Neutron / gamma separation using scintillation signal in LAr

Oleynikov Vladislav
Italy, Napoli, December 2016

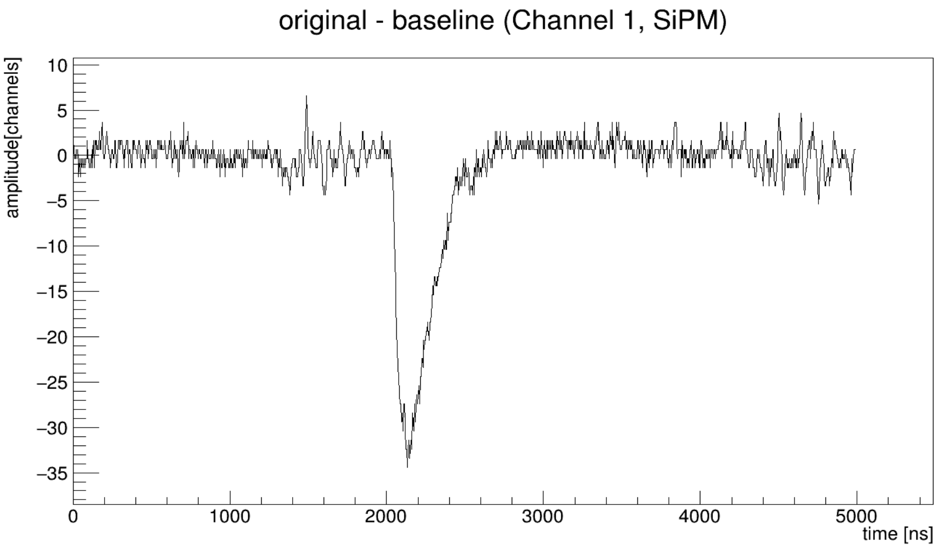
Content

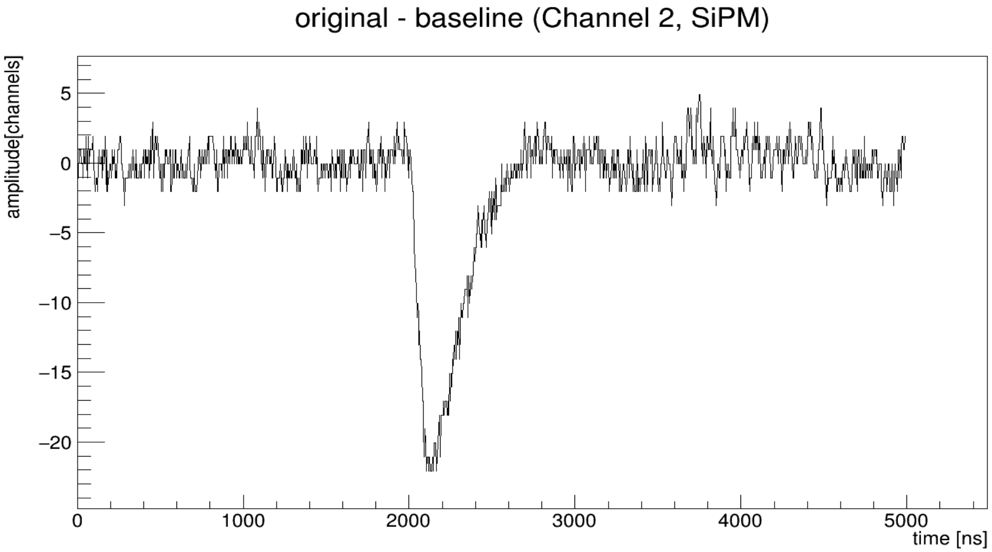
- Separation in theory
- Detector description
- Laser run 6061
  - Typical signals
  - averaged signals
  - integration spectra
  - Fourier transform
- Am run 6064
  - Typical signals
  - averaged signals
  - Fourier transform
  - Inverse Fourier transform
- Optimum in theory
- Signal reconstruction
  - Find derivative and identify peaks
  - Approximation by sum of individual responses
  - Deconvolution
- Deconvolution algorithms
  - Root CERN Tspectrum::Deconvolution
- Ratio vs N_pe
- Sigmas comparison
- What are reasons of broadening?
- Further steps

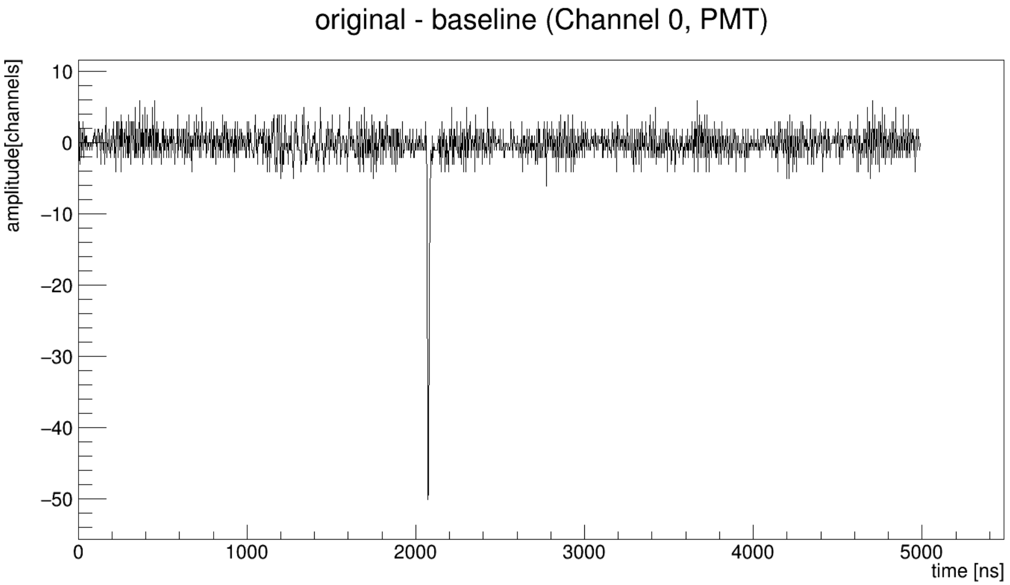
Separation in theory

Detector description

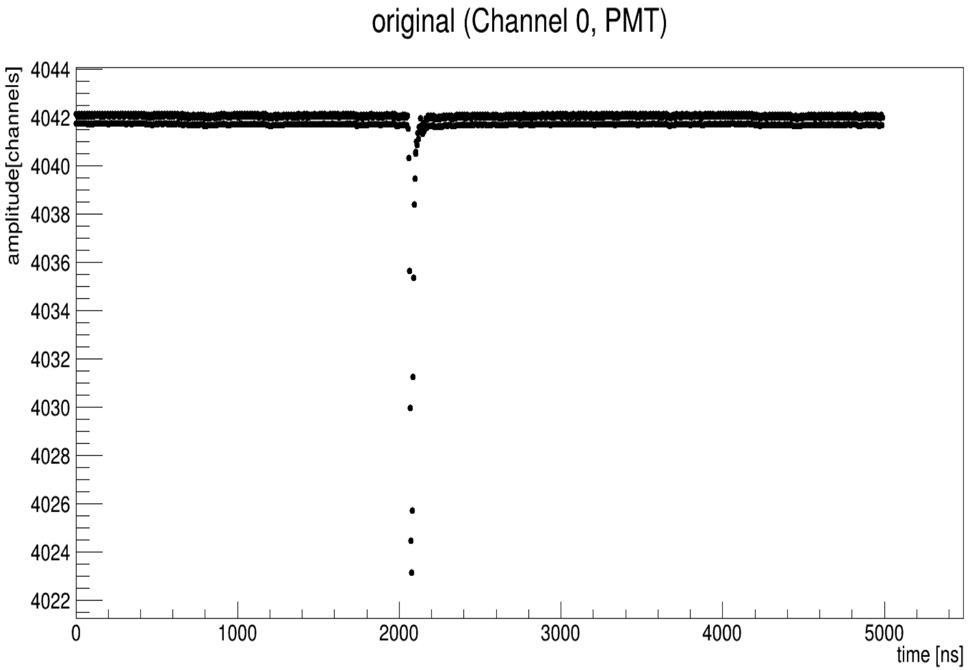
Laser run 6061. Typical signals

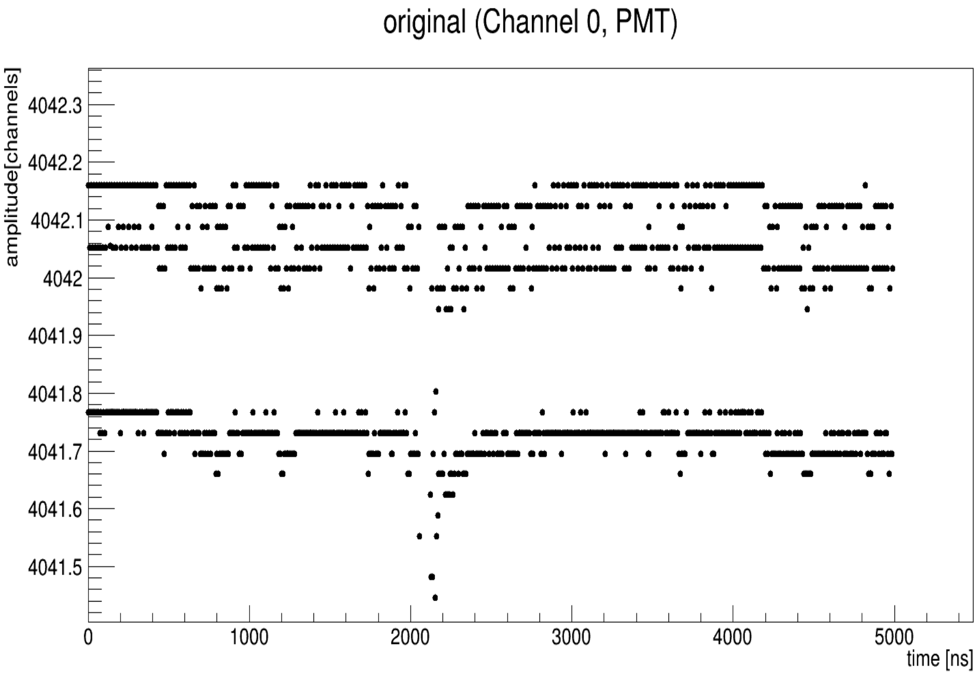


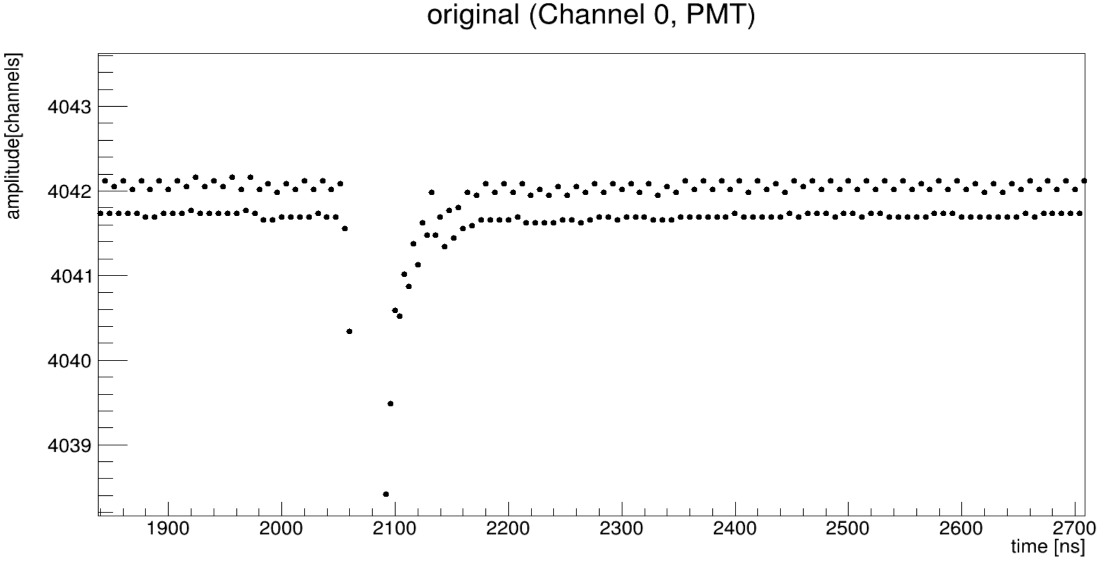




Laser run 6061. Averaged signals

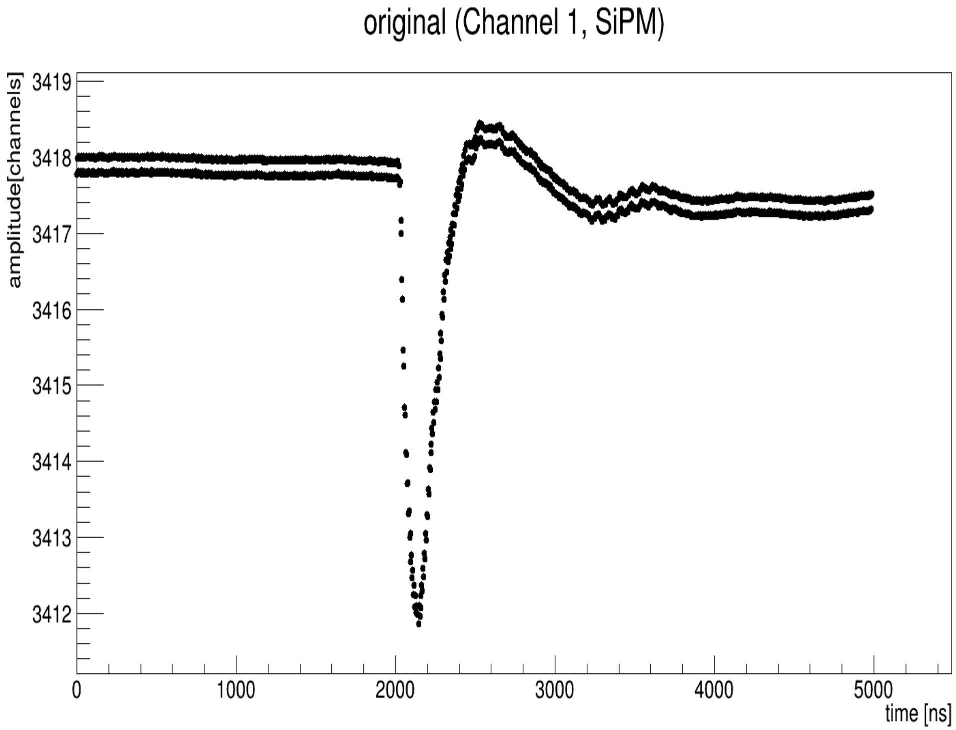


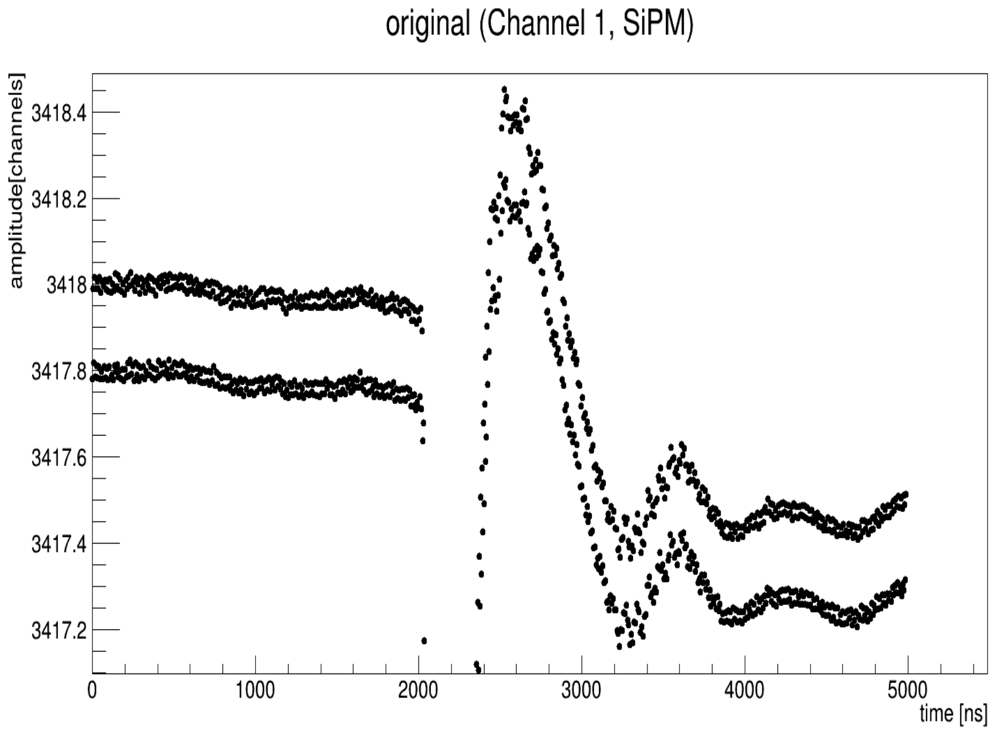


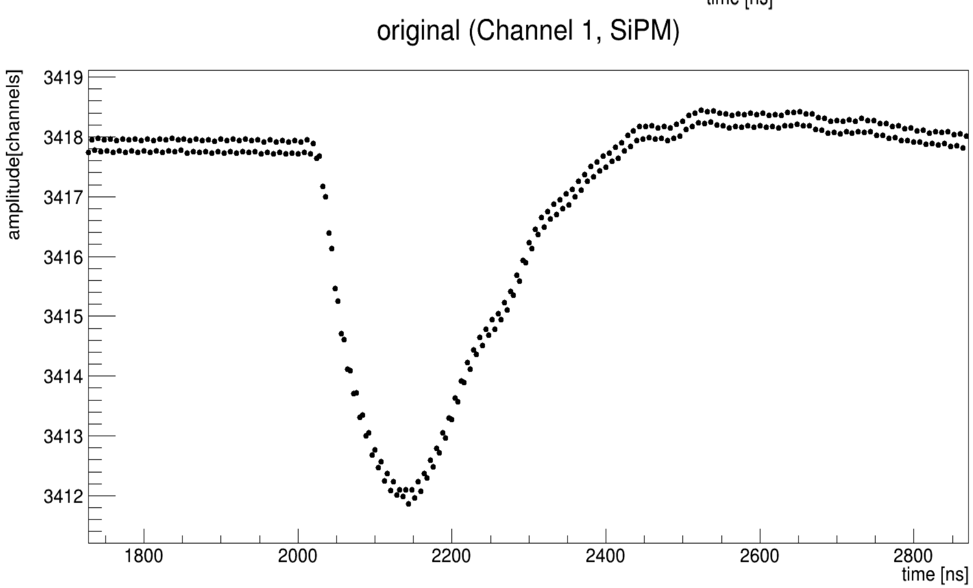


Avr_signal[i] += signal[i] / n_events

Laser run 6061. Averaged signals

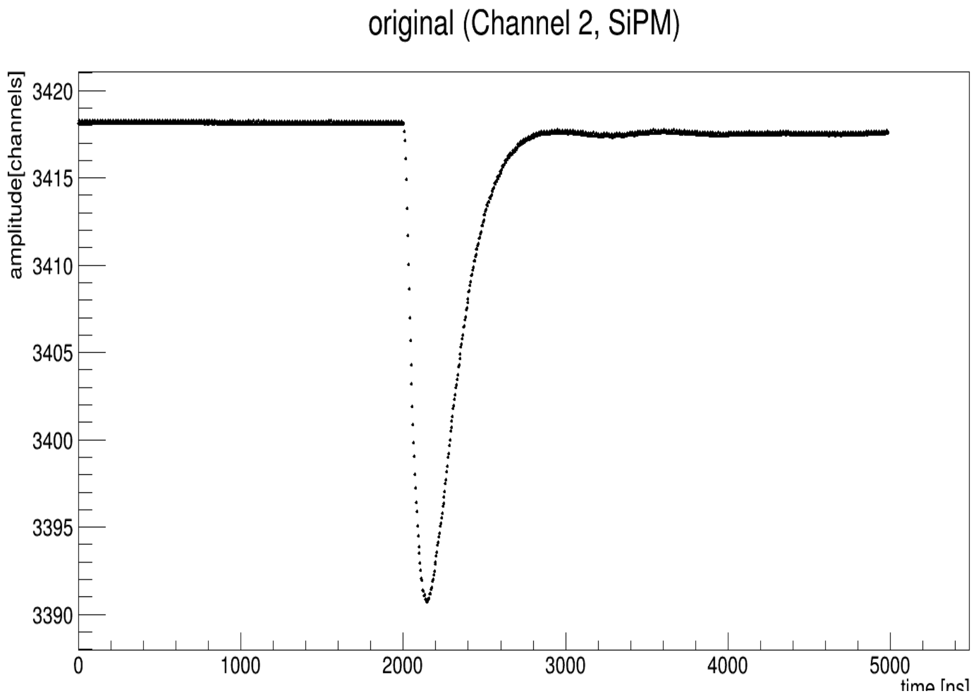


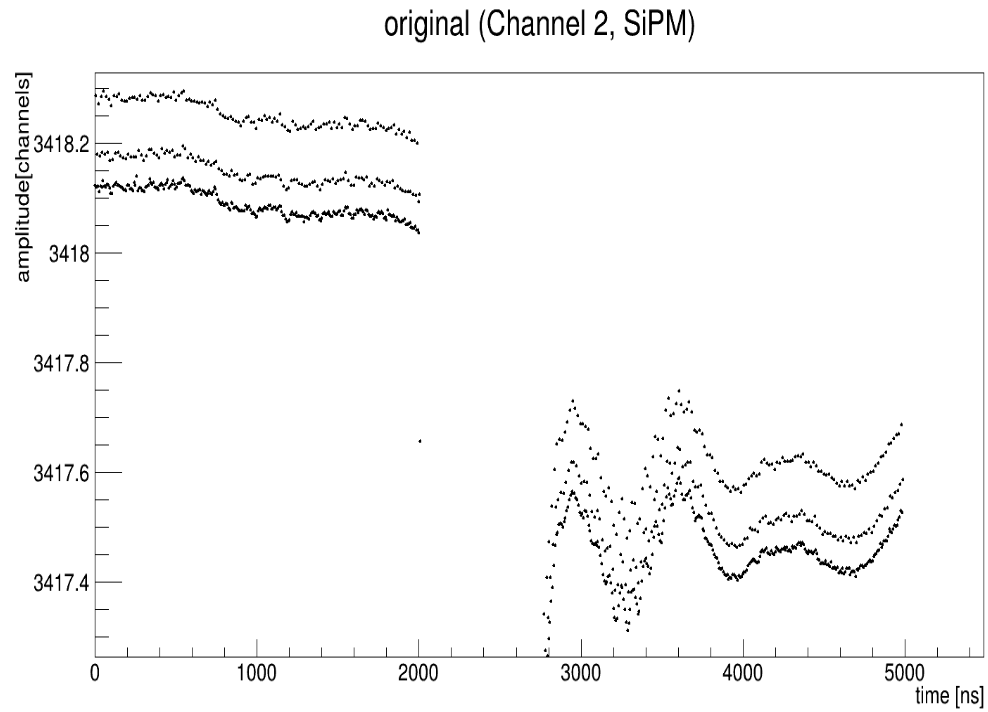


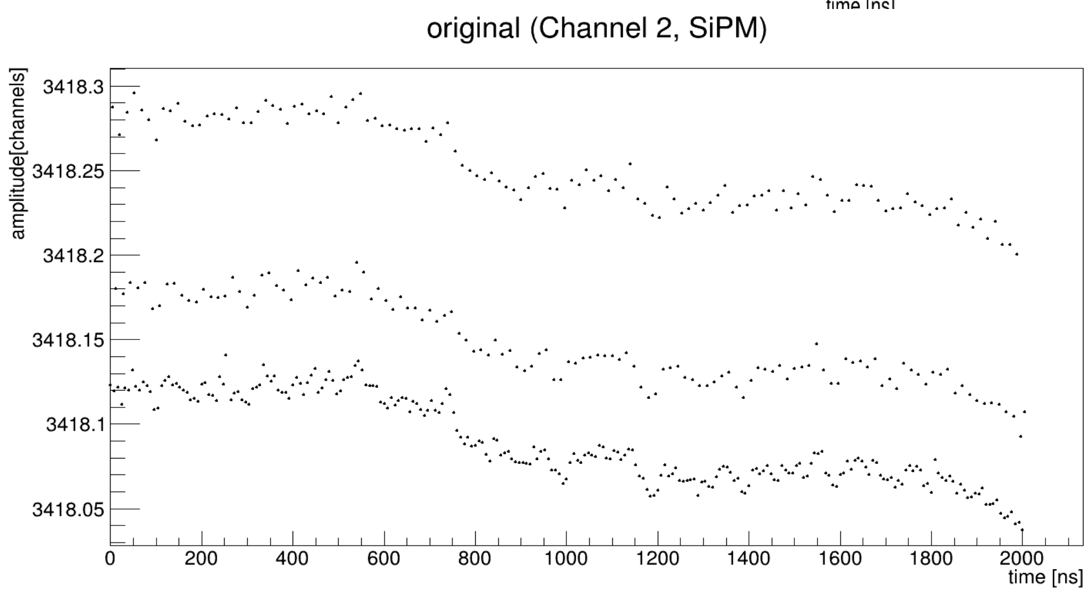


Avr_signal[i] += signal[i] / n_events

Laser run 6061. Averaged signals







Avr_signal[i] += signal[i] / n_events

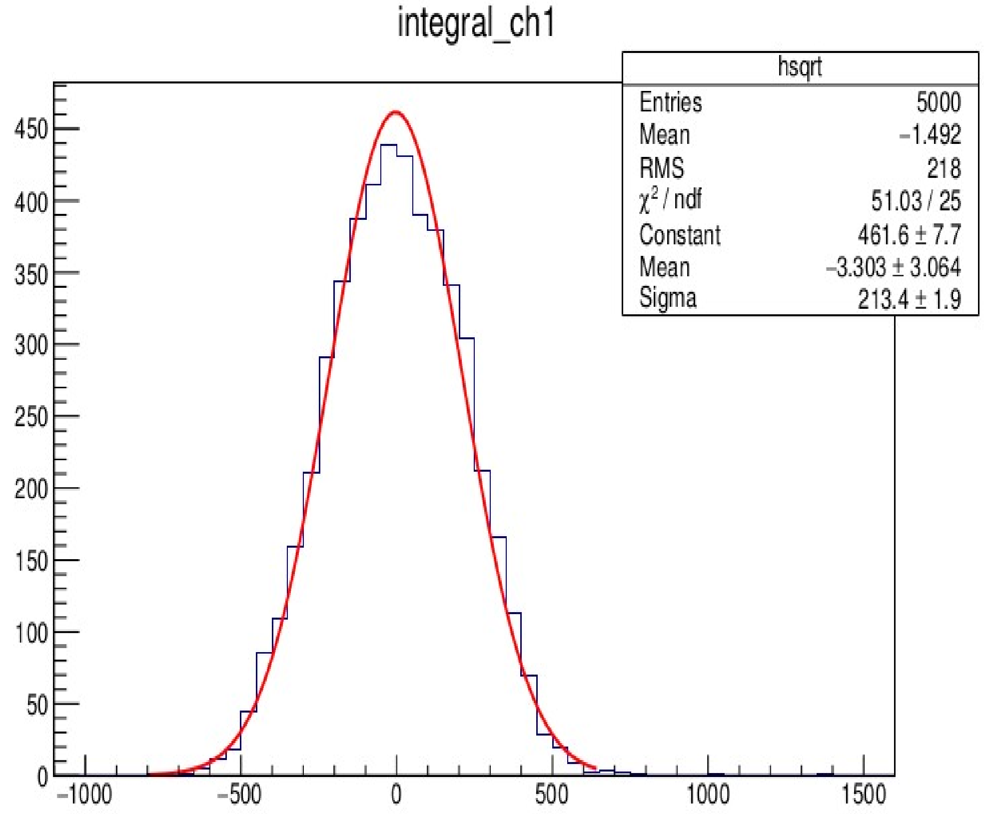
What is reason of such signal?

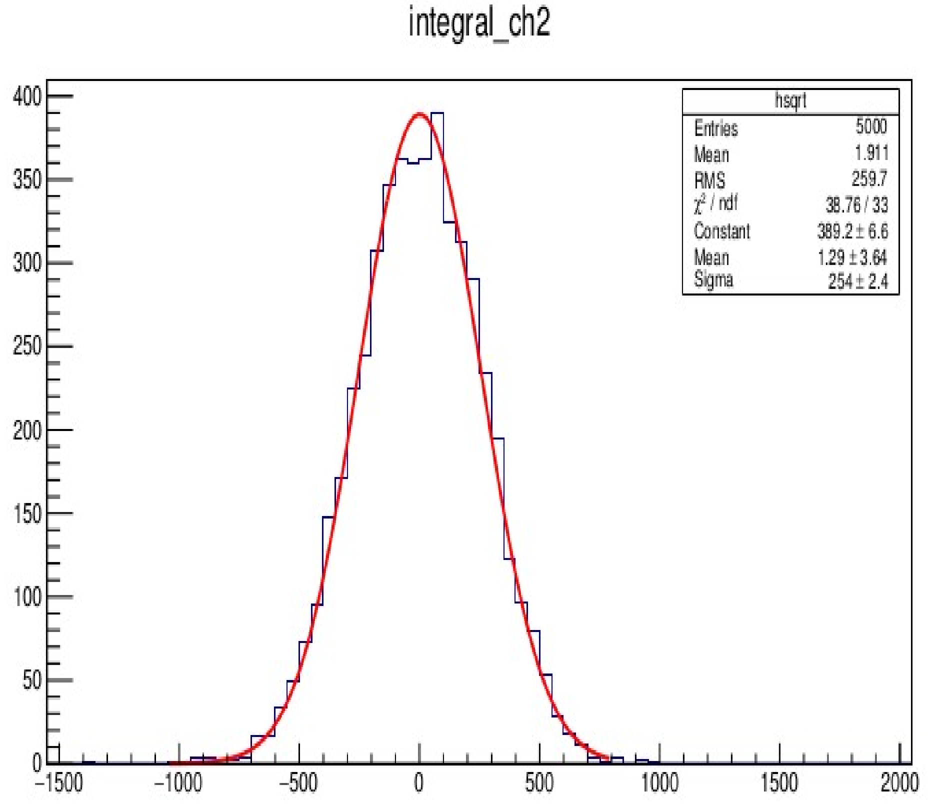
- I expected smooth signal. Even if baseline has fluctuation +-n_channels this signal have to be smooth.

Laser run 6061. Integration spectra

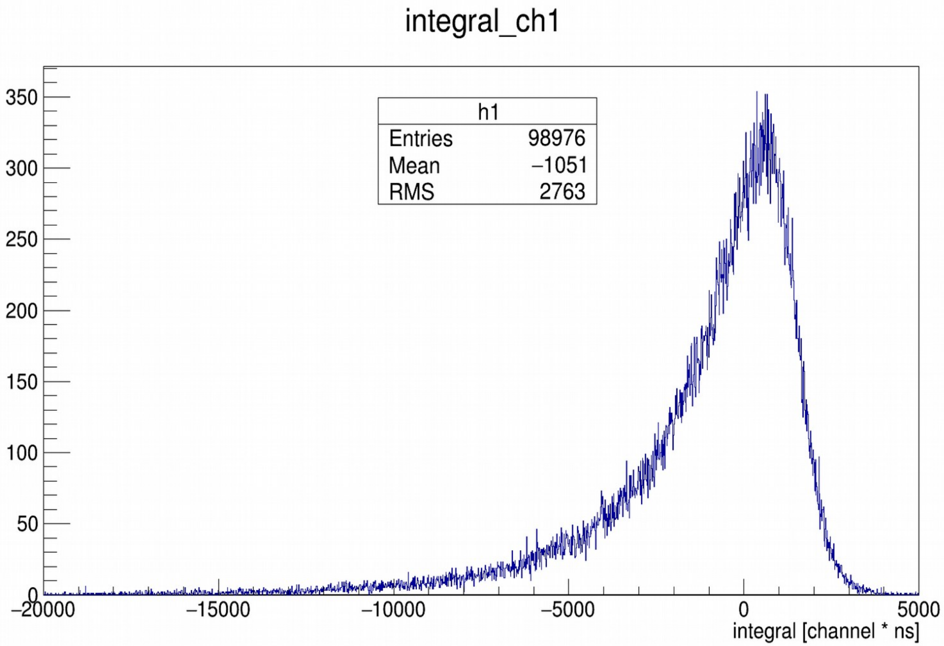
- Integral = sum(y[i] – baseline[i]) * sampling_rate

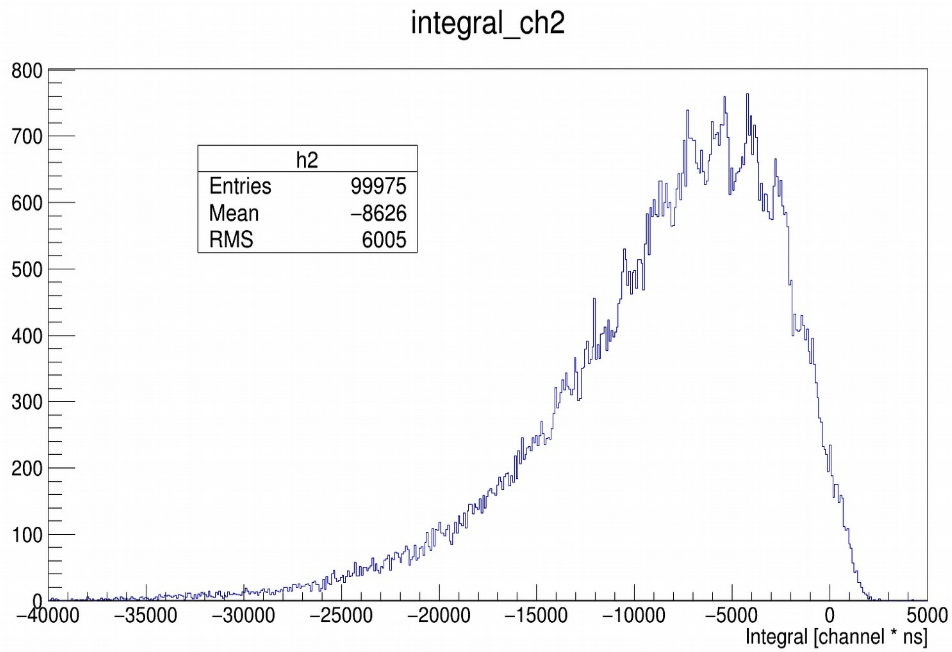
  baseline = avr(y) from 0 to 1600 ns
- Let's calculate integral from 0 to 1600 ns just to check

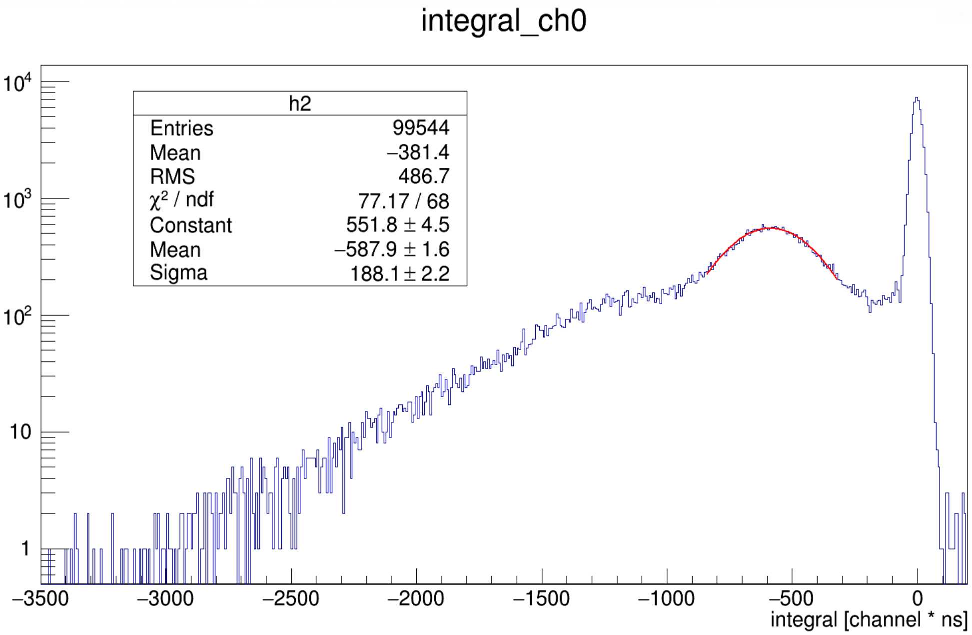




Laser run 6061. Integration spectra







For ch0:
Integral from 2050 to 2100 ns

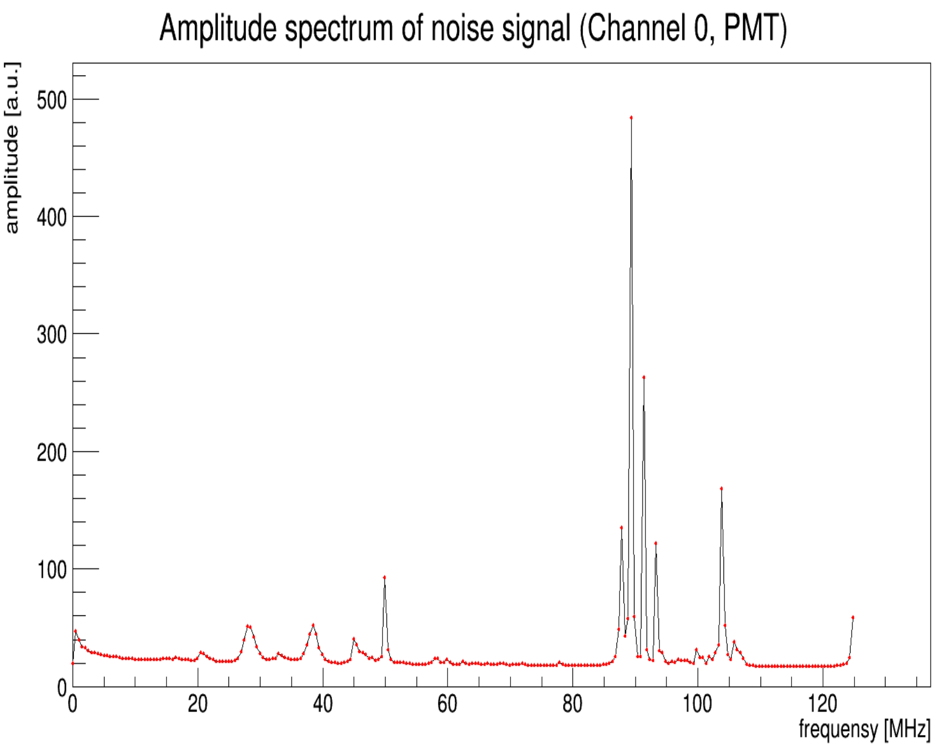
For ch1 and ch2:
Integral from 1950 to 2650 ns

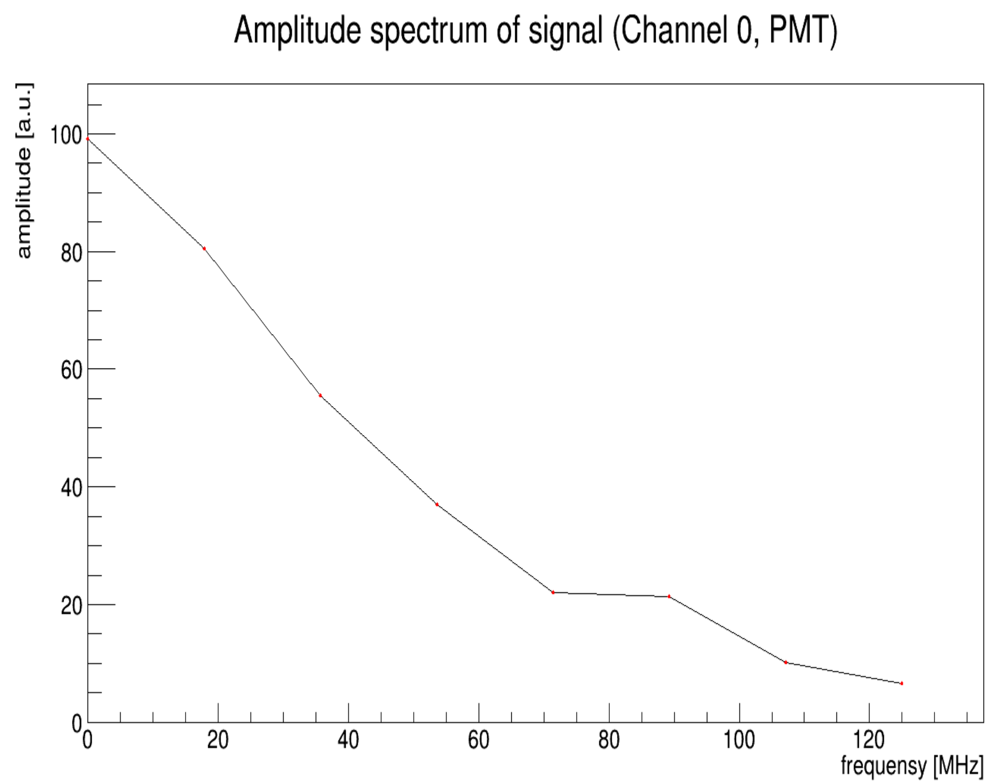
I have not found any peaks for ch1 (for any binning)
Avr spe integral for ch2 is -1484 [channel * ns]

Laser run 6061. Fourier transform

Fourier transform for individual signal is noisy, so let's get avr spectra

Baseline was subtracted before ft.

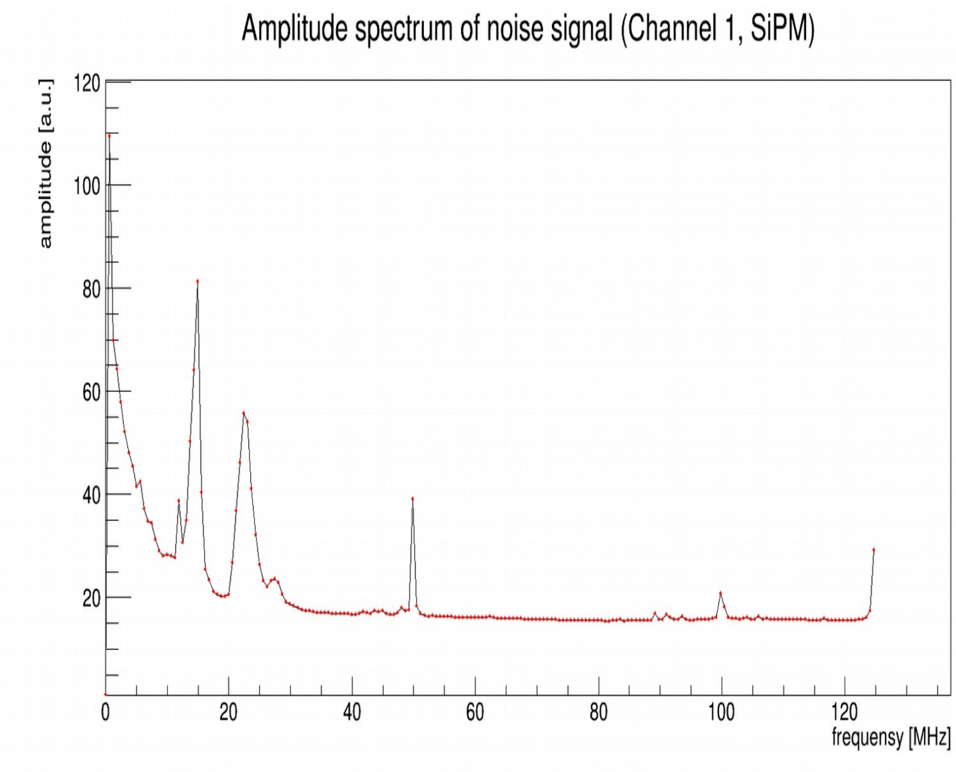


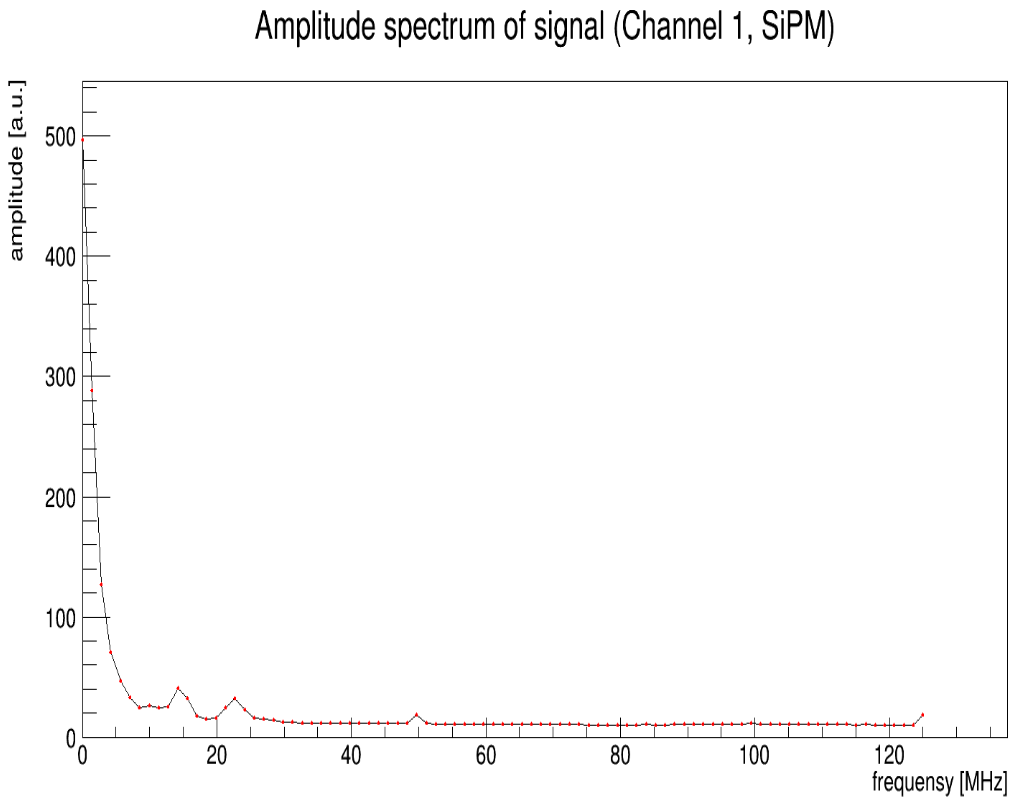


PMT signal is short → amplitude spectrum of signal is wide (ft of delta-func is constant)
PMT signal is short and sampling rate is low →bad frequency resolution

Range for noise from 0 to 2000 ns
Range for signal from 2050 to 2100 ns

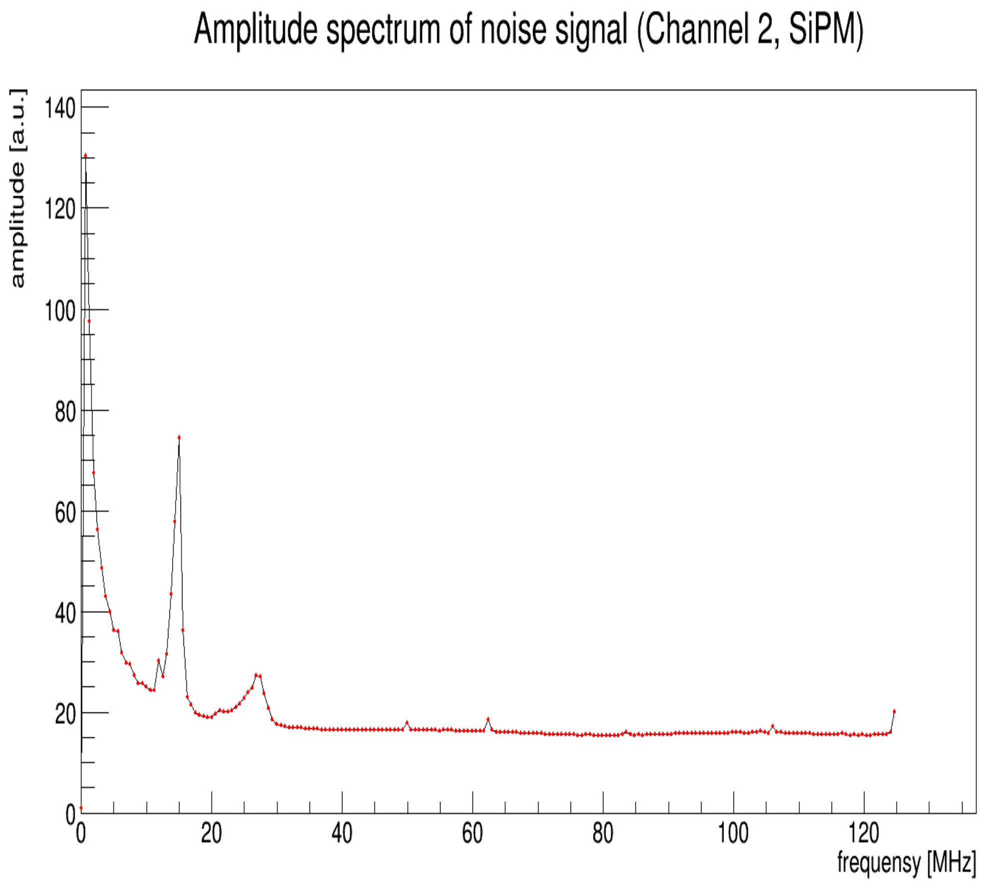
Laser run 6061. Fourier transform

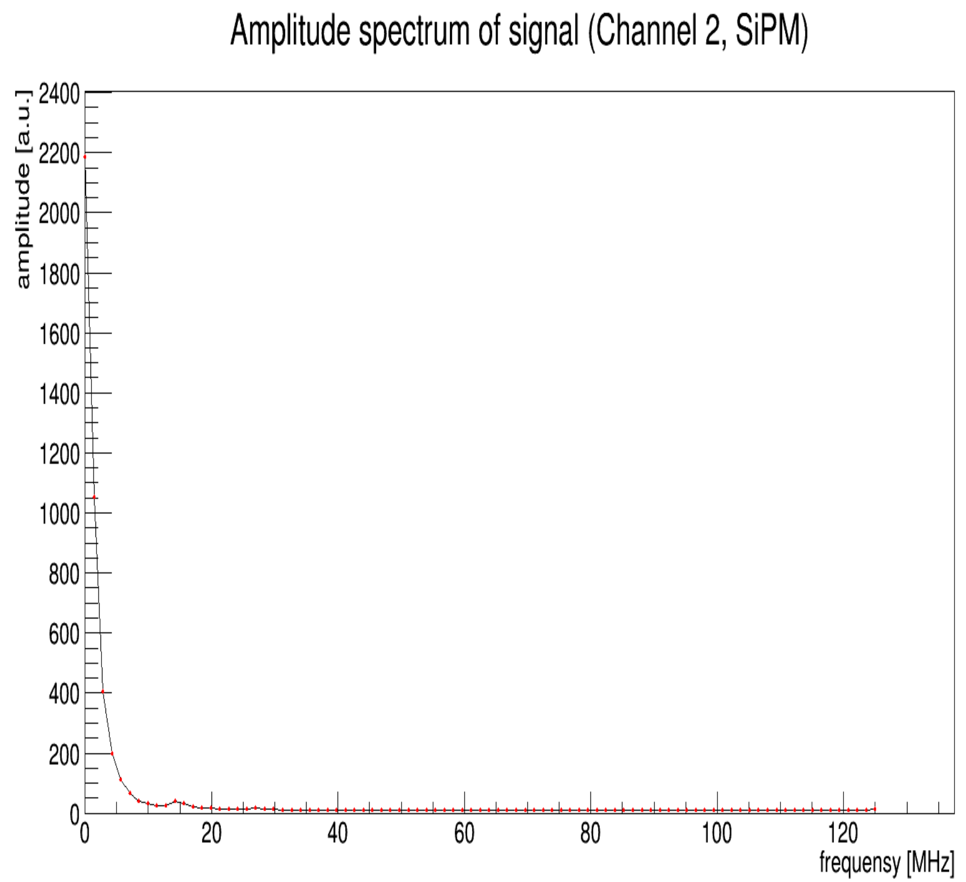




Range for noise from 0 to 1600 ns
Range for signal from 1950 to 2650 ns

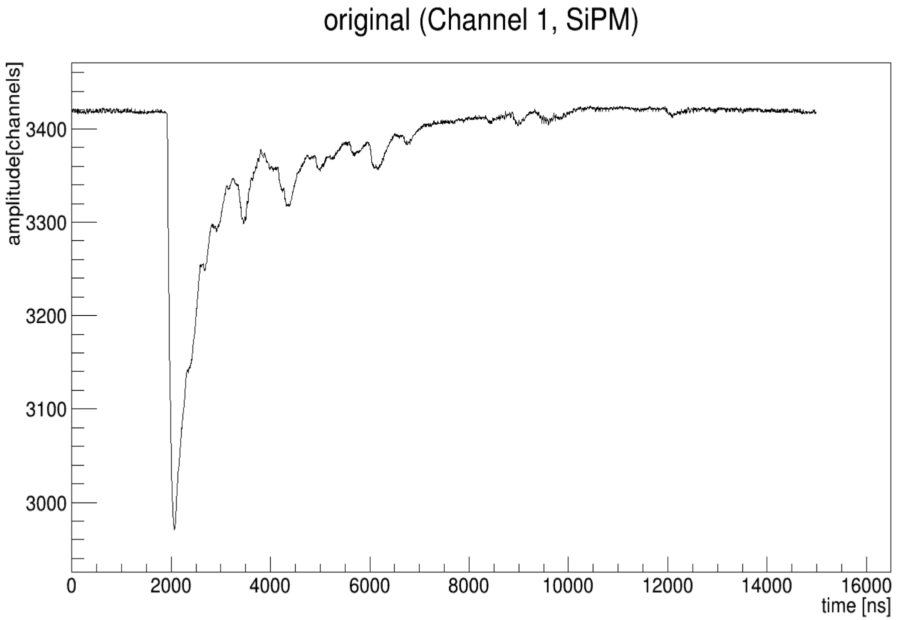
Laser run 6061. Fourier transform

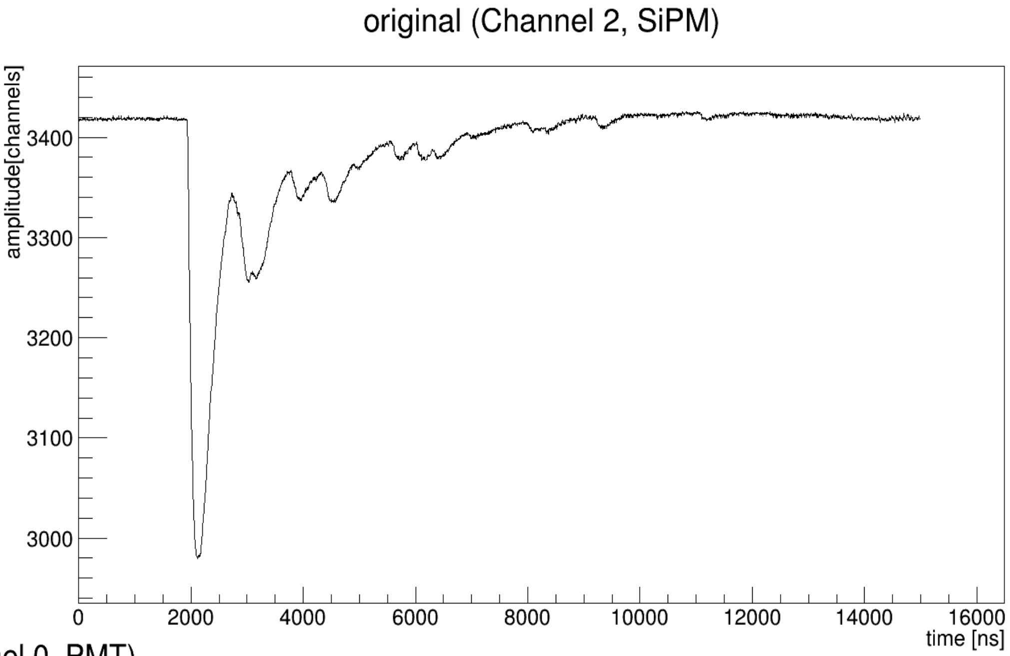


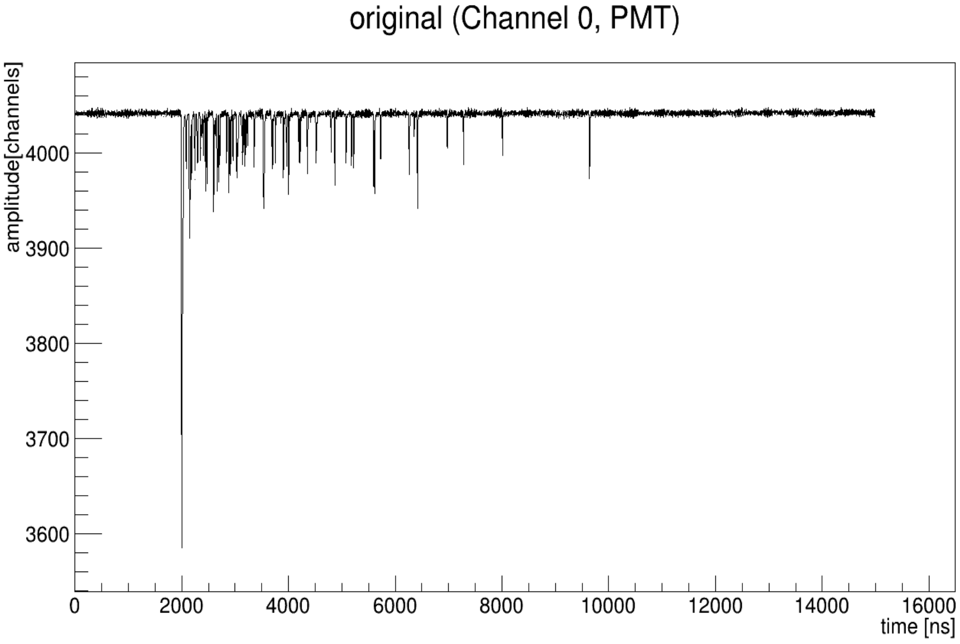


Range for noise from 0 to 1600 ns
Range for signal from 1950 to 2650 ns

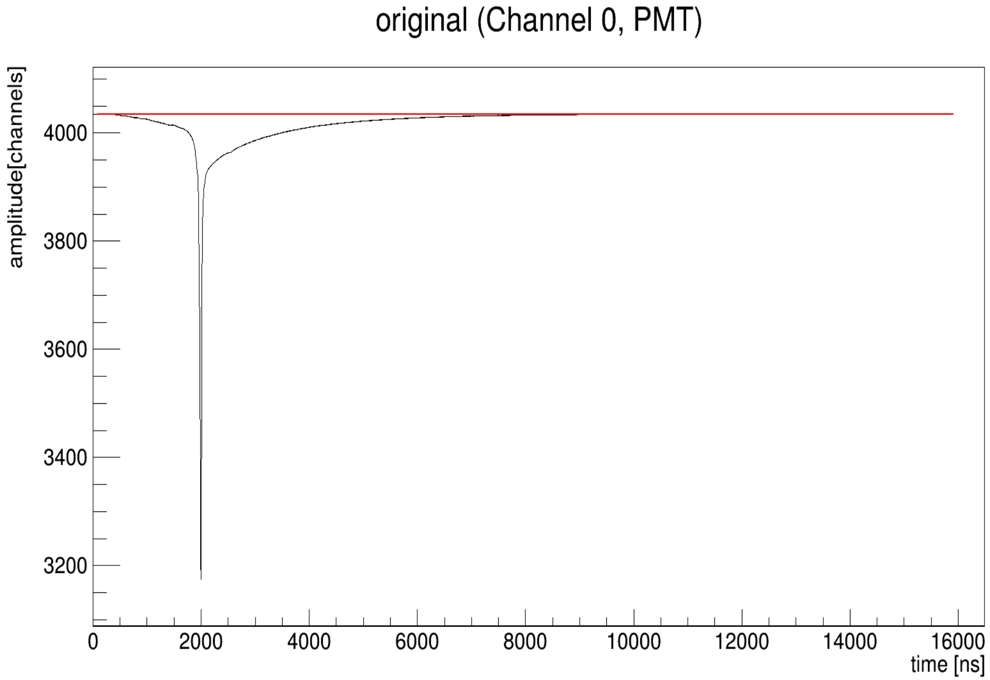
Am run 6064. Typical signals

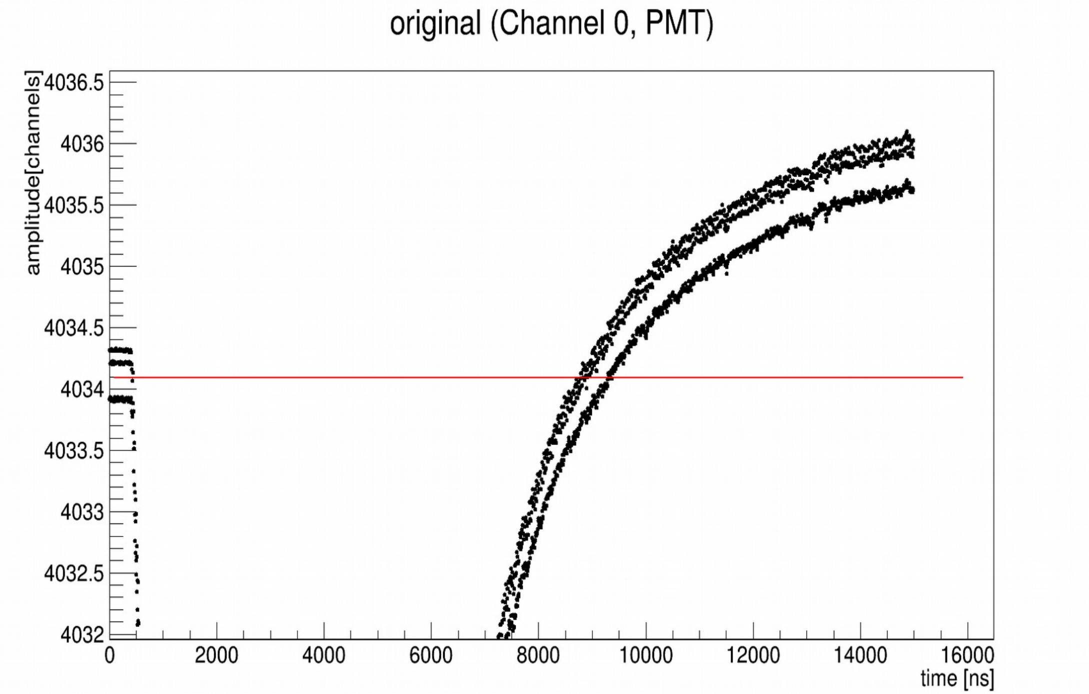


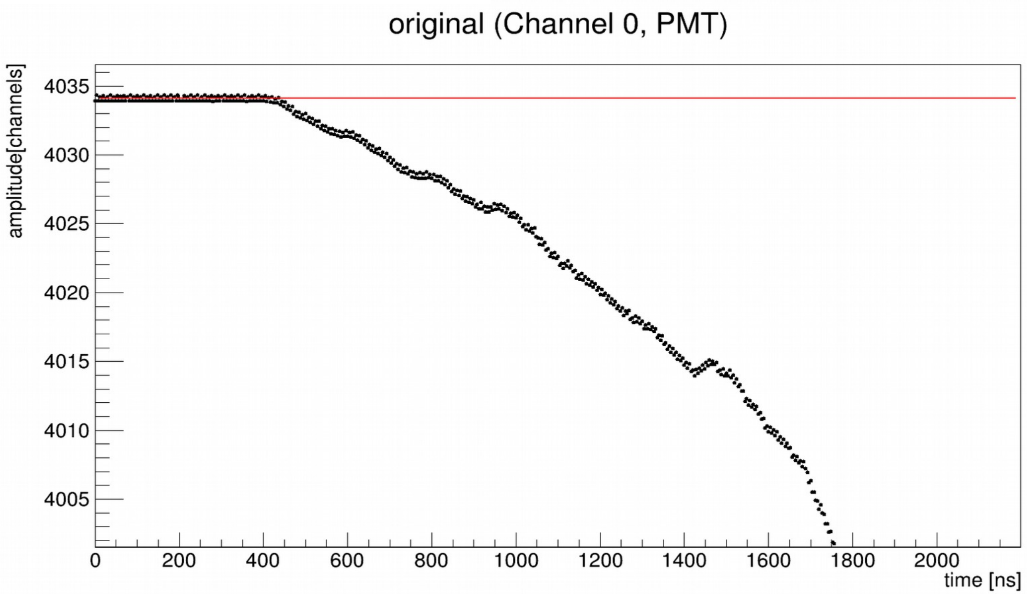




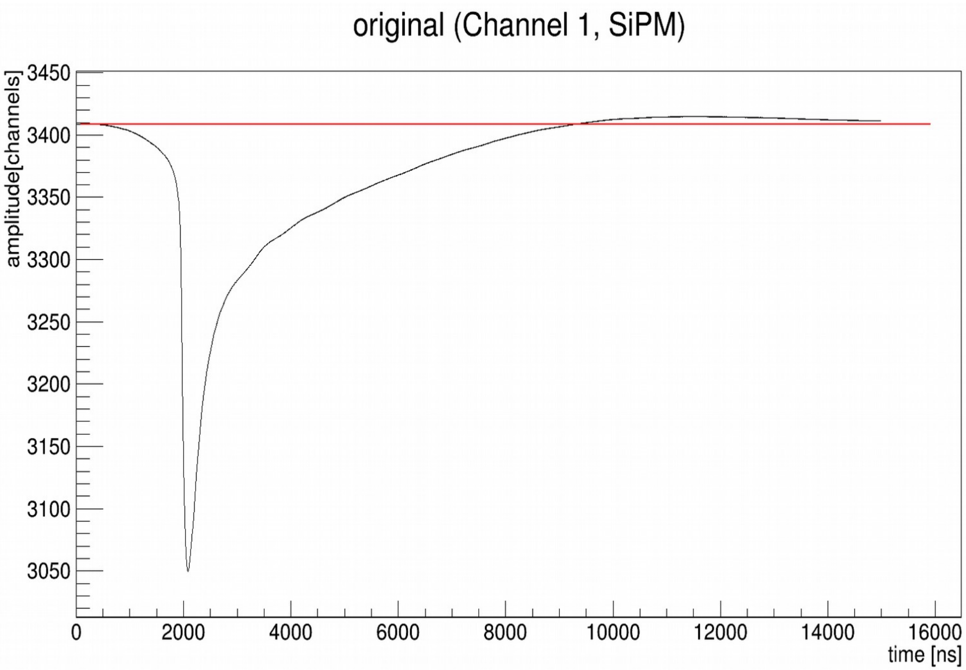
Am run 6064. Averaged signals

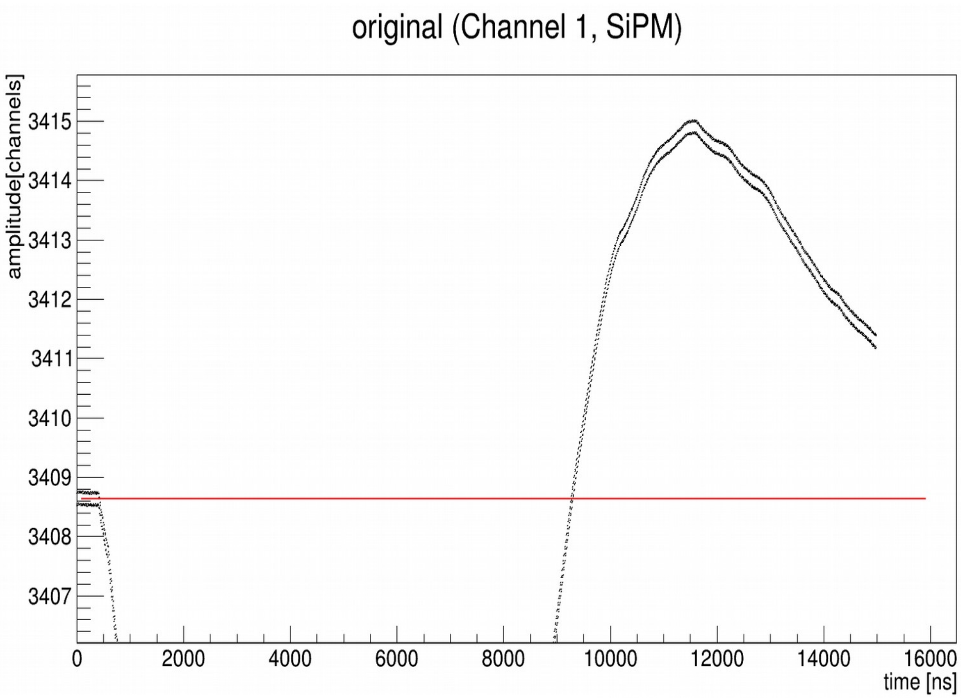


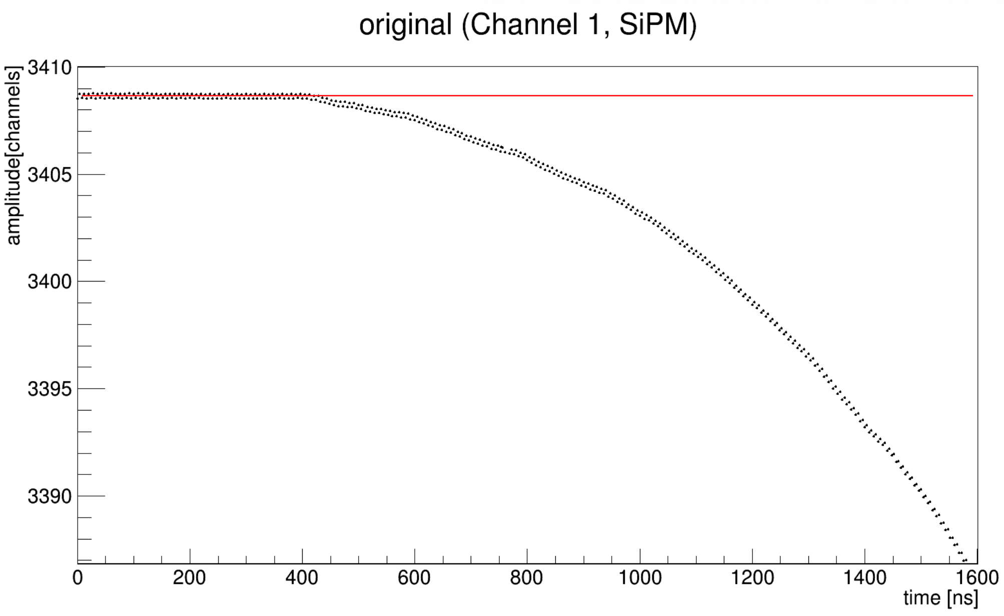




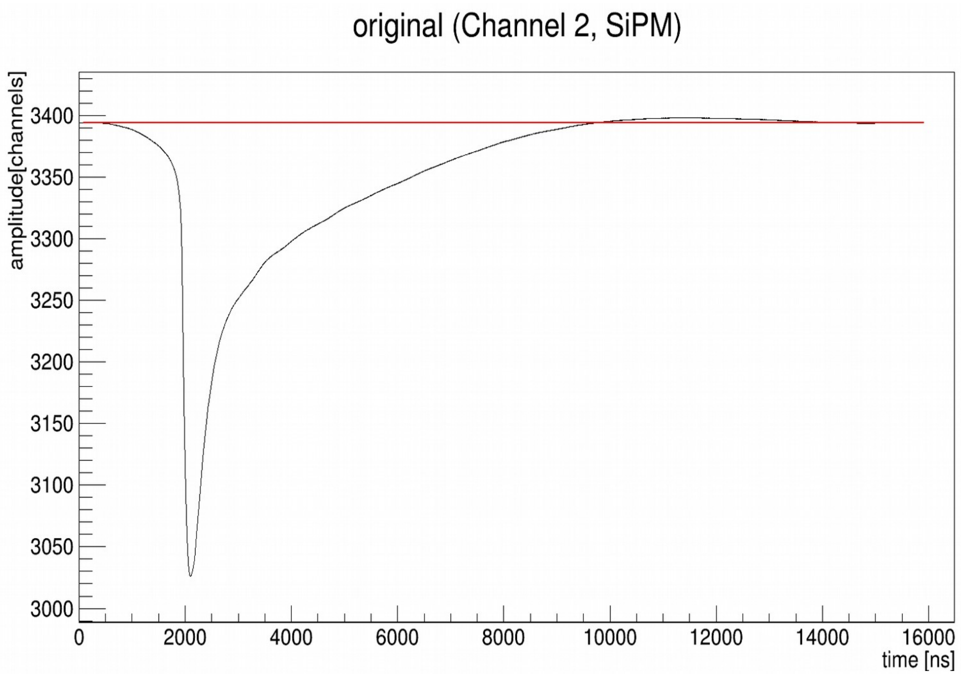
Am run 6064. Averaged signals

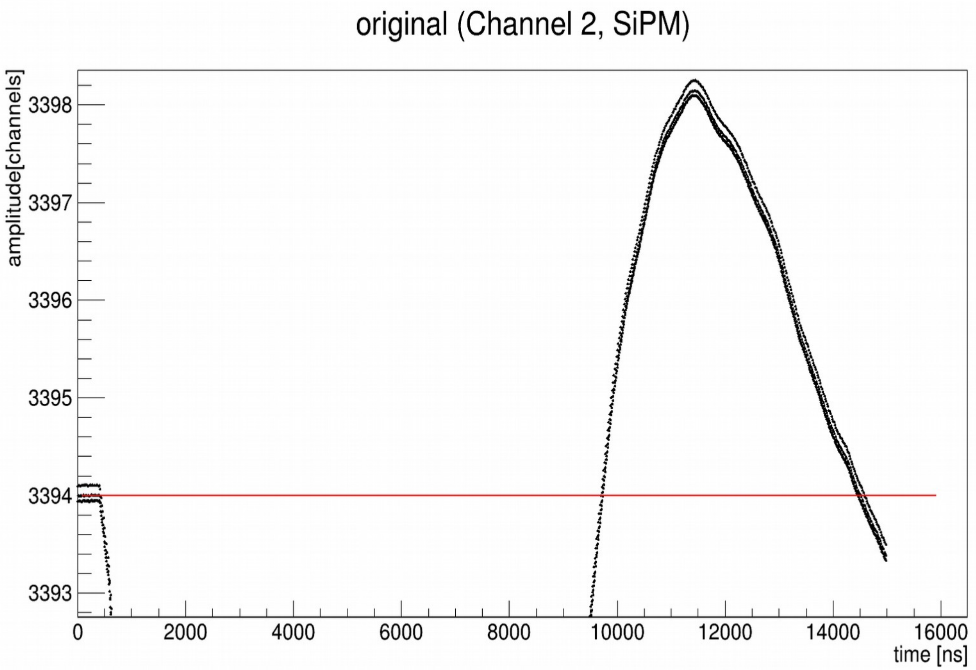


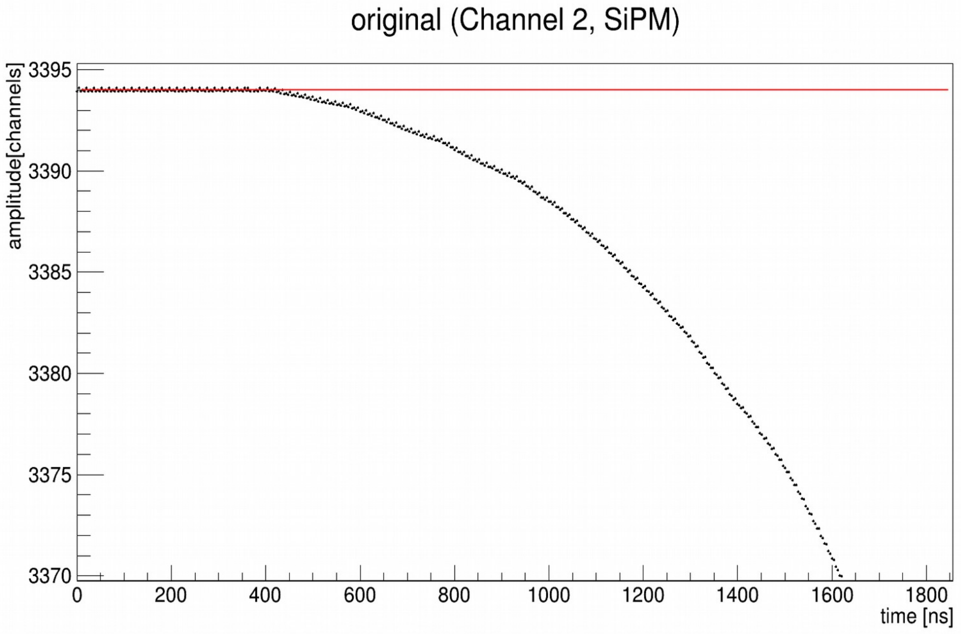




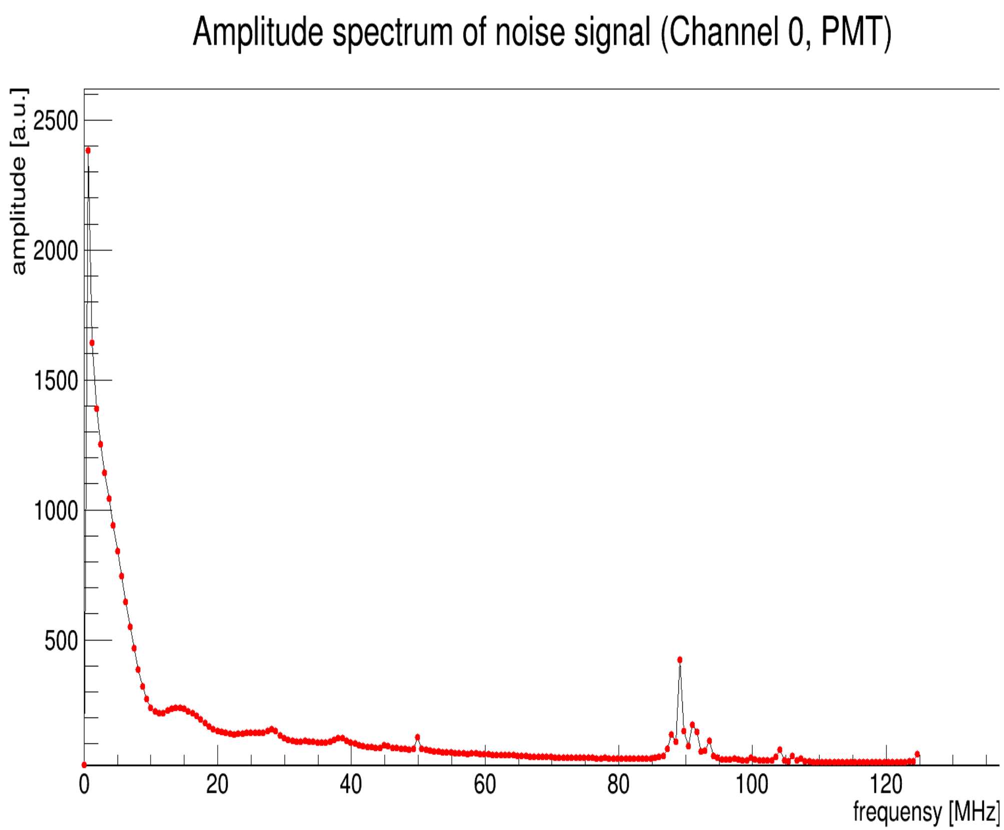
Am run 6064. Averaged signals

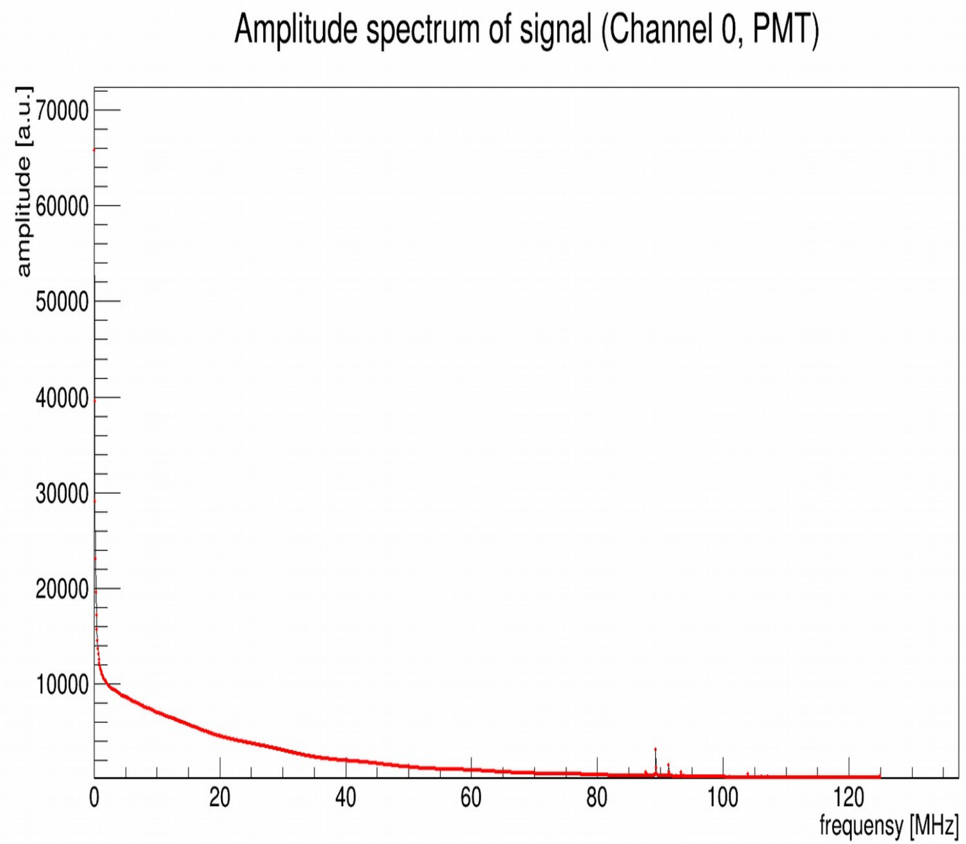






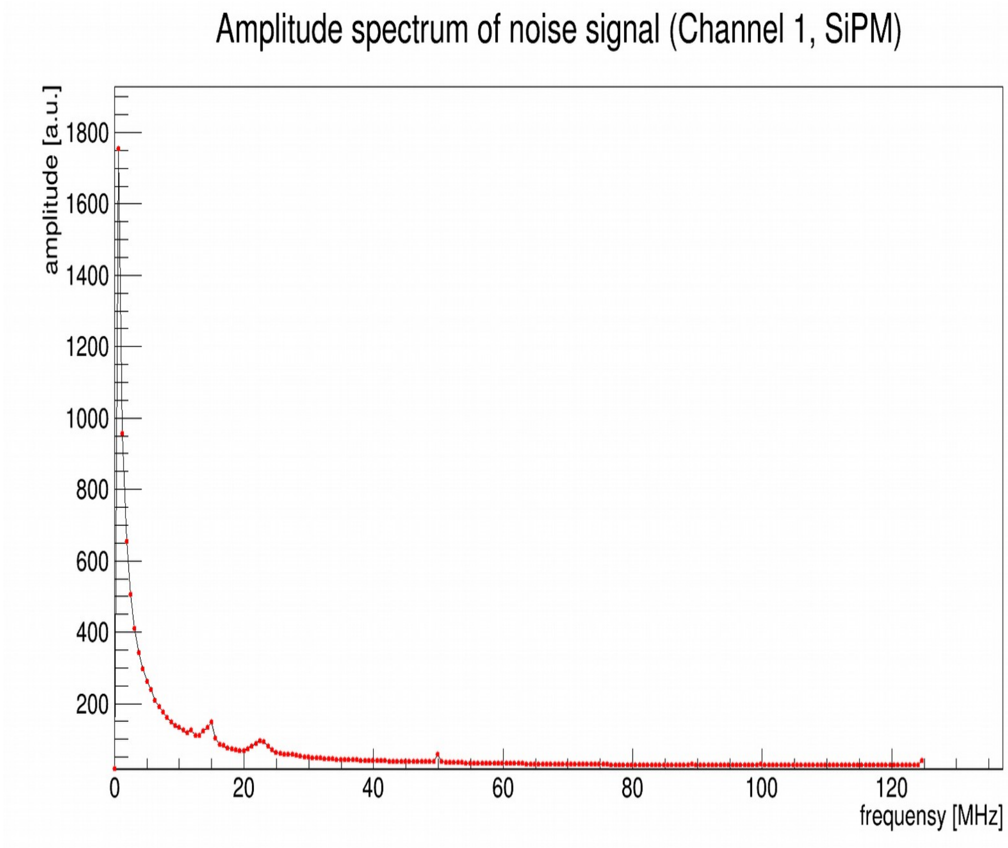
Am run 6064. Fourier transform

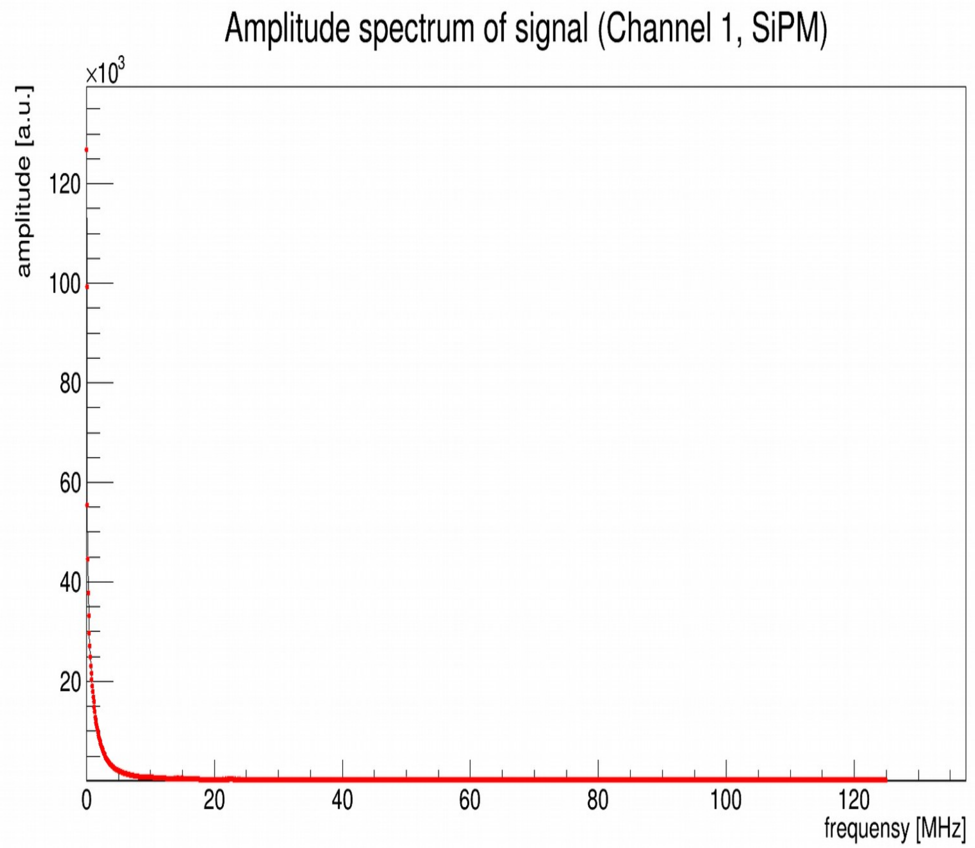




Range for noise from 0 to 1600 ns
Range for signal from 1900 to 14900 ns

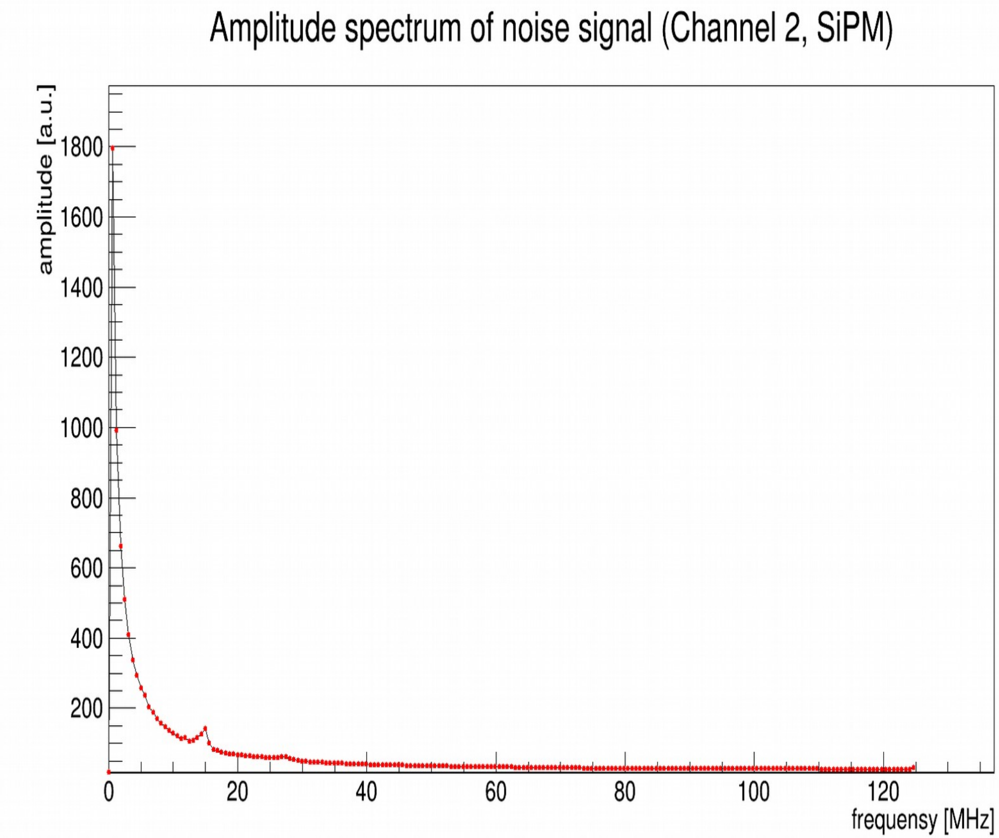
Am run 6064. Fourier transform

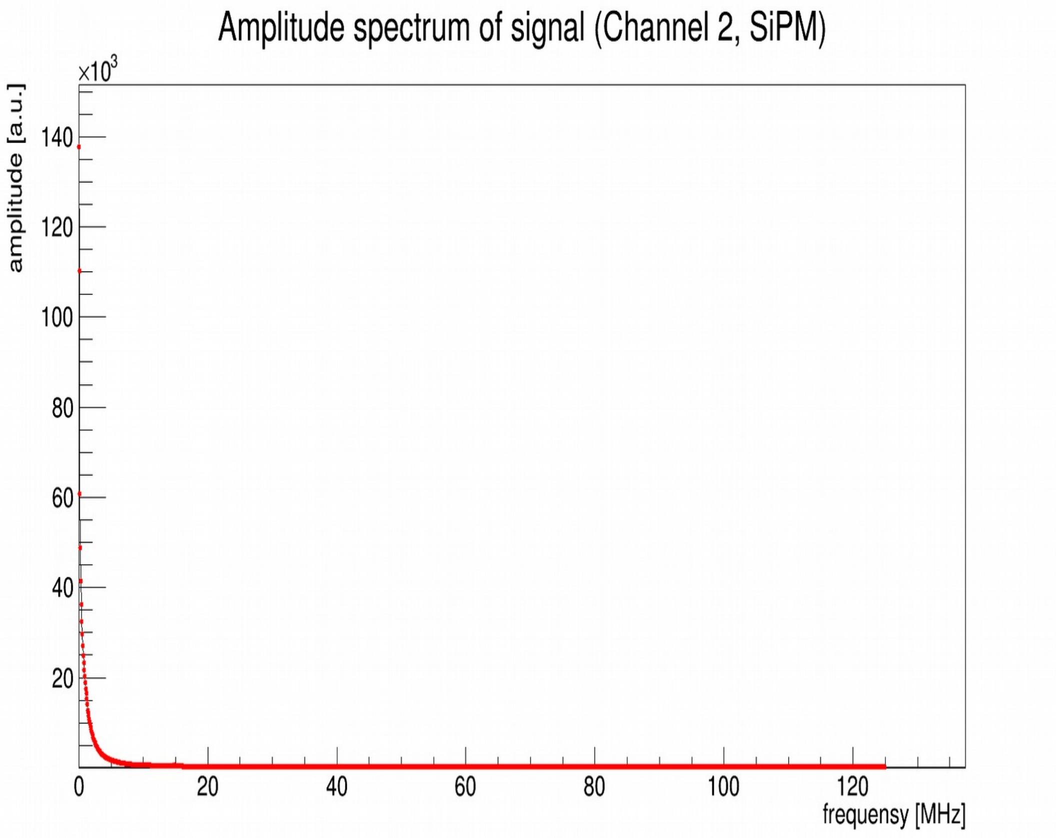




Range for noise from 0 to 1600 ns
Range for signal from 1900 to 14900 ns

Am run 6064. Fourier transform



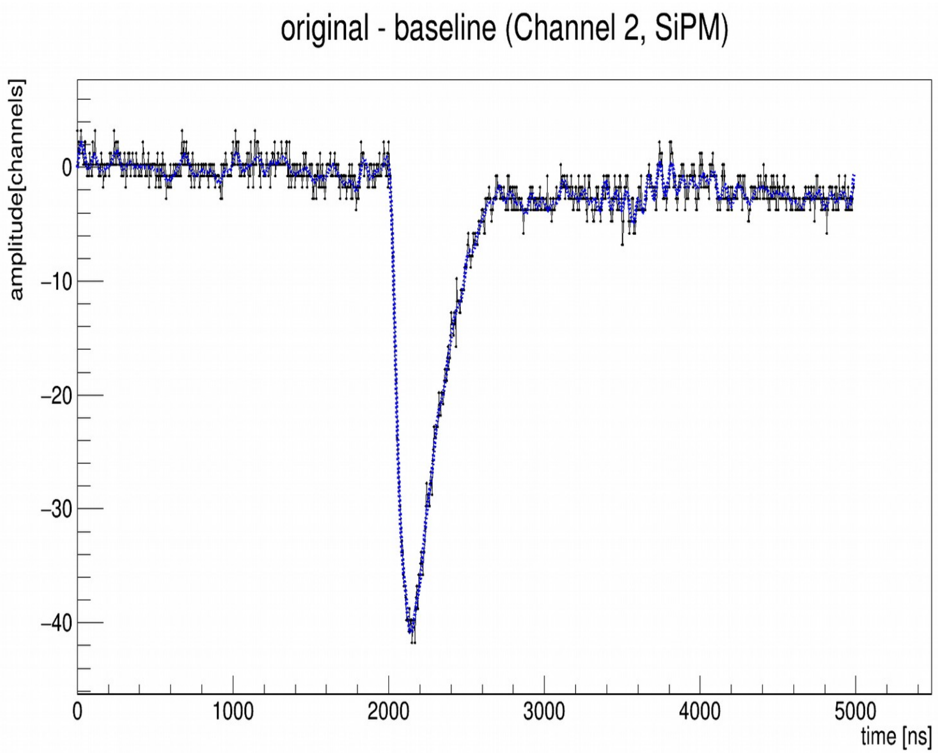


Range for noise from 0 to 1600 ns
Range for signal from 1900 to 14900 ns

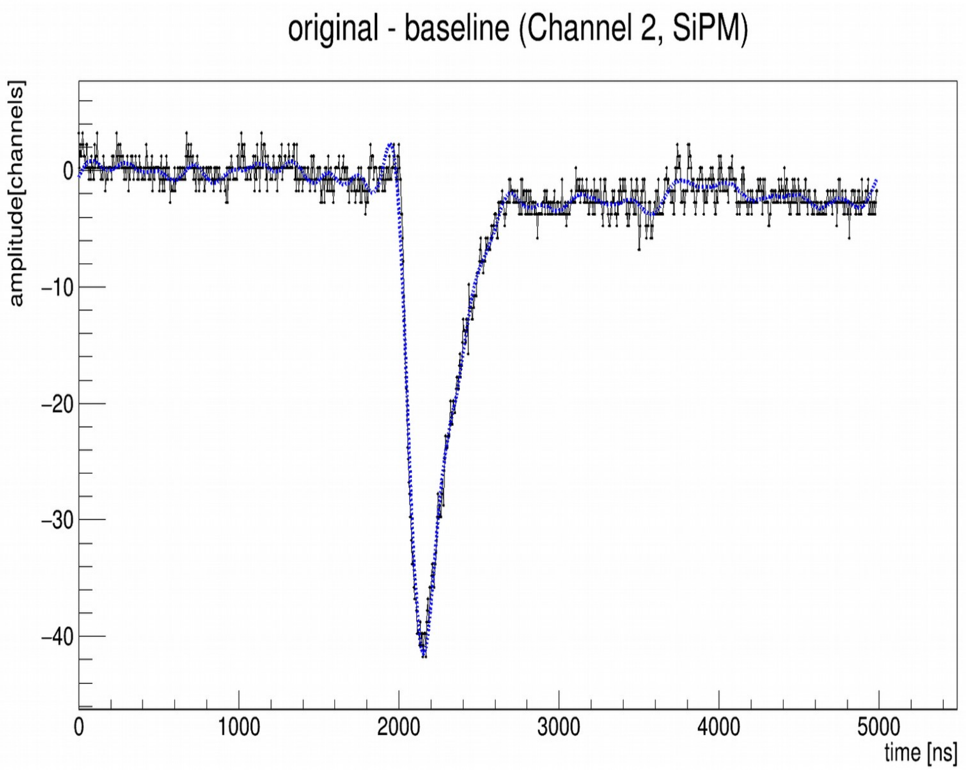
Inverse Fourier transform

- Can we cut high frequencies (noise) and save signal?
  Yes, but there are no serious advantages, because amplitude spectra of noise and signal are overlapped significantly.

15 MHz cut

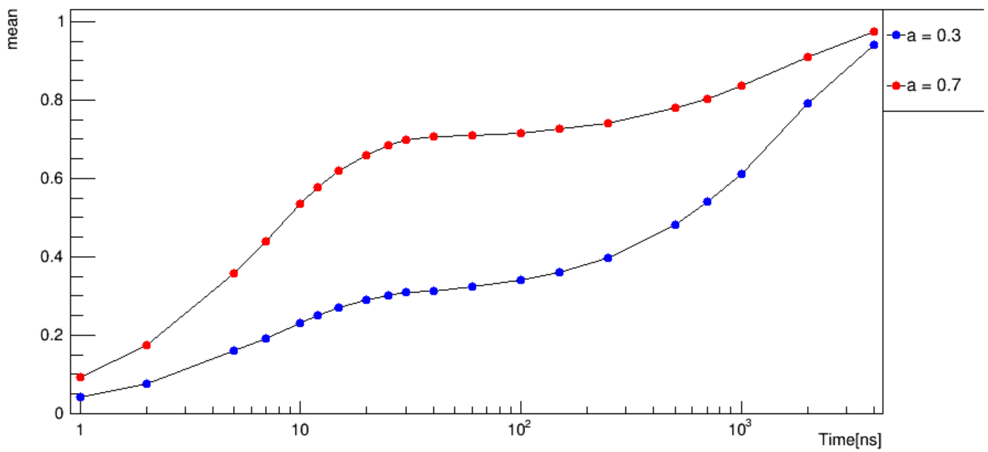


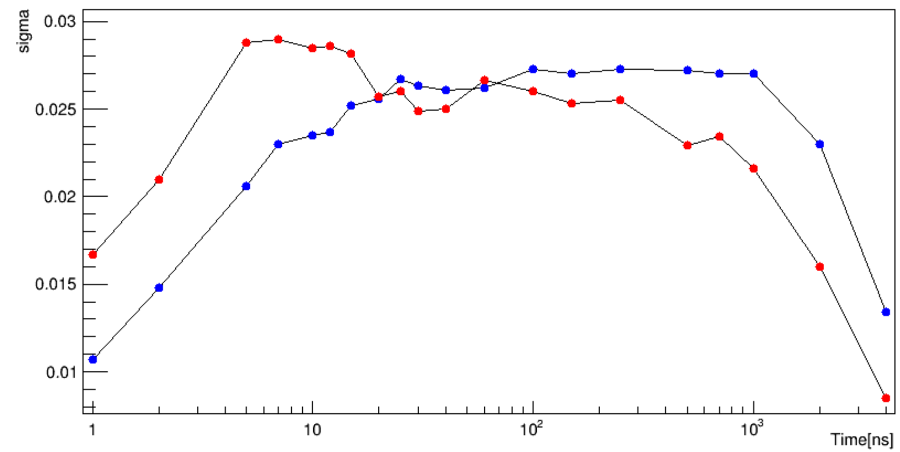
5 MHz cut

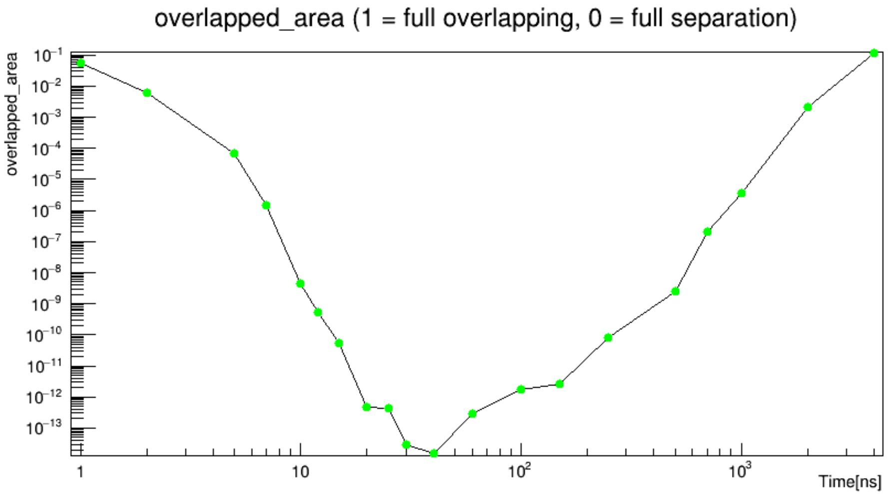


Optimum in theory

Consider the case with delta-function response. N_pe = 300





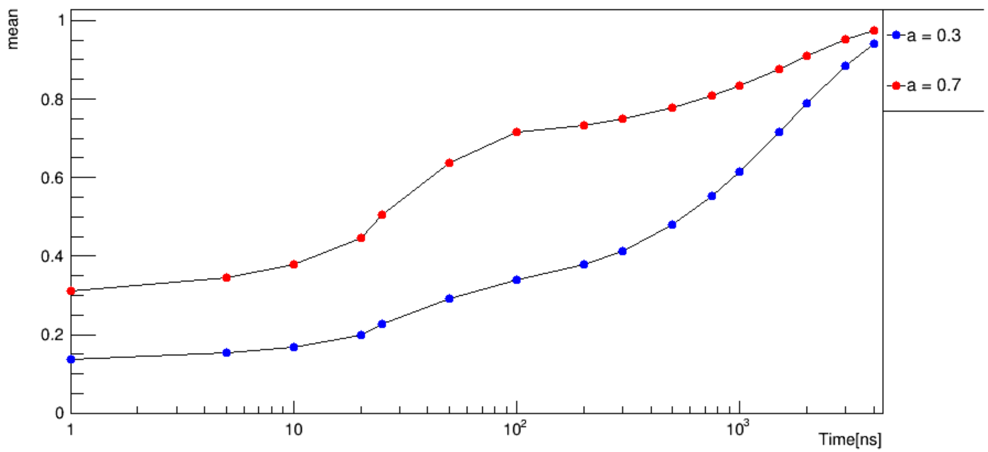


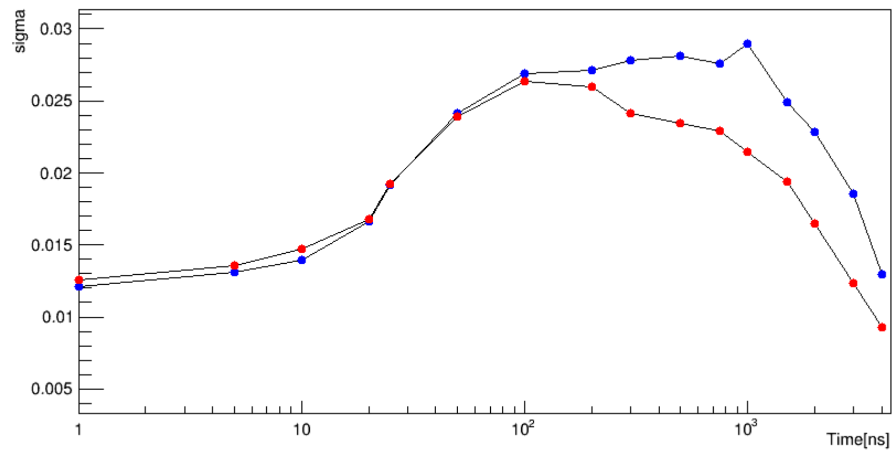
PDF = A * PDF_fast + (1 – A) * PDF_slow
A = intensity_fast / (intensity_fast + intensity_slow)
A = 0.2308 for e / gamma
A = 0.75 for neutrons

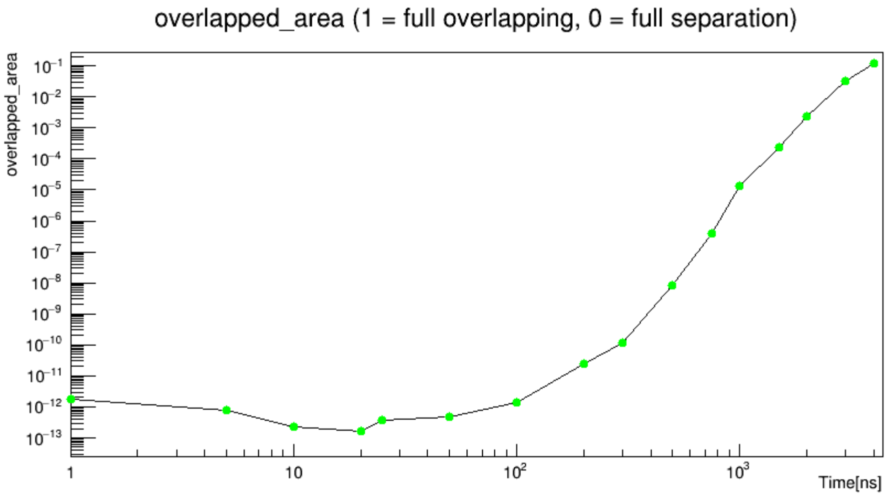
*this pictures for A=0.3 and A=0.7, because I chose wrong parameters.
But I think, that for right parameters the result will be very close to this.

Optimum in theory

Consider the case with gauss response (sigma = 33ns). N_pe = 300







PDF = A * PDF_fast + (1 – A) * PDF_slow
A = intensity_fast / (intensity_fast + intensity_slow)
A = 0.2308 for e / gamma
A = 0.75 for neutrons

*this pictures for A=0.3 and A=0.7, because I chose wrong parameters.
But I think, that for right parameters the result will be very close to this.

Signal reconstruction

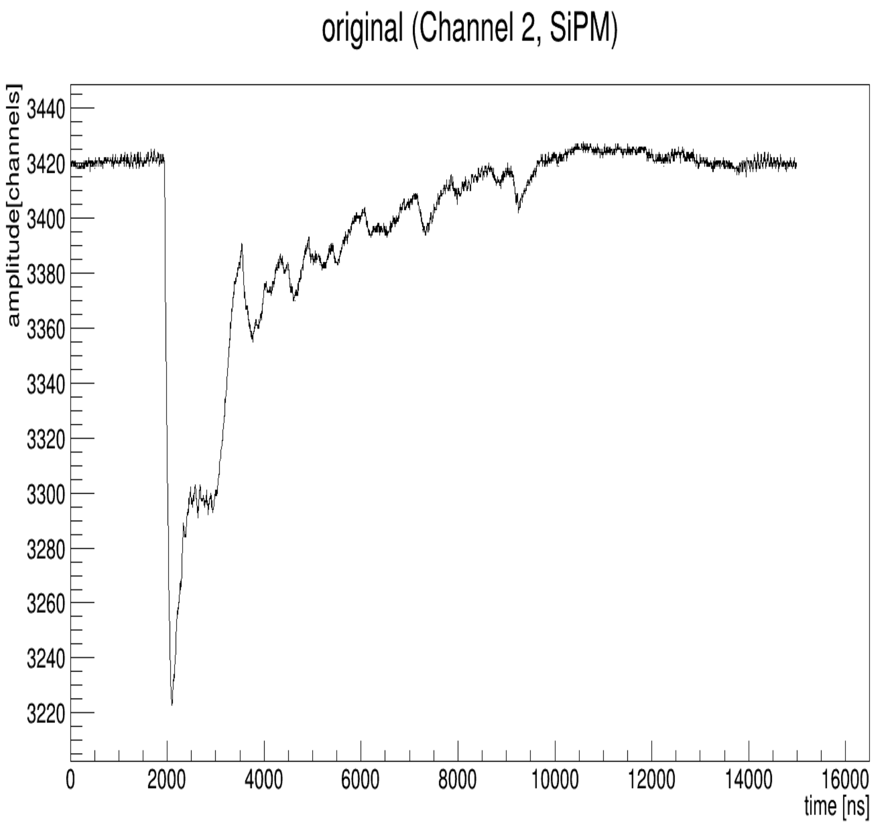
We want to find the arrival time for individual photons. This information will help to improve neutron / gamma separation. There are several methods.

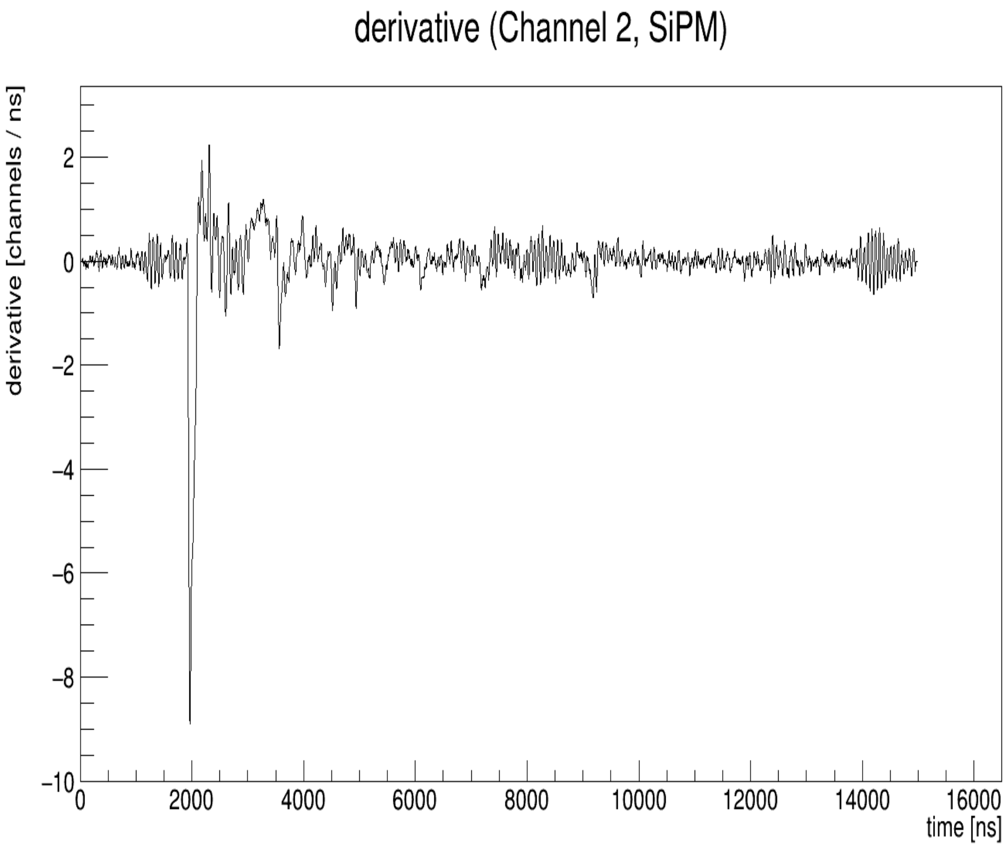
- Find derivative and identify peaks
- Approximation by sum of individual responses
- Deconvolution

Find derivative and identify peaks

"+" The simplest method

"-" Works incorrect for close signals
Overlapping of negative part of the 1st signal with positive part of the 2nd (for derivative)





Approximation by sum of individual responses

“+”
There are a lot of parameters, which we can change to get the best results for arrival time.
Can separate close peaks.

“-”
High CPU consumption (especially if there are a lot of signals).
It is difficult to find initial parameters for approximation.
A lot of manual work.
Each case is individual and require proper initialization for initial parameters.

Deconvolution

“+”
A variety of methods. Can reconstructs signal close to delta function (narrow peaks).

“-”
High CPU consumption (but independent from number of signals).
There are artifacts of deconvolution.
We should understand how it works in presence of noise.
Some algorithm only for positive data.

Deconvolution algorithms

- Strictly speaking, deconvolution is to solve overdetermined linear system. This topic requires a separate presentation

Some realized algorithms

- Root CERN TUnfold
- Gold deconvolution (Root CERN TSpectrum::Deconvolution)
- Richardson–Lucy deconvolution (Root CERN TSpectrum::DeconvolutionRL)

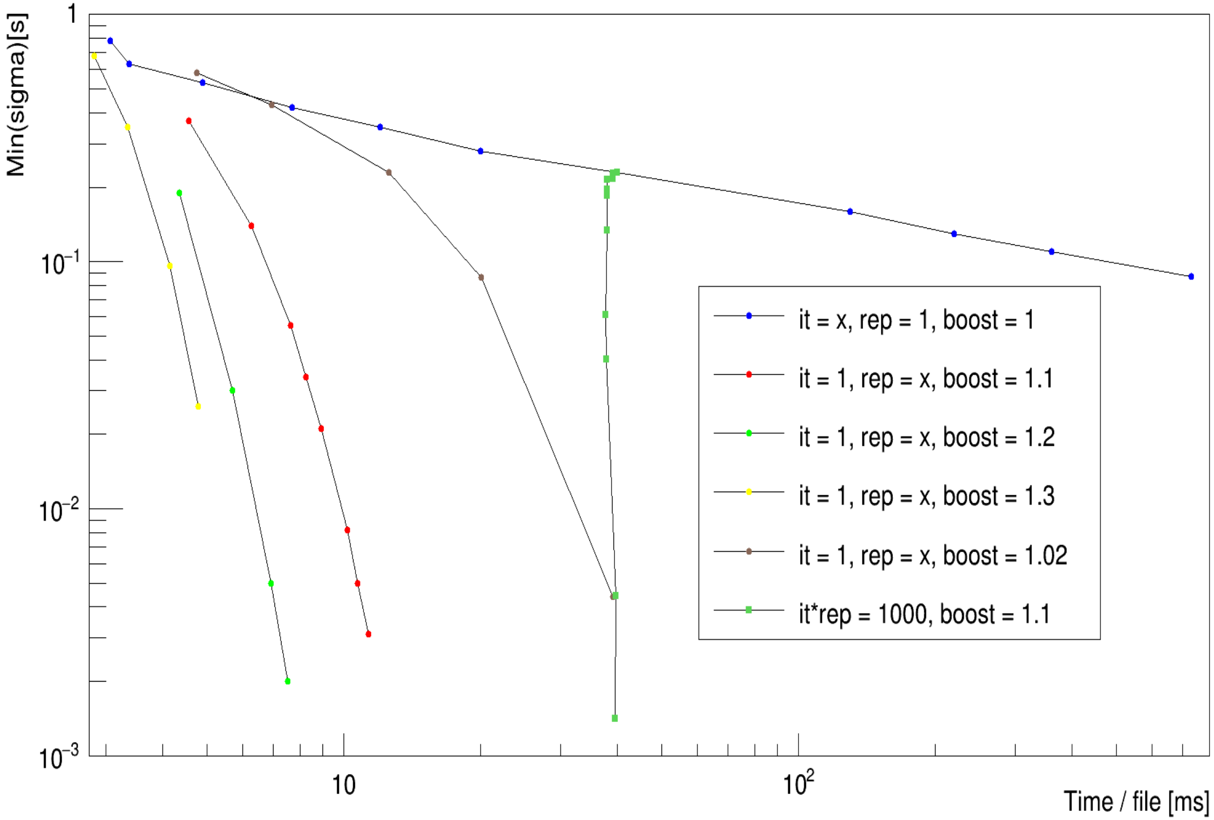
Root CERN TSpectrum::Deconvolution

3 parameters: N_iter, N_rep, boost

1) N_iter >1, N_rep = 1, boost = 1
Works correct, but slowly

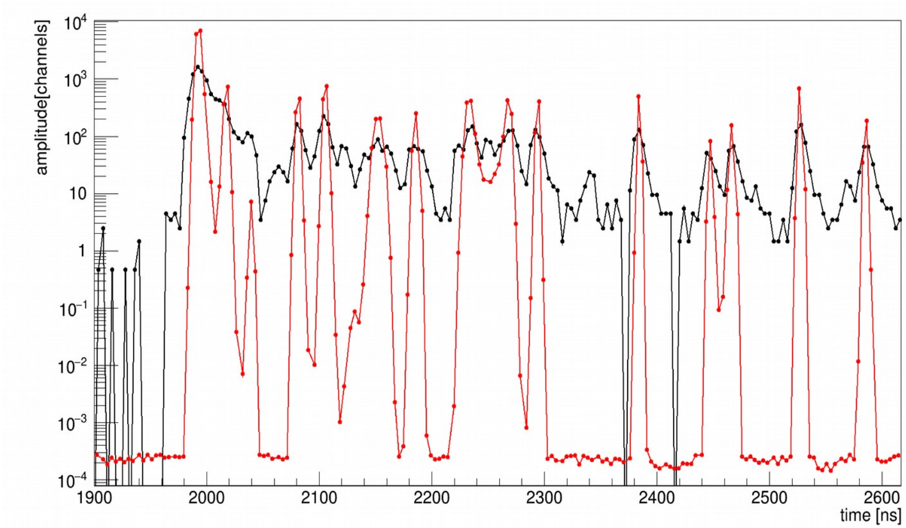
2) N_iter >1, N_rep > 1, boost >1
Distorts small signals, but can be very fast

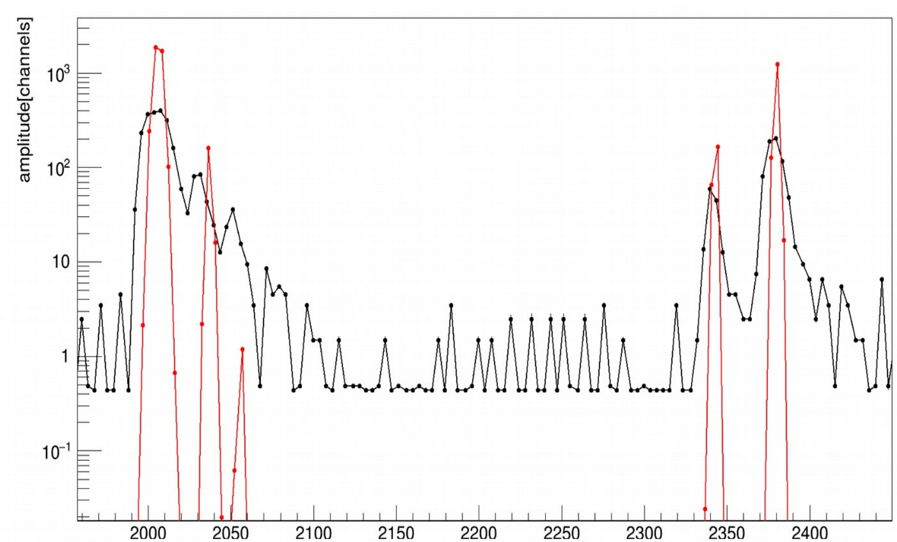
Simulation (in order to measure width correct)

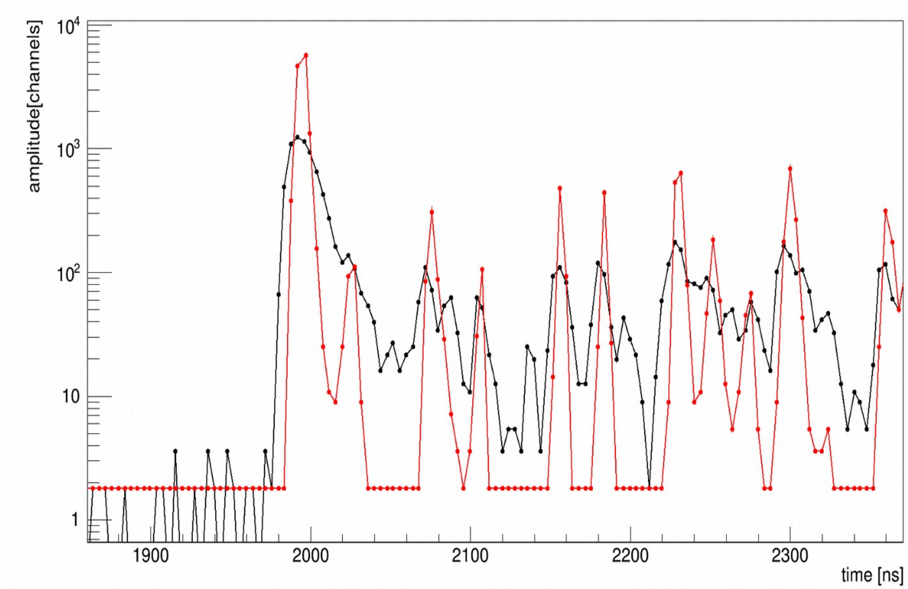


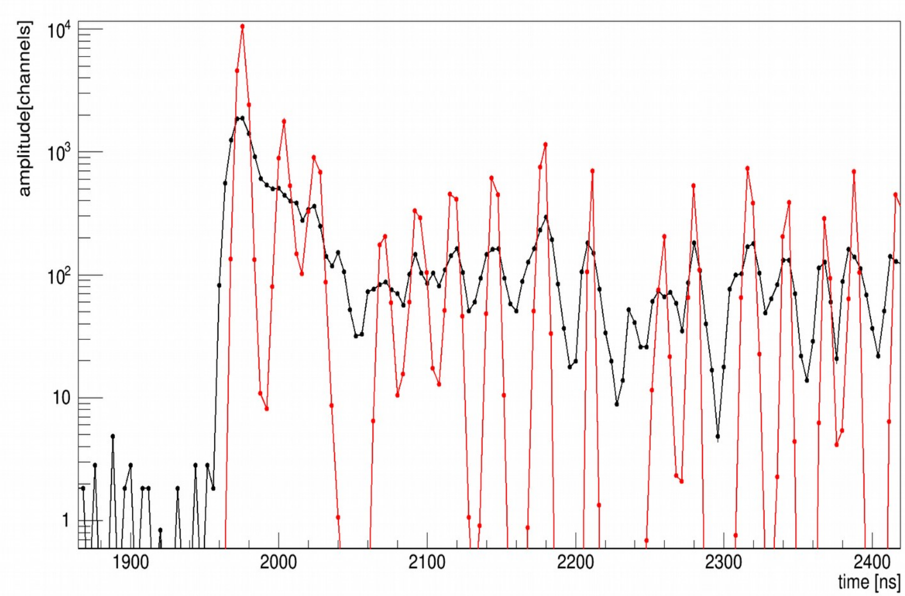
Tspectrum::Deconvolution ch0 results

N_iter = 100, N_rep = 1, boost = 1



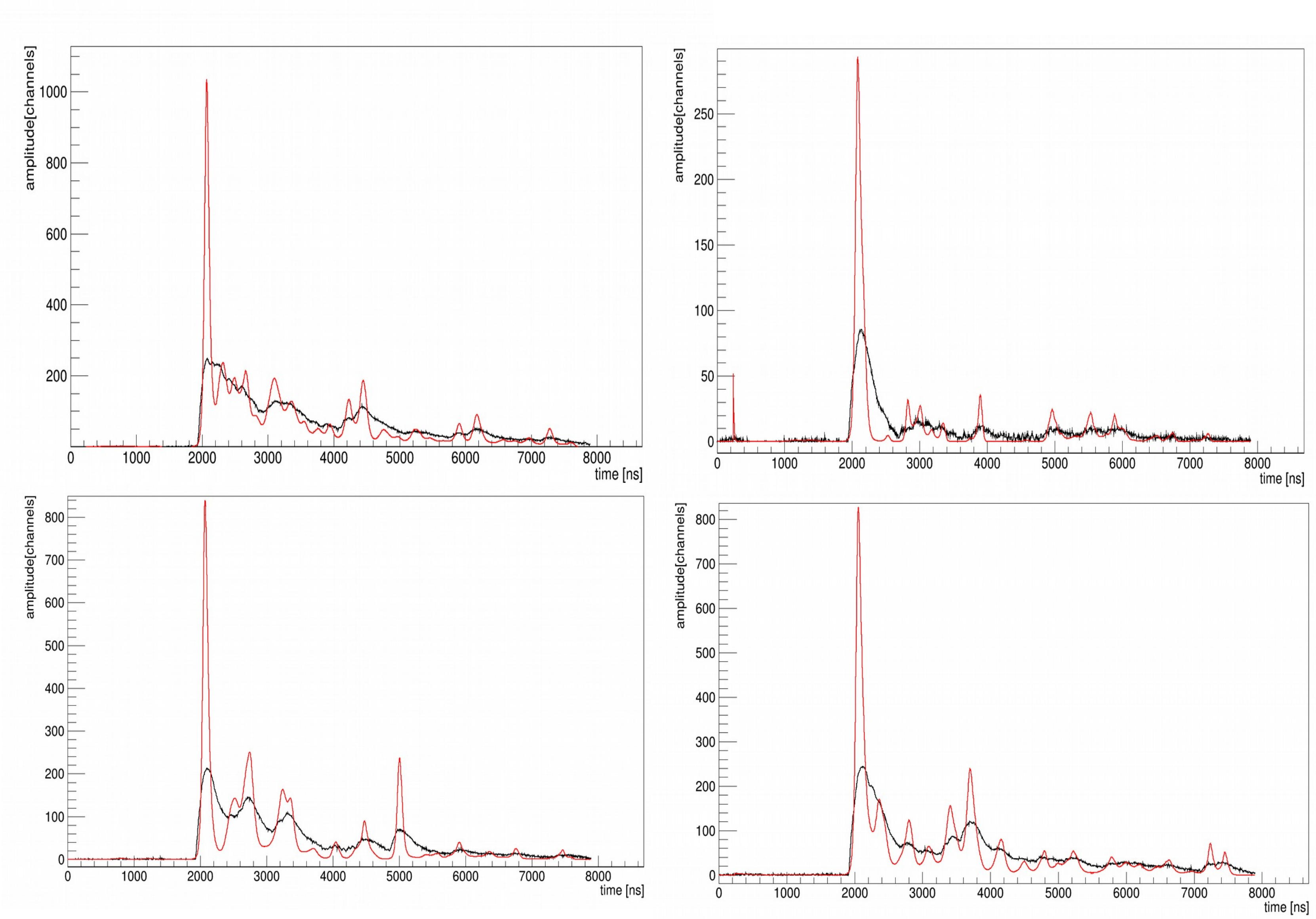






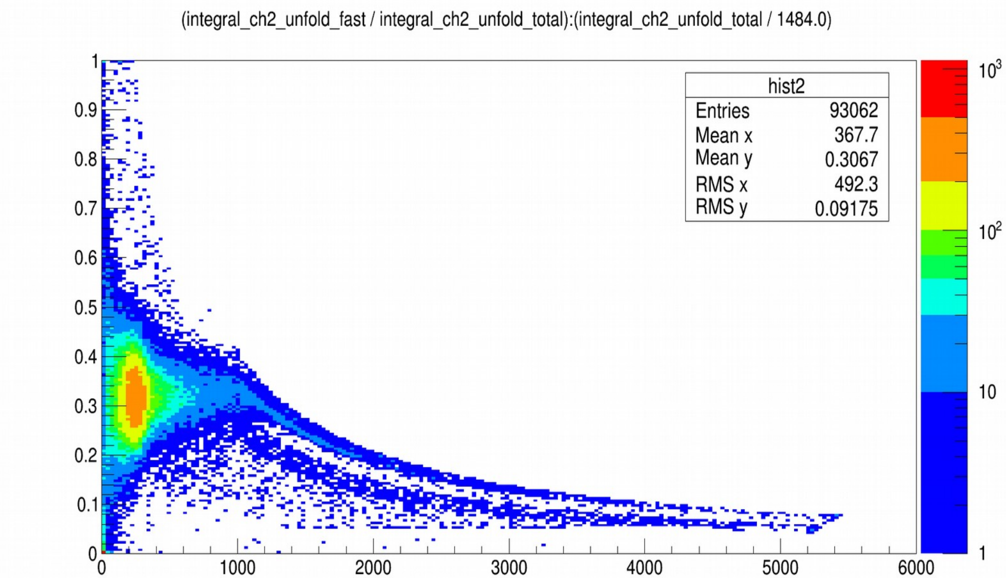
Tspectrum::Deconvolution ch2 results

N_iter = 100, N_rep = 1, boost = 1

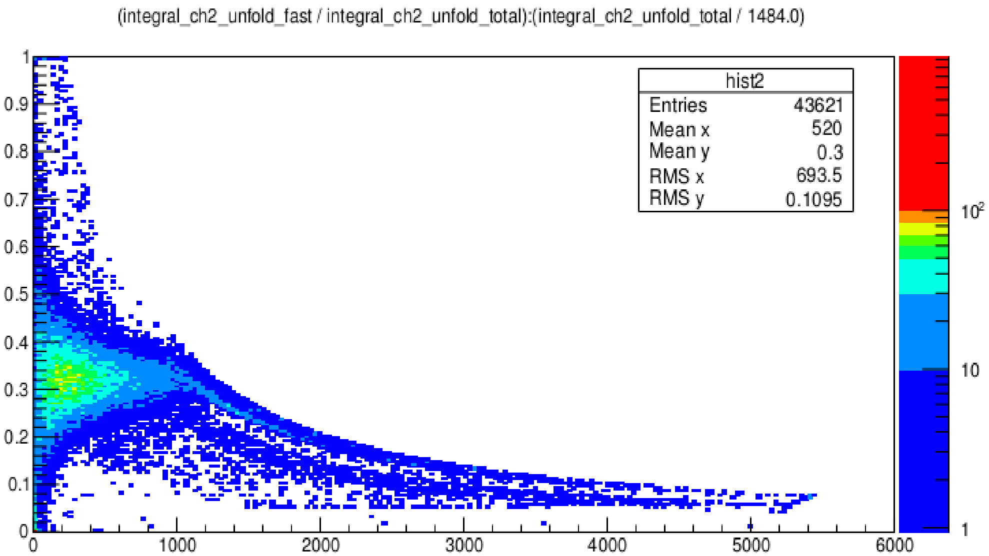


Ratio vs N_pe (ch 2, SiPM)

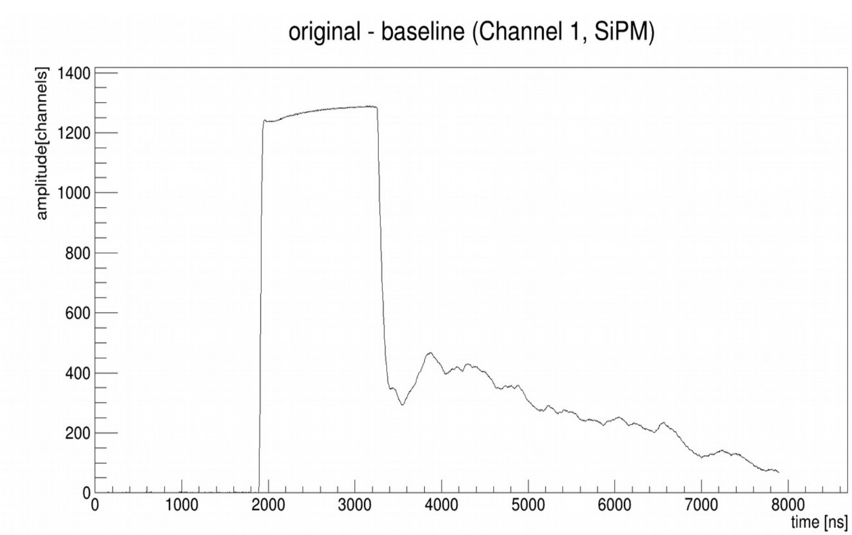
Run 6064 Am

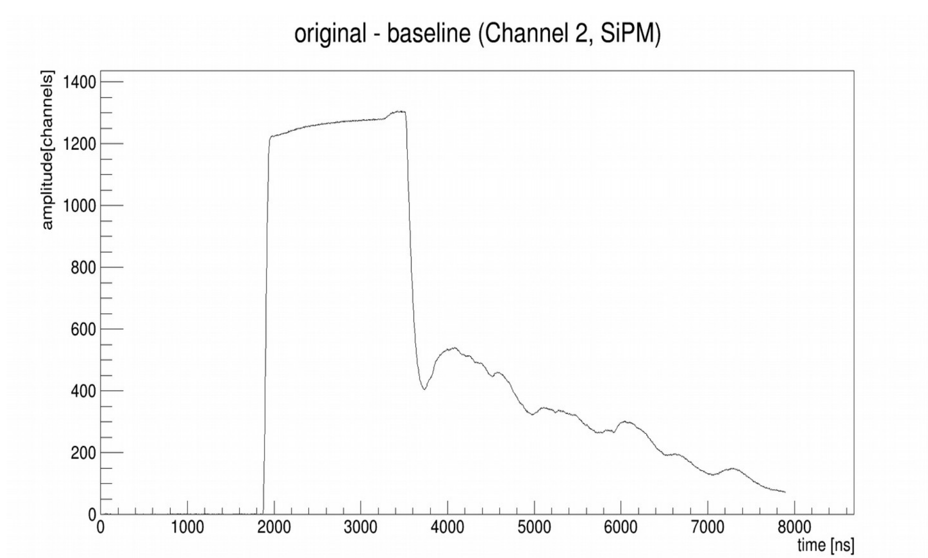


Run 6053 bg



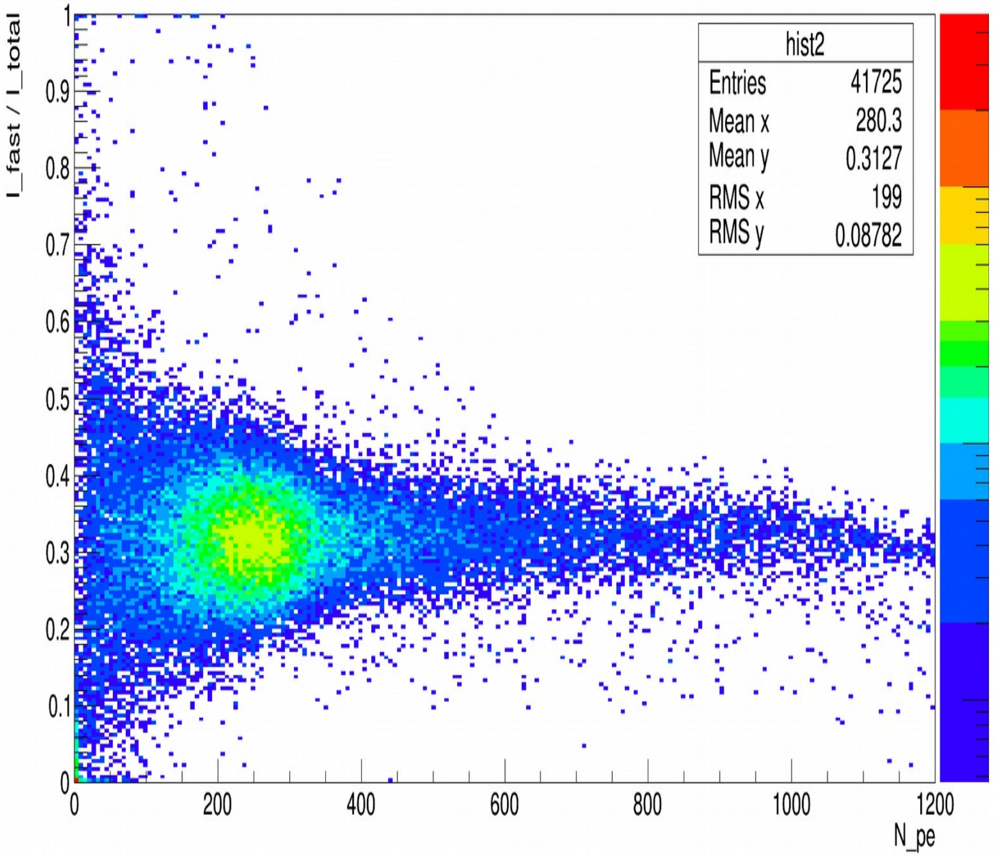
The reason of tail above 1200 pe is saturation of electronics



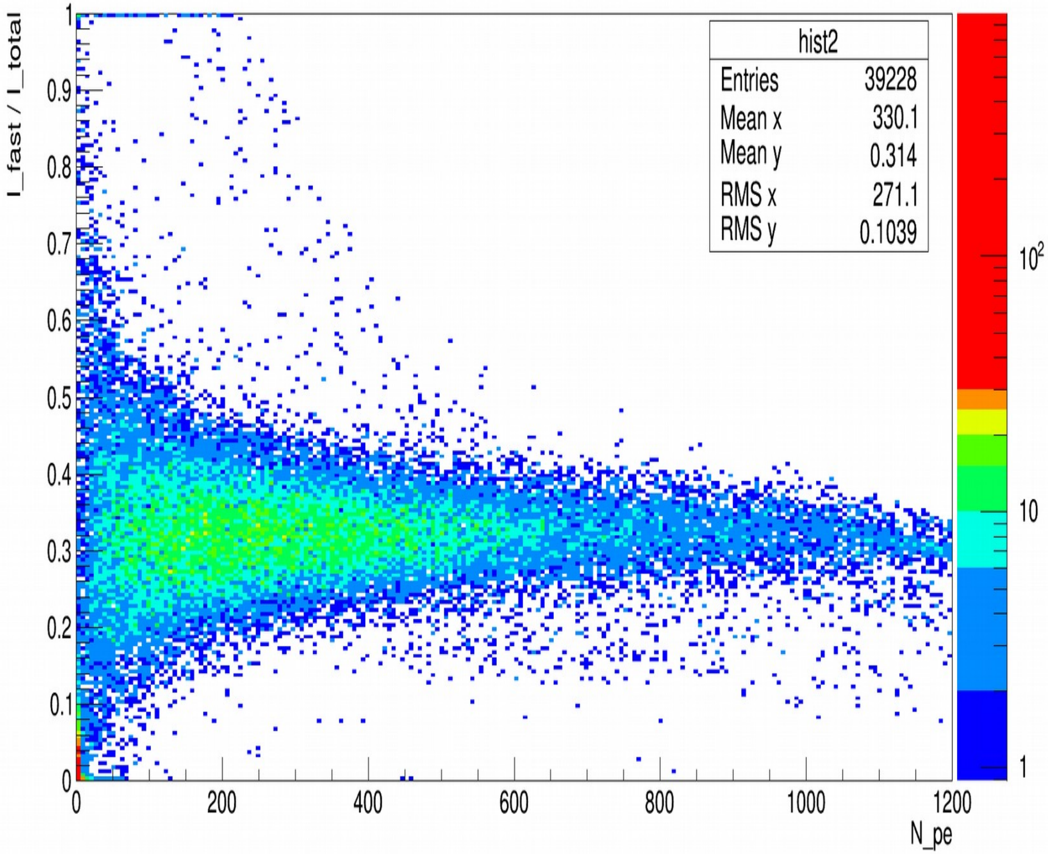


Ratio vs N_pe (ch 2, SiPM)

Run 6064 Am



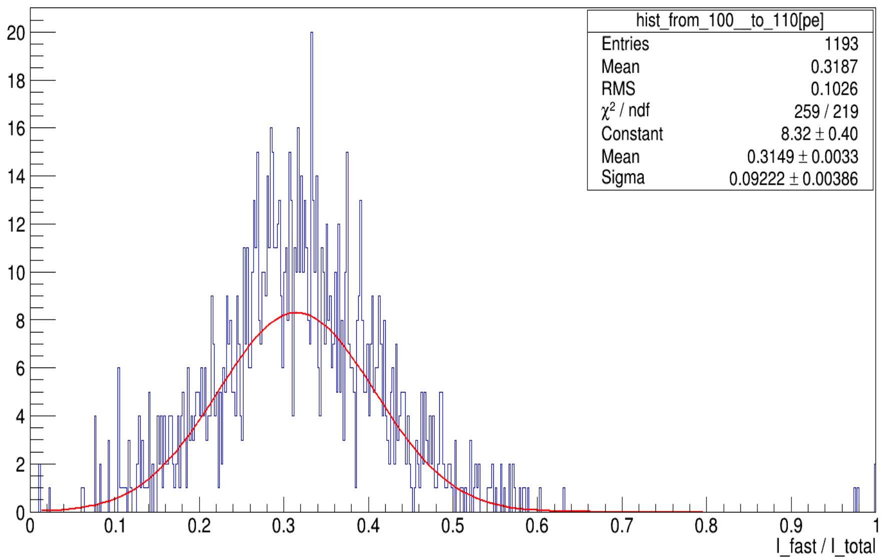
Run 6053 bg

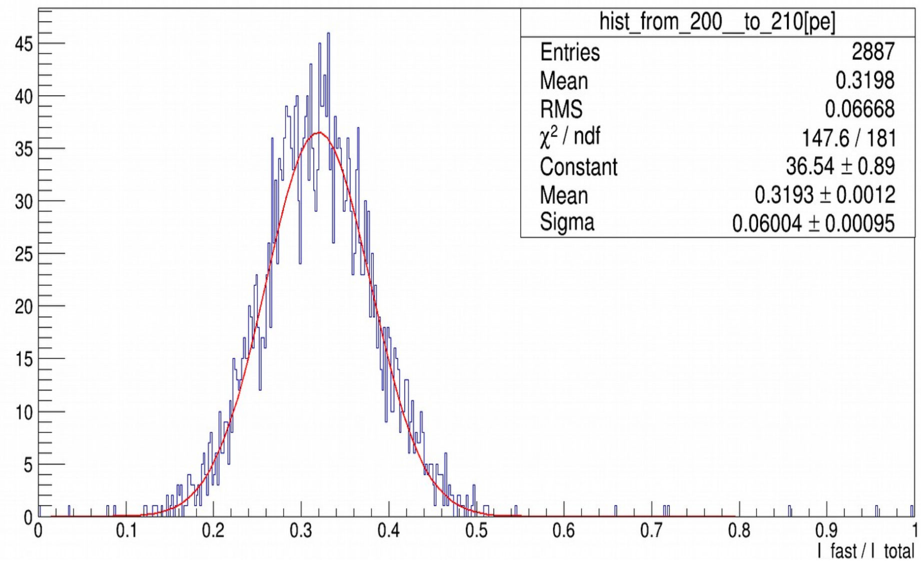


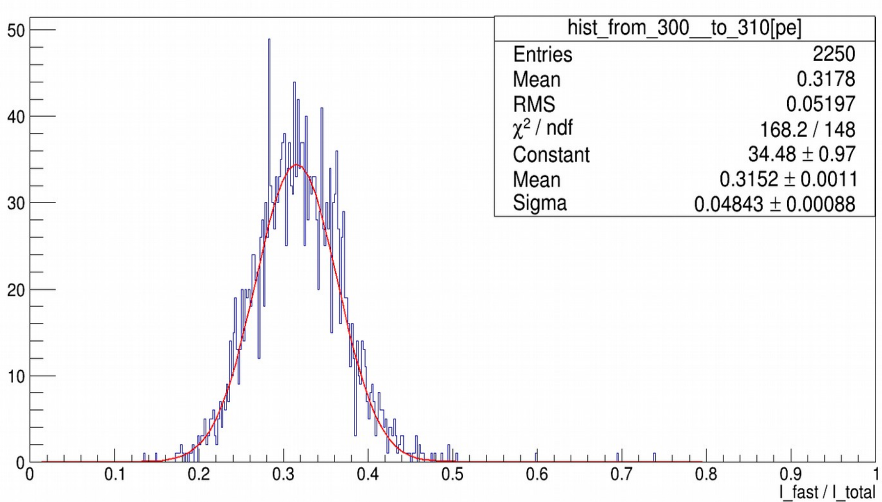
Ratio vs N_pe (ch 2, SiPM): slices

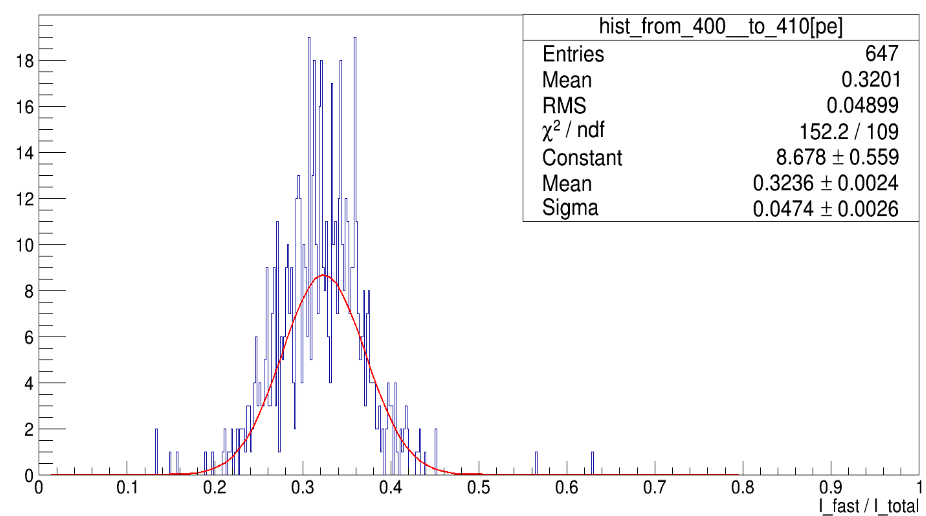
Let's project 2D histogram to N_pe axis with 10 pe step and make gauss fit (binned fit)

Run 6064 Am



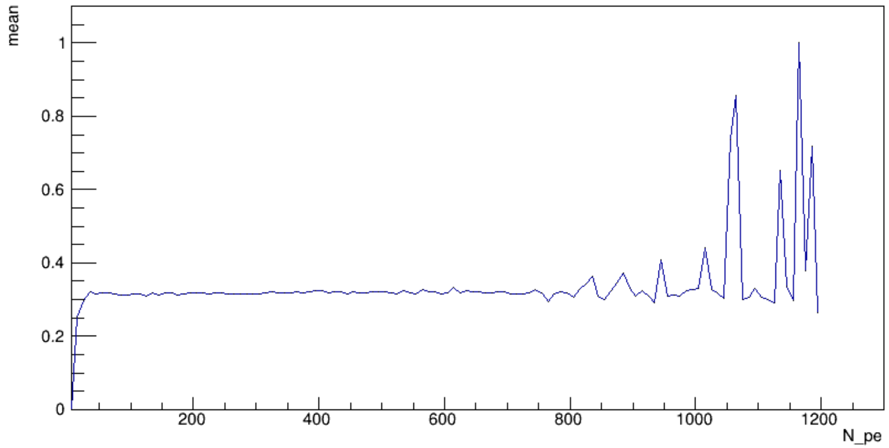


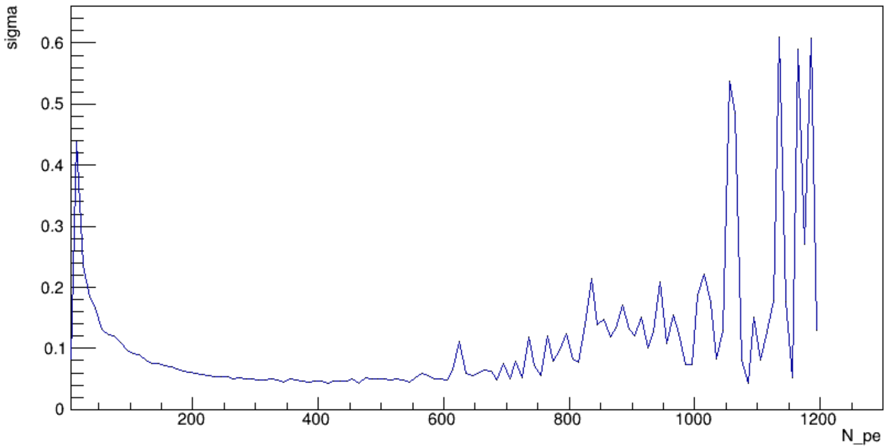


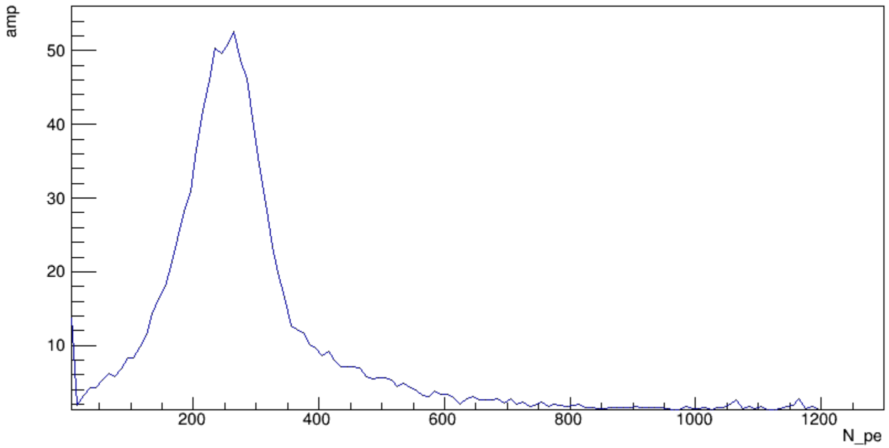


Ratio vs N_pe (ch 2, SiPM): slices

Run 6064 Am

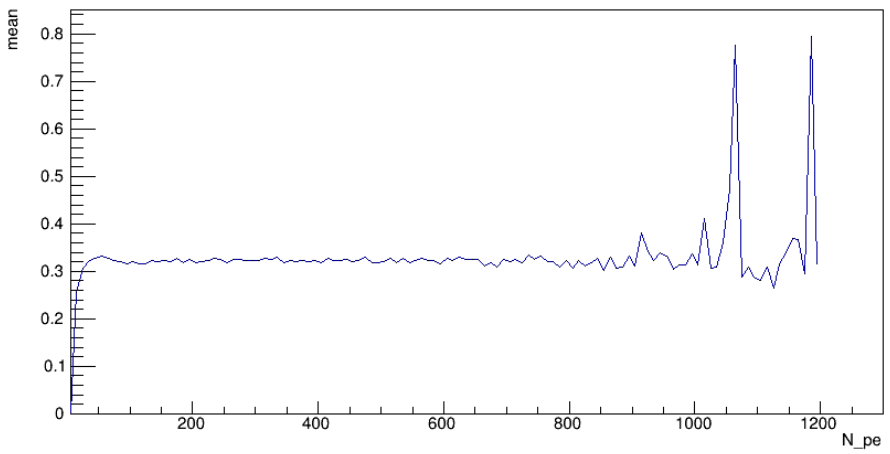


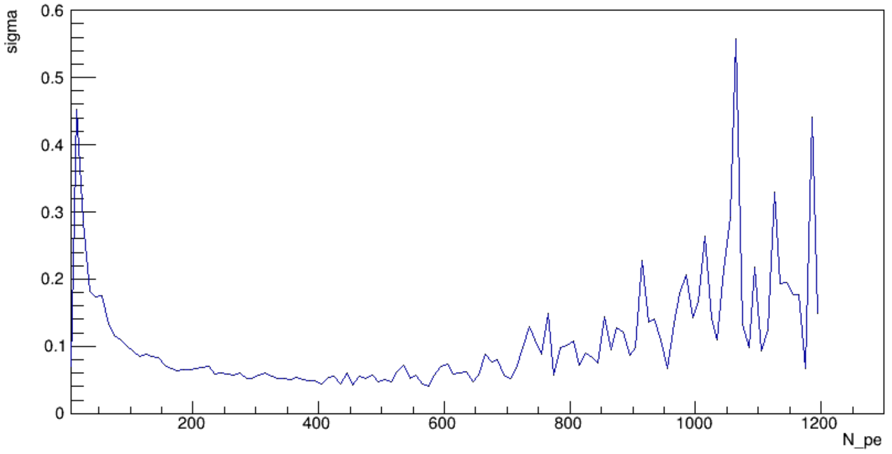


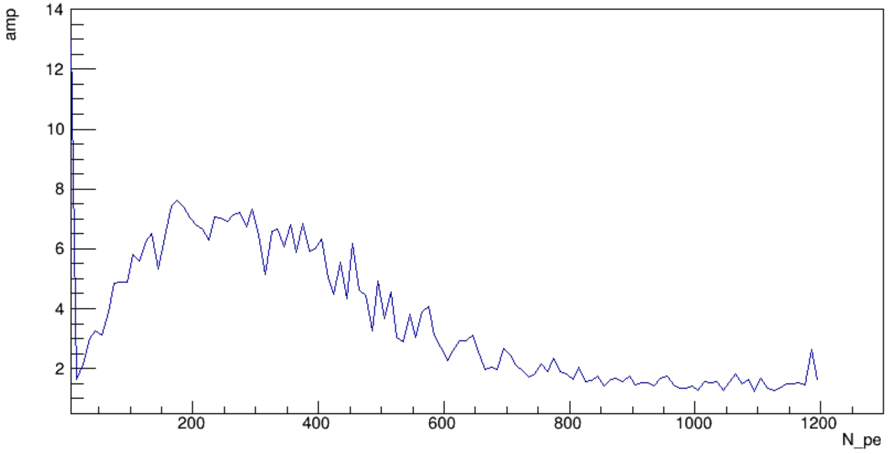


Ratio vs N_pe (ch 2, SiPM): slices

Run 6053 bg



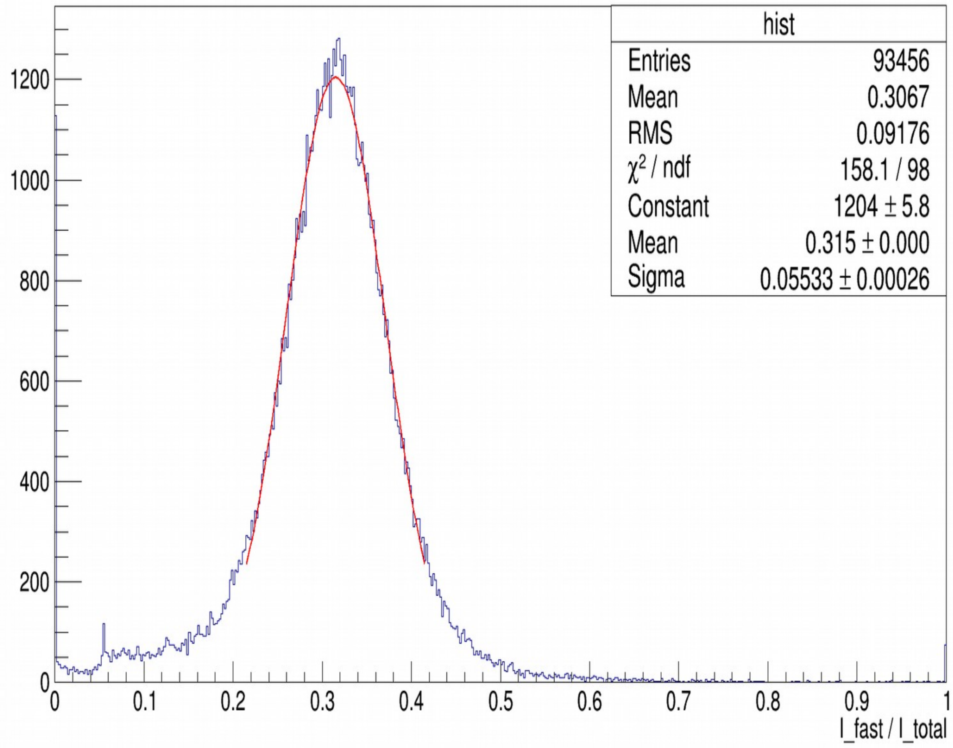




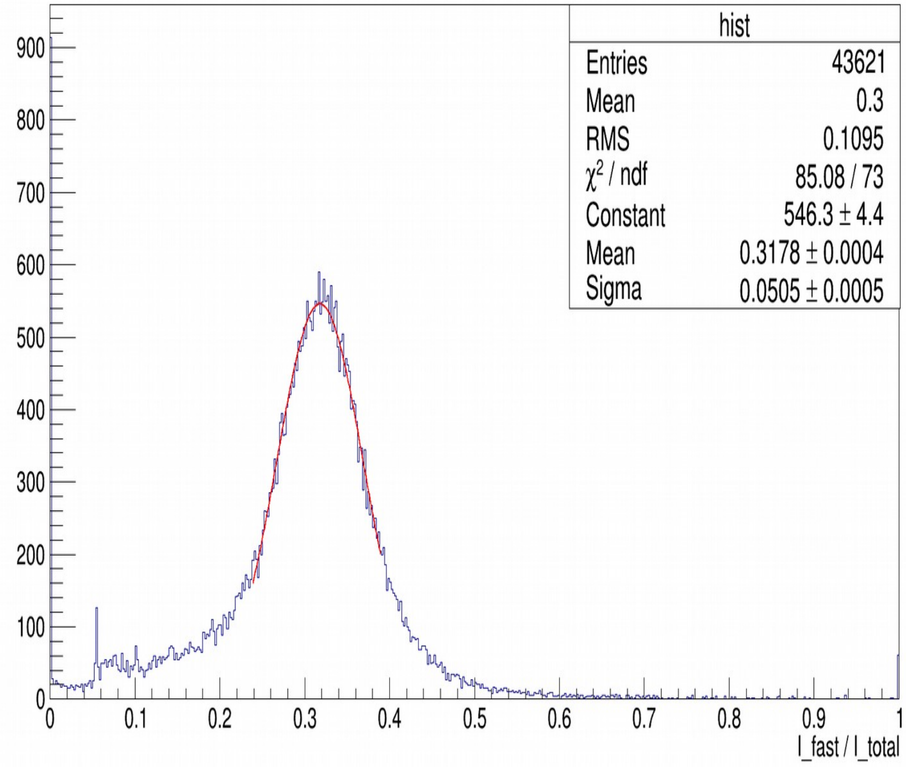
Ratio vs N_pe (ch 2, SiPM)

Let's project 2D histogram to I_fast / I_total axis and make gauss fit (binned fit)

Run 6064 Am

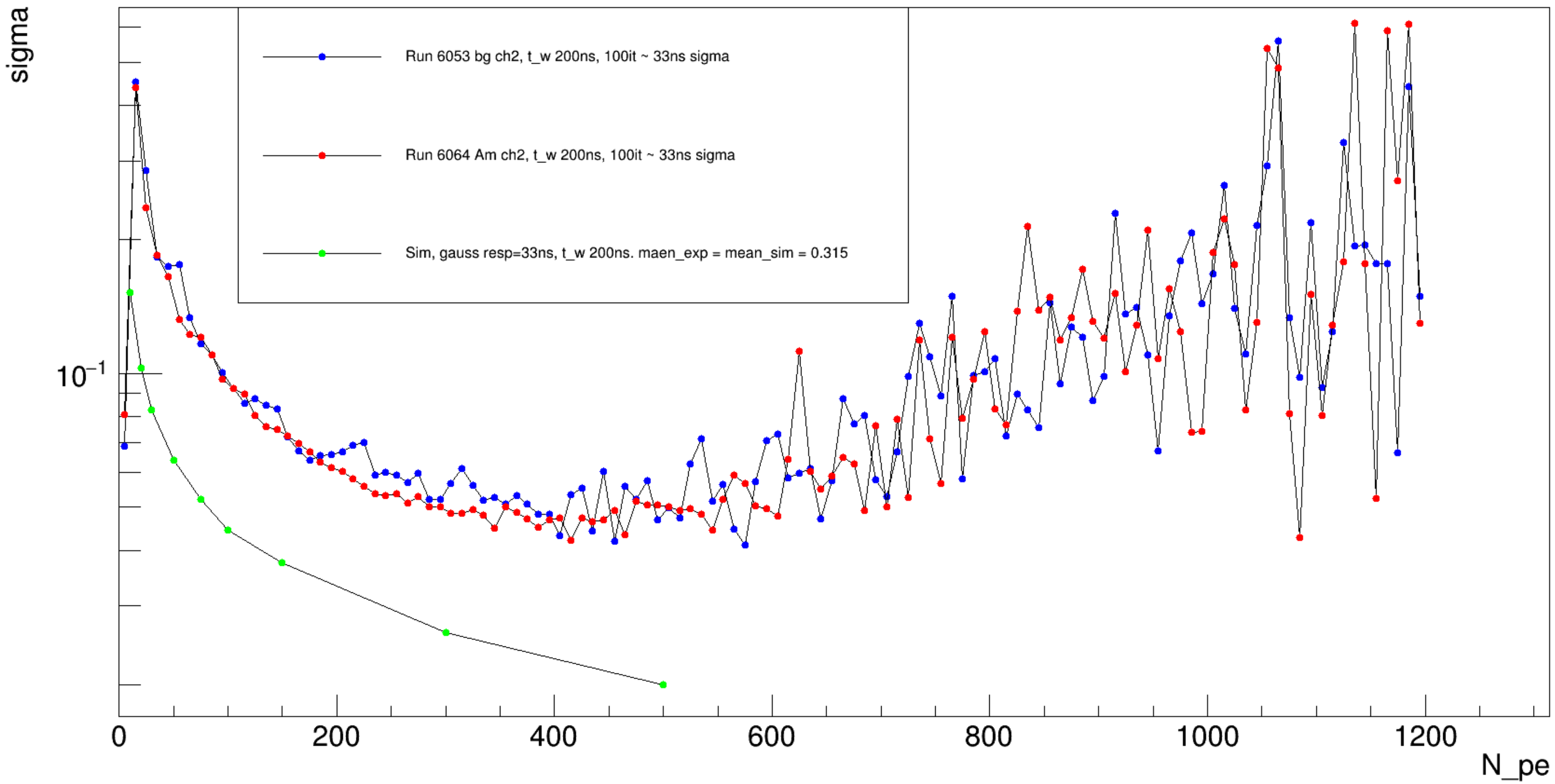


Run 6053 bg



Sigmas comparison

Sigma vs N_pe. Sigma is a param of gauss fit of I_fast / I_slow distribution.
In simulation sigma of gauss response is 33 ns.
PDF = A * PDF_fast + (1 – A) * PDF_slow; A = 0.2308



What are reasons of broadening?

- Another PDF in simulation.
  May be because of wavelength shifter.
- Low frequency noise.
  Integration window is 200ns and noise concentrated from 0 to 5 MHz (i.e. period more than 200 ns). So, it is very probable to have non zero integral of noise.
- We do not know full information about response.
  Length of signal in laser run is 5 us.
  It is not enough, because baseline does not return to origin value.
  Also, we know only estimation of true response, because of experimental errors.
- Consequence of deconvolution algorithm applied to noise data.
  In simulation without noise Gold deconvolution algorithm do not change position and total integral, may be the reason of broadening (I should check). Also we should understand what will be in presence of noise.

Further steps

- Write laser run with longer period (~20 us)
- Understand the reason of non-smooth averaged signal
- Simulate signals for real response, add some noise and make deconvolution
- Find fast algorithm which will work with positive and negative data correctly
- Find optimum of separation algorithm for real data (need neutron run)

You can find source code here
https://github.com/Vlad-ole/s1_fast_slow_separation_darkside

Thank you for your attention!