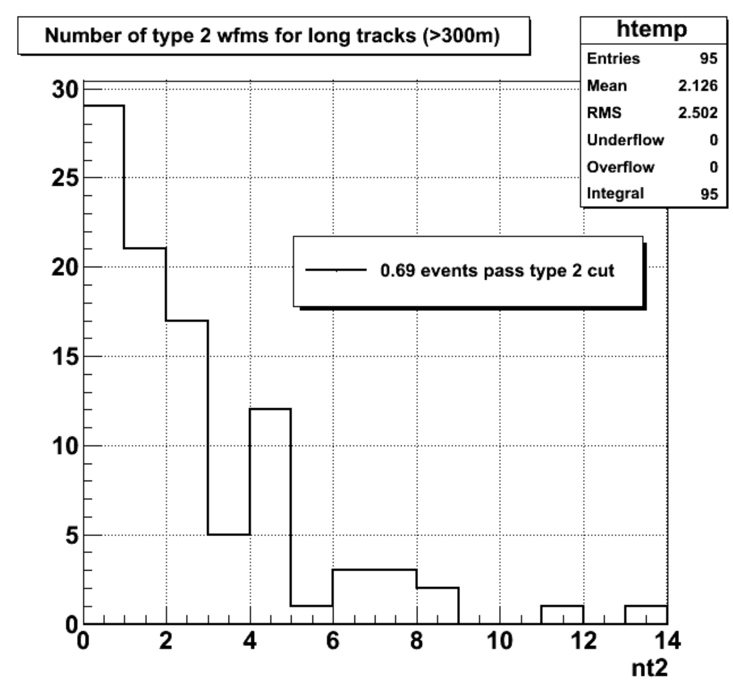
Type 2 cut



Type 2 cut means that event has at least one type 2 waveform







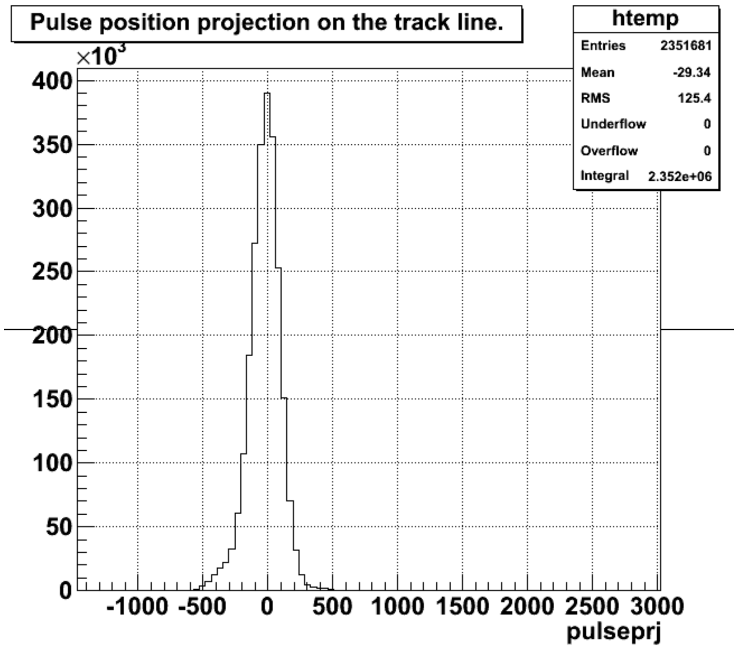
Summary and conclusions

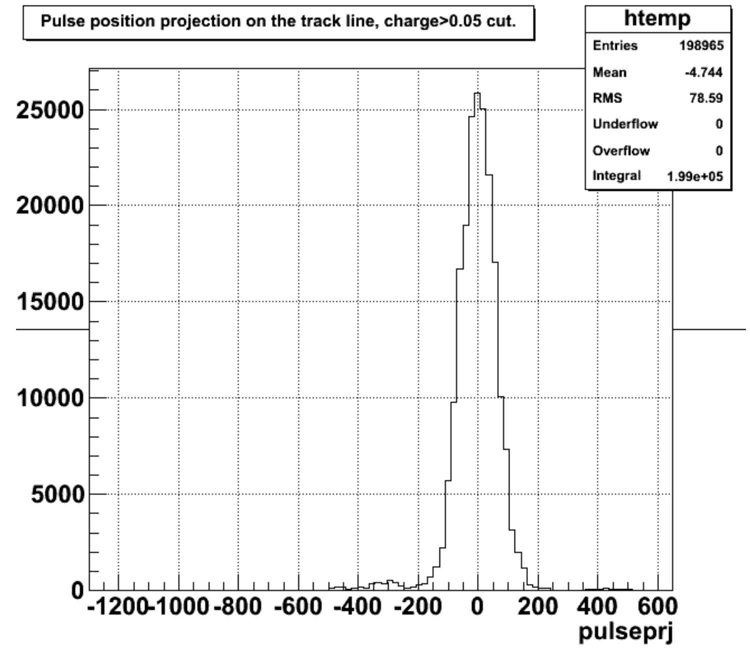
- fit seeded with split seeds produces slightly better results than fit using clast seed
- both seed splitting methods produce very similar results
- Loglikelihood distributions are not very different for double pulse fit and credo fit

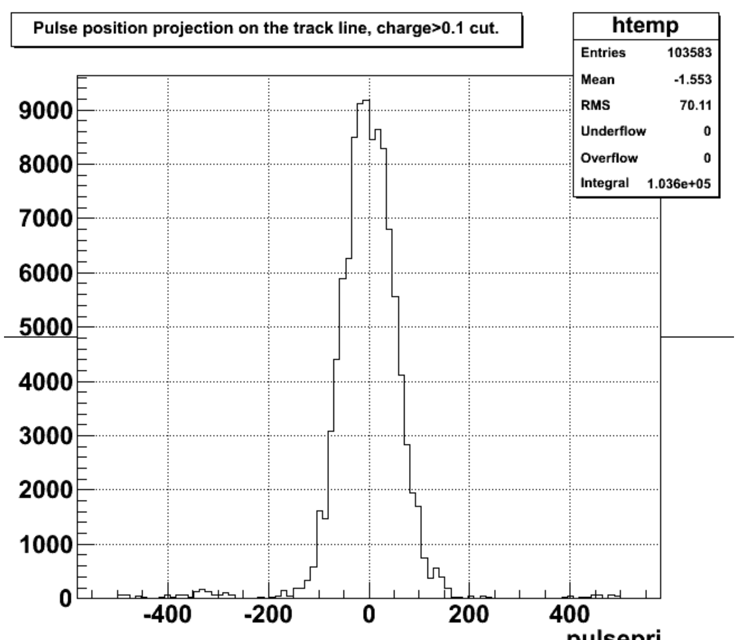
- Type 2 waveform cut provides a good discriminating power but not very good efficiency.

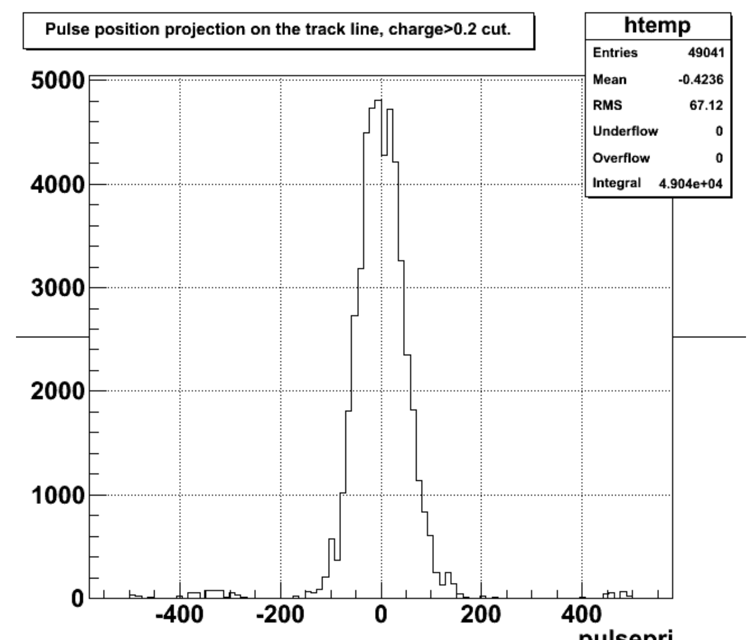
Backup

Position projection for all tracks ($l_{\tau\ track} > 30m$)









Pulse time for all tracks ($l_{\tau\ track}$>30m)