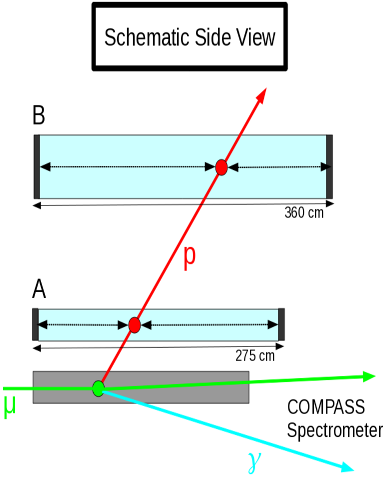
Pulse Analysis

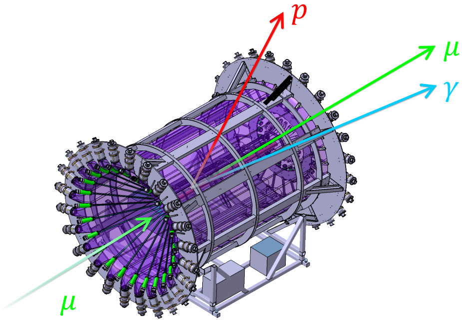
Philipp Jörg HK 35.6

Albert-Ludwigs-University Freiburg

DPG 2012

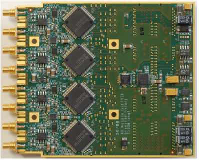
CAMERA-Detector at COMPASS



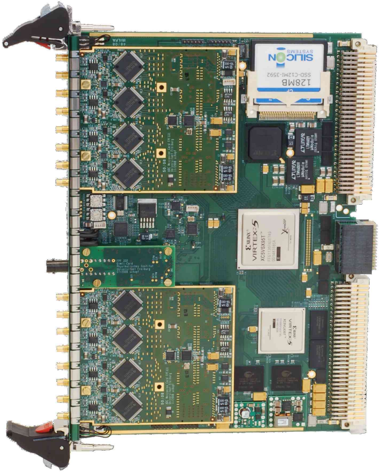


The GANDALF Framework

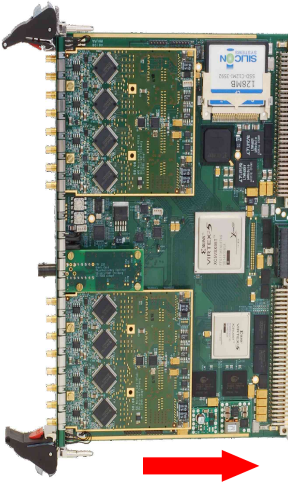
- The GANDALF Framework (see HK 34.5 - Max Büchele and 53.8 - Florian Herrmann)

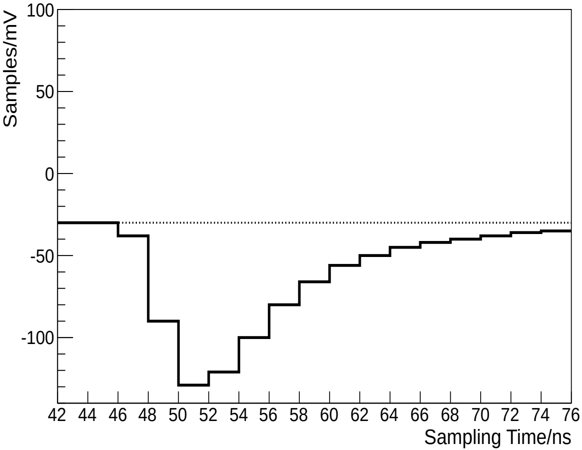


- 12 bit Sampling ADC
- Sampling rate 500 MHz - 1GHz



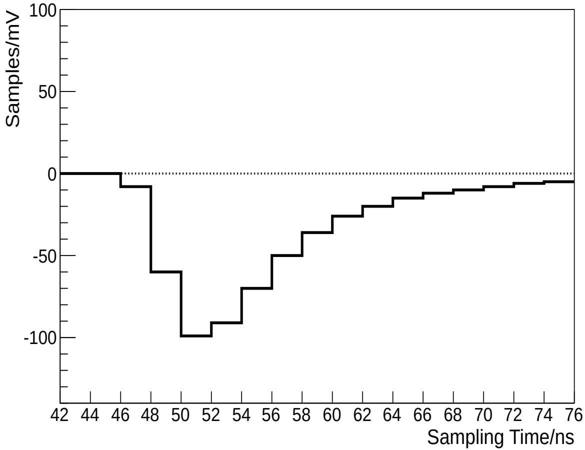
Constant Fraction Algorithm





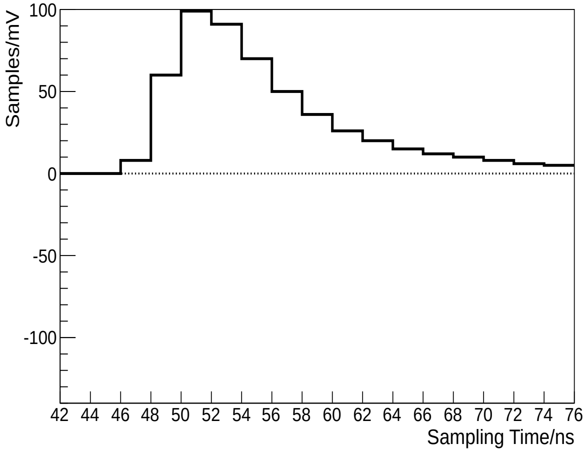
Constant Fraction Algorithm

- Remove baseline bias



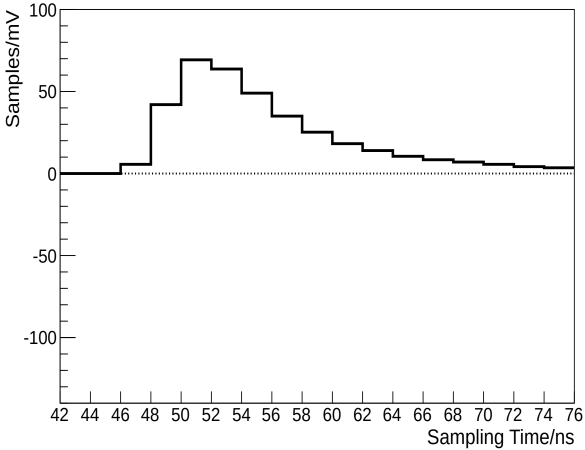
Constant Fraction Algorithm

- Remove baseline bias
- Invert



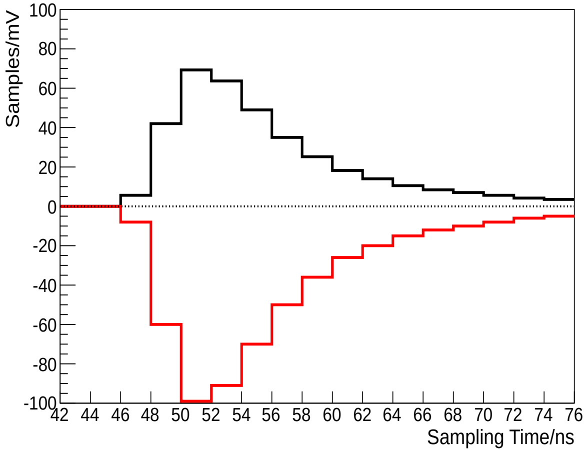
Constant Fraction Algorithm

- Remove baseline bias
- Invert
- Apply Fraction factor



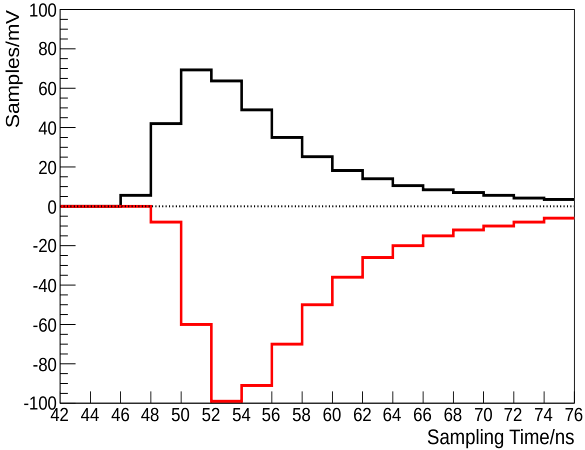
Constant Fraction Algorithm

- Remove baseline bias
- Invert
- Apply Fraction Factor
- Take original



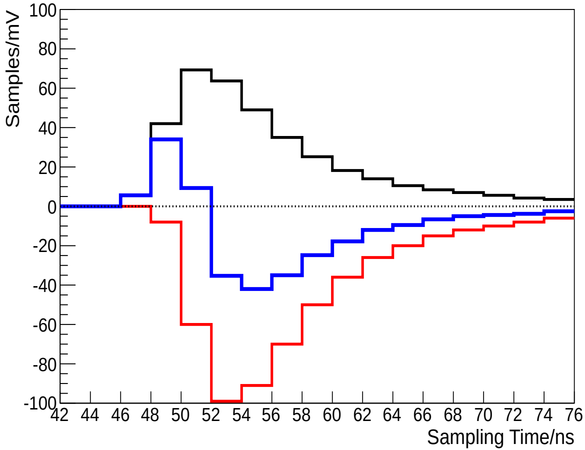
Constant Fraction Algorithm

- Remove baseline bias
- Invert
- Apply Fraction Factor
- Take original & delay

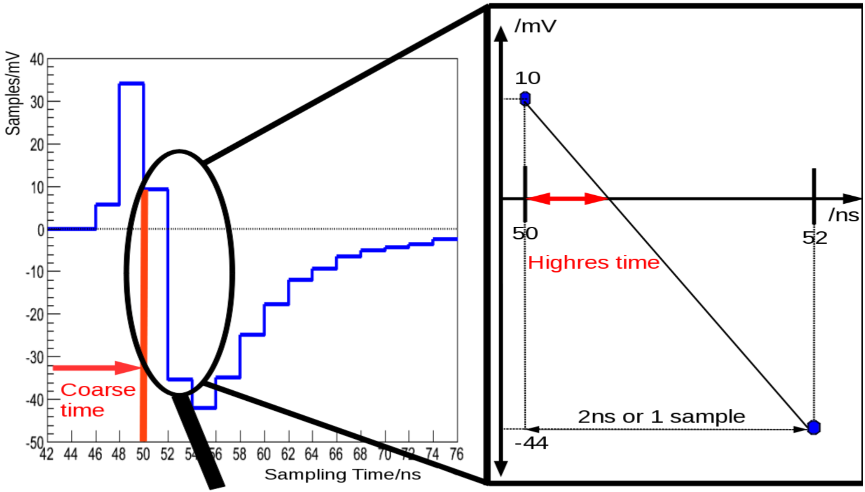


Constant Fraction Algorithm

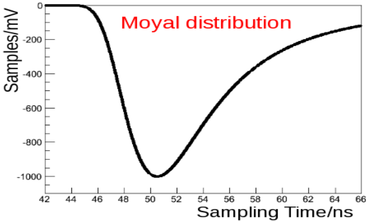
- Remove baseline bias
- invert
- Apply Fraction Factor
- Take original & delay
- Add the pulses



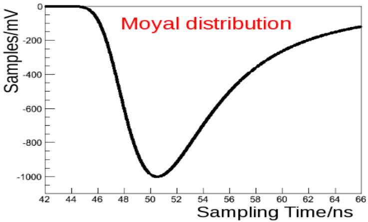
Constant Fraction Algorithm

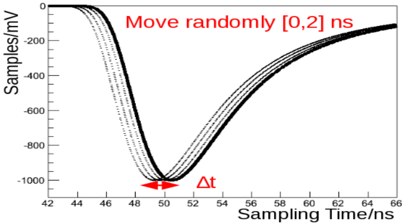


Simulation Input

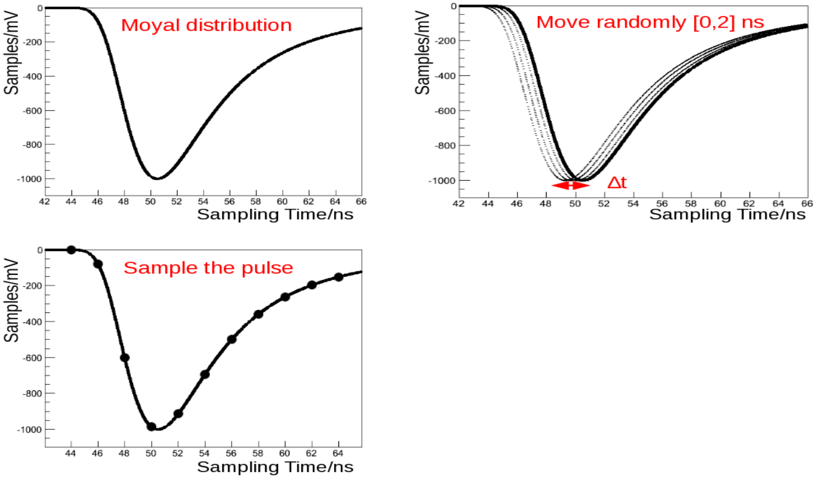


Simulation Input

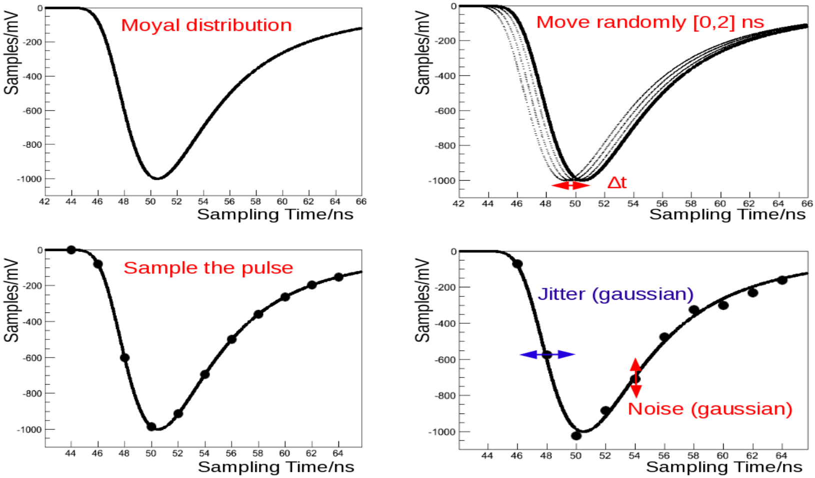




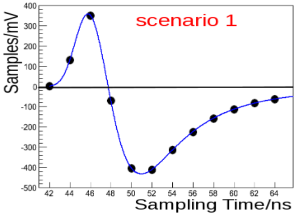
Simulation Input

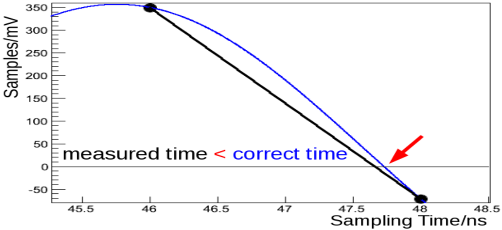


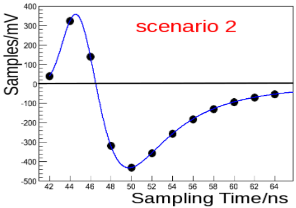
Simulation Input

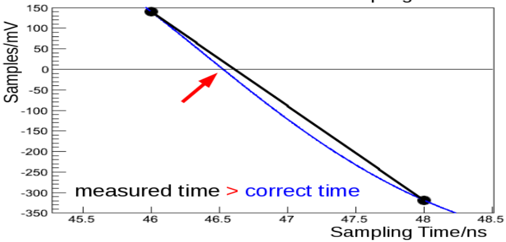


Simulation Input

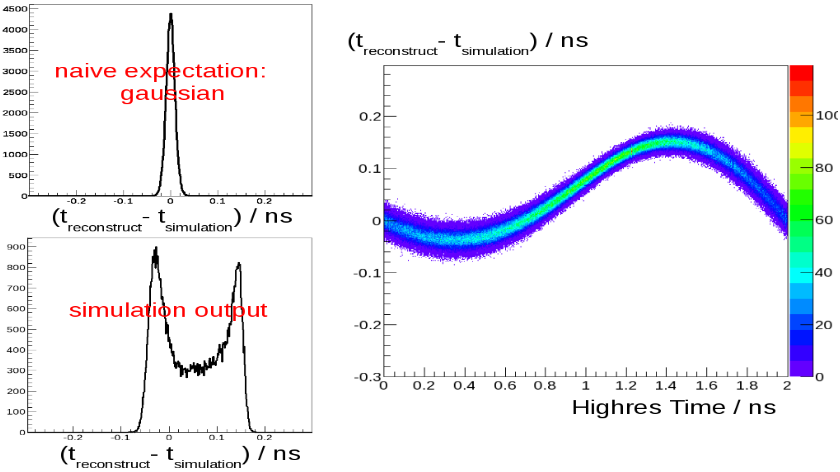




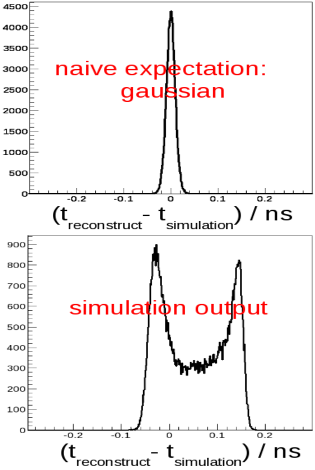


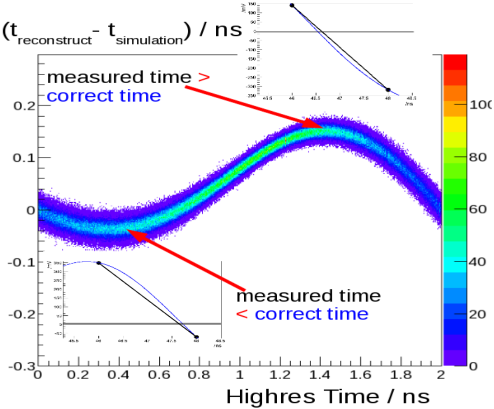


Without Correction

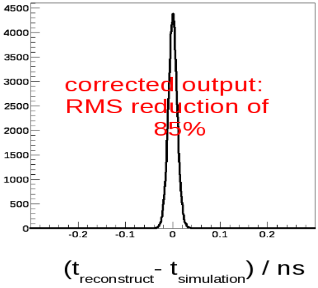


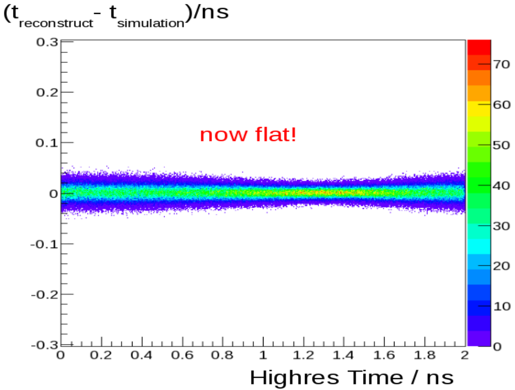
Without Correction



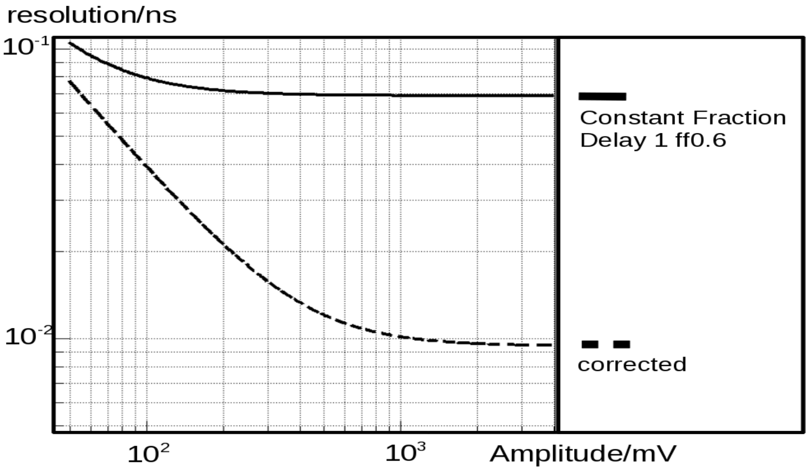


Corrected Data

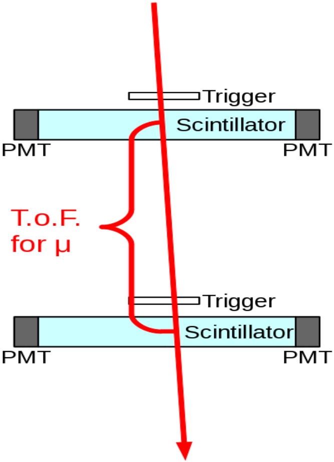


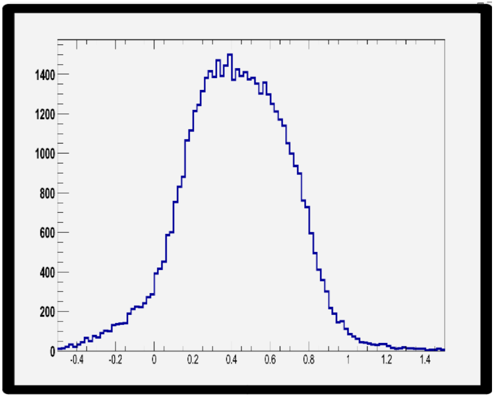


Resolution vs Amplitude



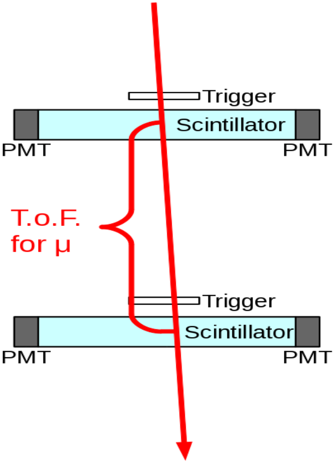
Conventional Constant Fraction

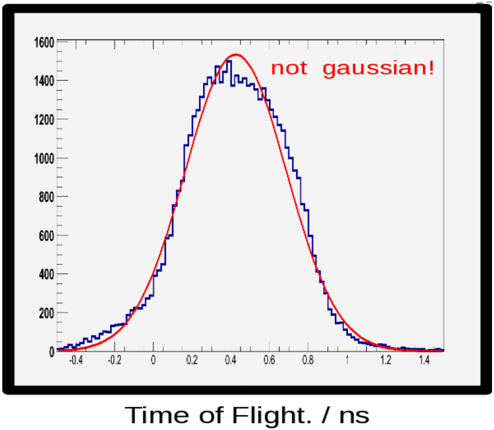




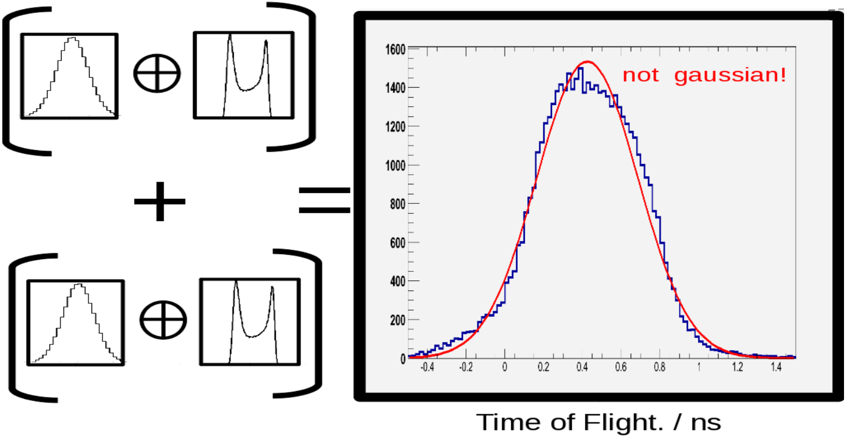
Time of Flight. / ns

Conventional Constant Fraction

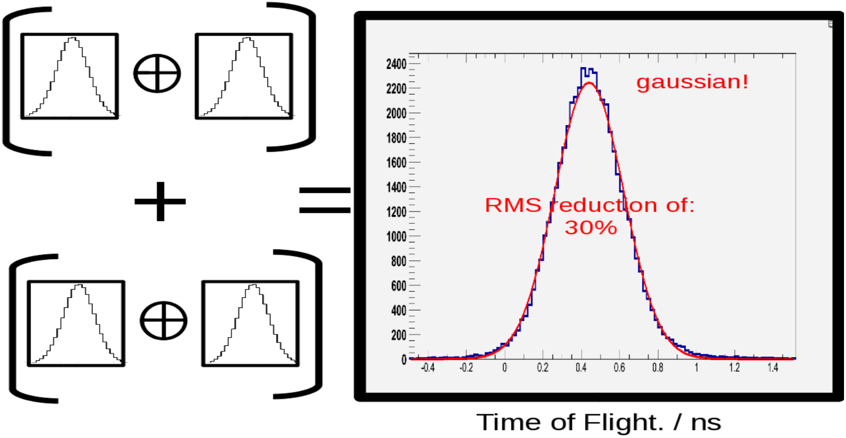




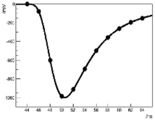
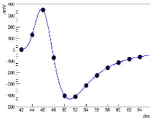
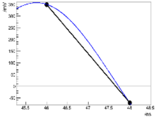
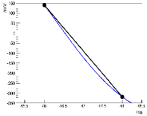
Conventional Constant Fraction



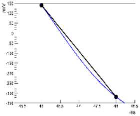
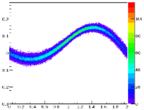
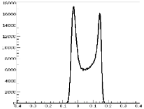
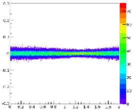
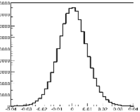
Corrected Constant Fraction



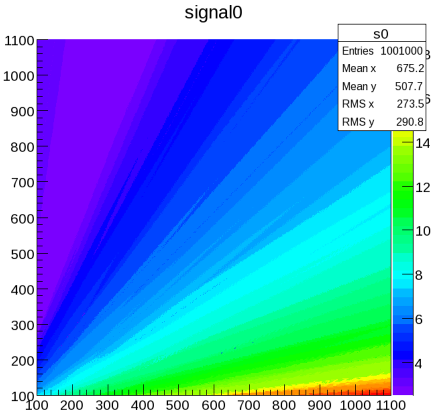
Conclusion

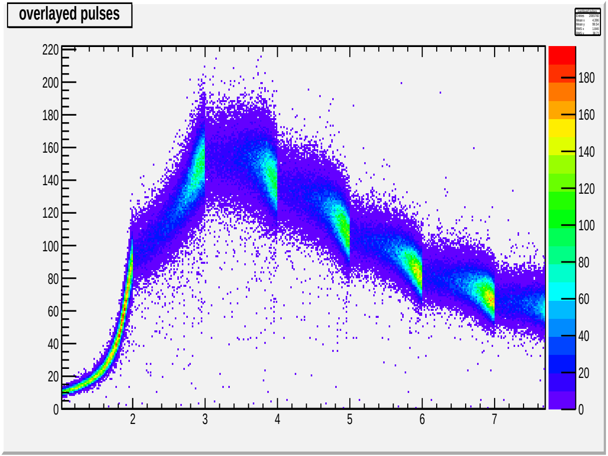
- Method to increase the time resolution
- In case of the T.o.F. measurement: an improvement in time resolution of about 30 %
- In any case a reduction of systematic errors of the algorithm
- hadron.physik.uni-freiburg.de/gandalf

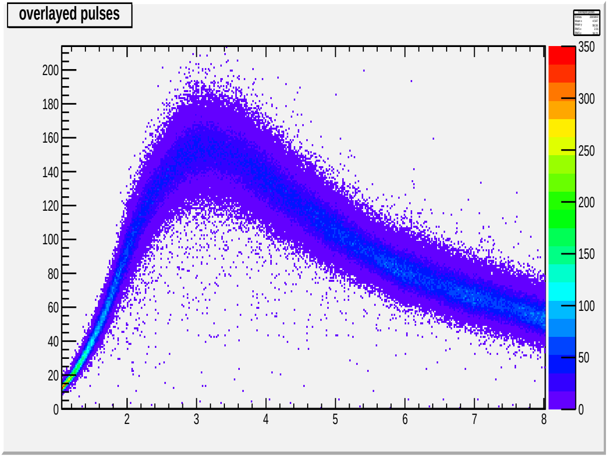
double pulse plot



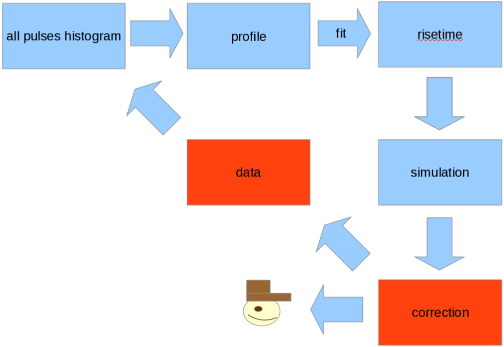
saclay unco


overlayed pulses
220
200
180
160
140
120
100
80
60
40
20
0
2
3
4
5
6
7
180
160
140
120
100
80
60
40
20
0

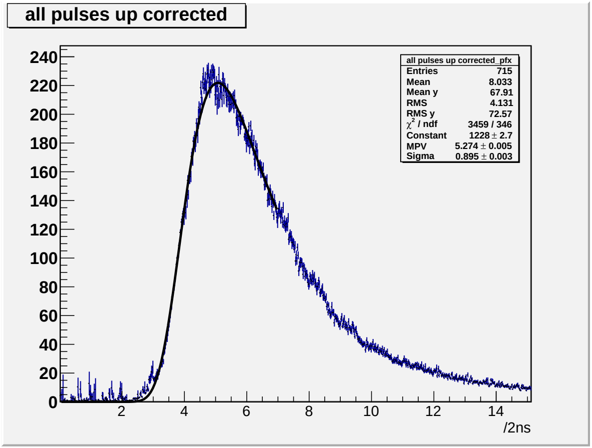
saclay co


overlayed pulses
200
180
160
140
120
100
80
60
40
20
0
2
3
4
5
6
7
8
350
300
250
200
150
100
50
0

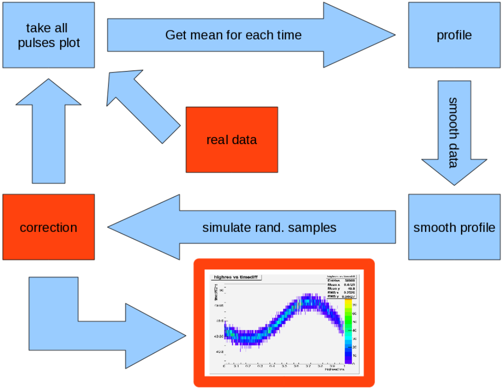
get the correction: first method



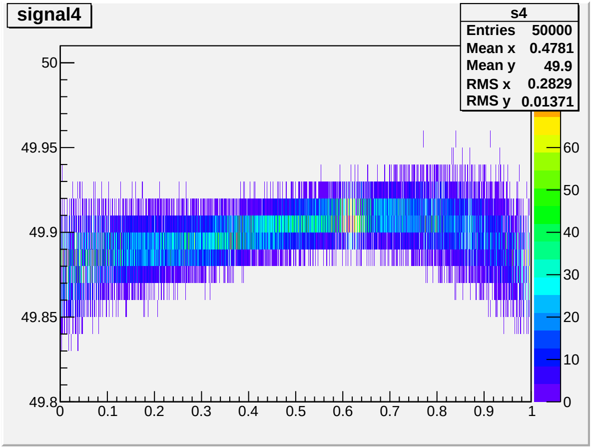
profile and fit Aup



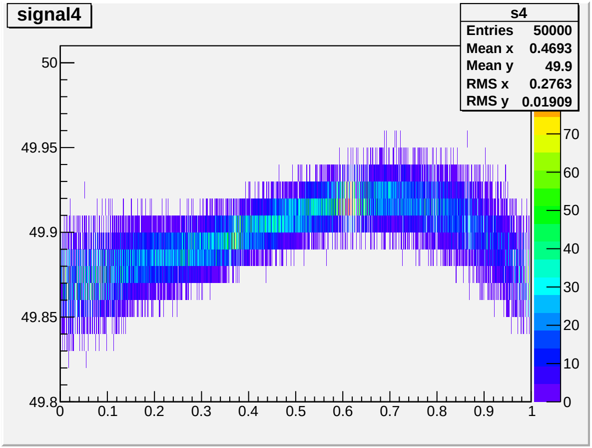
Iteration Process



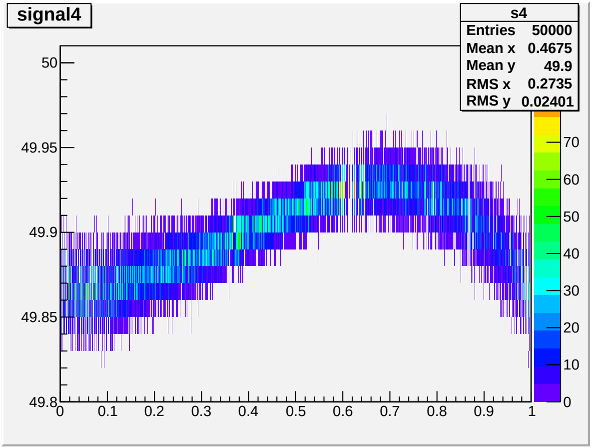
First Iteration



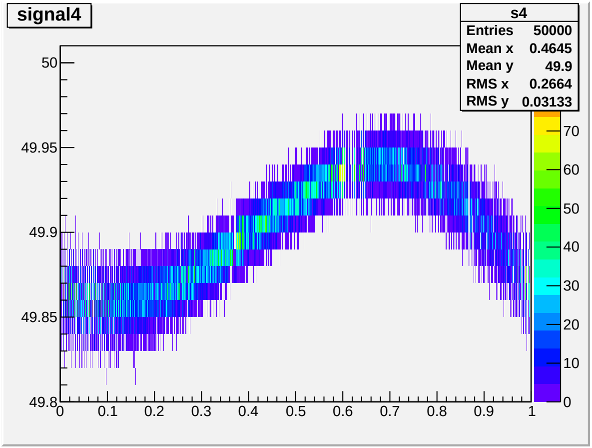
Second Iteration



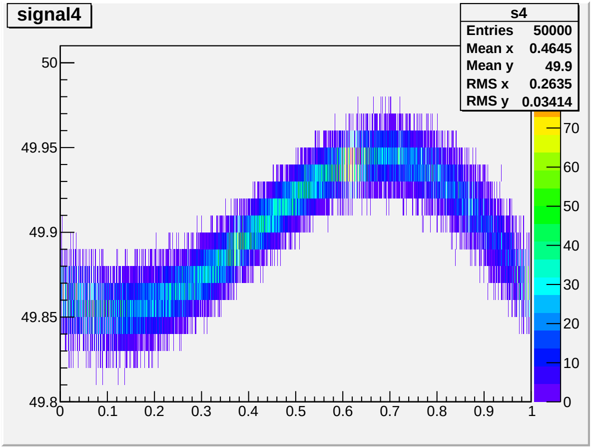
Third Iteration



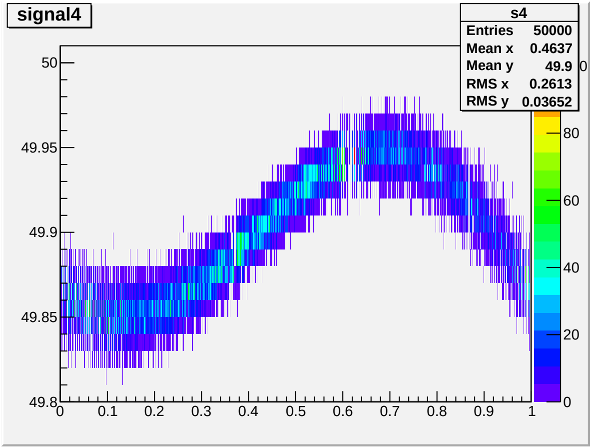
4 th. Iteration



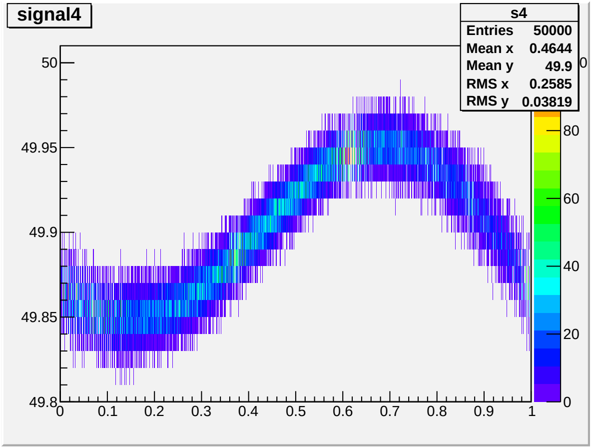
5 th. Iteration



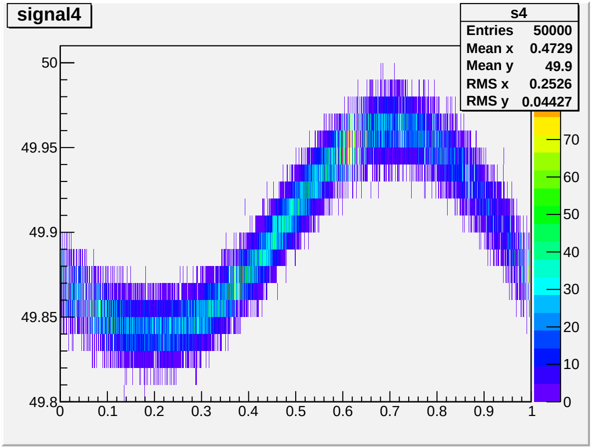
6 th. Iteration



7 th. Iteration

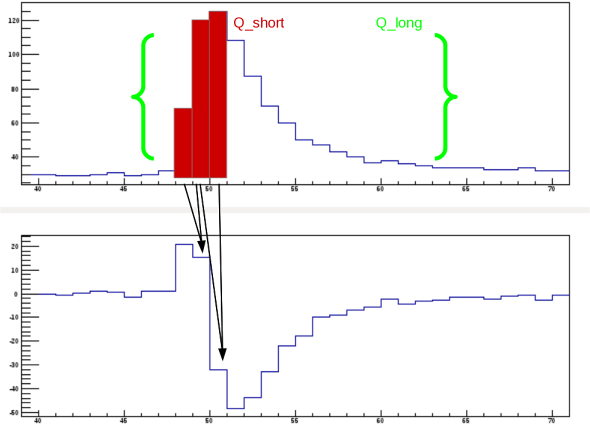


300 th. Iteration

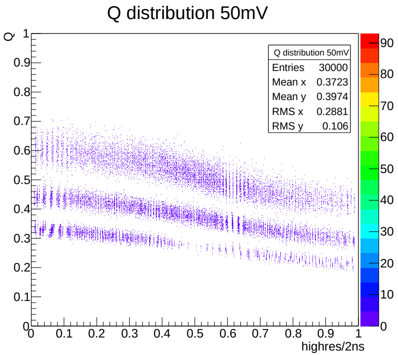


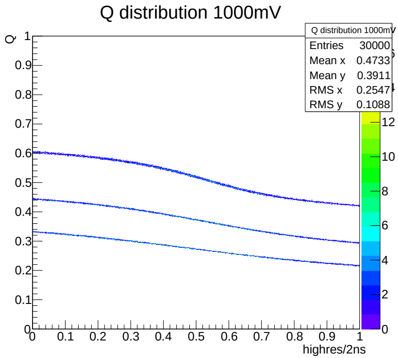
Definition of Q

- $Q = \frac{Q_{short}}{Q_{long}}$

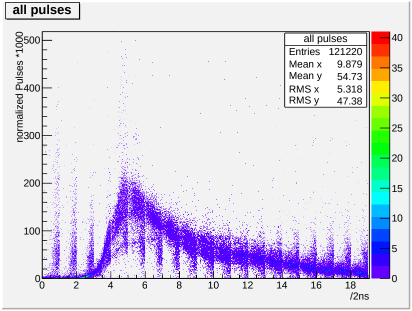


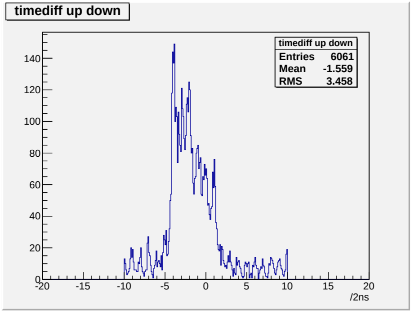
Q Classification Simulation: risetime 1, 1.5 and 2 samples



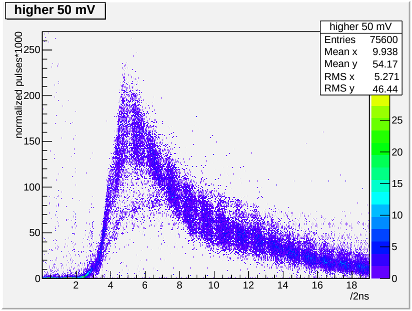


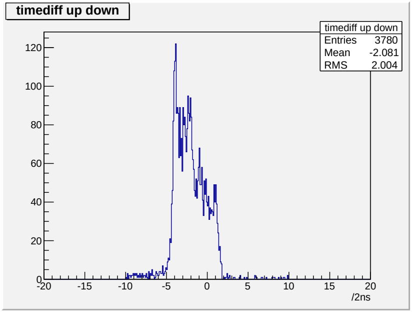
All Pulses


all pulses
all pulses
Entries 121220
Mean x 9.879
Mean y 54.73
RMS x 5.318
RMS y 47.38
normalized Pulses *1000
/2ns

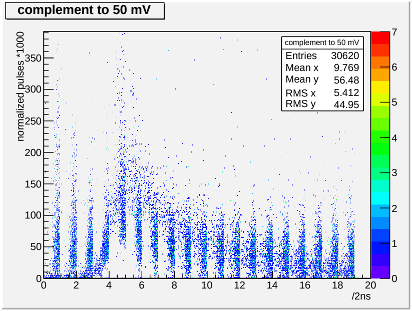

timediff up down
timediff up down
Entries 6061
Mean -1.559
RMS 3.458
/2ns

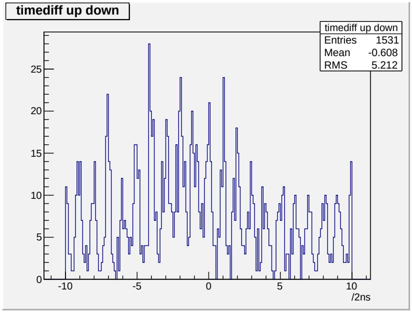
All Pulses higher 50 mV



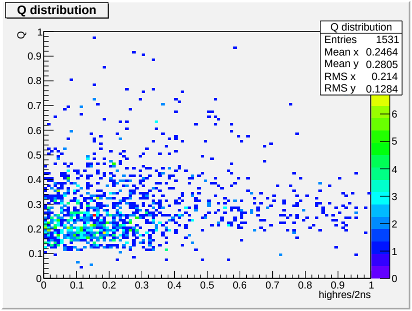


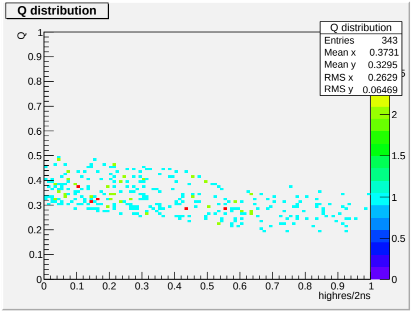
Complement to 50 mV



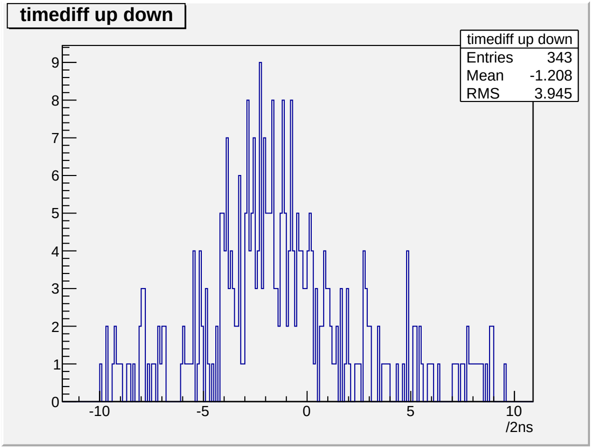


Q for complement and cut



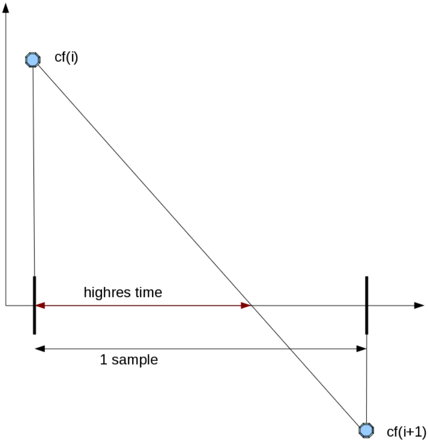


Time Difference for cutted complement



highres time

- $highres = \frac{-cf(i)}{cf(i+1) - cf(i)}$



calc

$$M(x) =$$
$$exp(\frac{-x}{2w} - \frac{exp(-x/w)}{2} - \frac{1}{2w} + \frac{1}{2w} - \frac{exp(-(x+1)/w)}{2} + \frac{exp(-(x+1)/w}{2})$$
$$= exp(1/2w)exp(exp(-(x+1)/w)/2)exp(exp(-x/w)/2)M(x+1)$$
mit $a = exp(\frac{1}{2w})$
$$= aexp((exp(-(x+1)/w) - exp(-x/w))/2)M(x+1)$$
$$= aexp(\frac{1}{2}(\frac{1}{a^2})^x(\frac{1}{a^2} - 1)M(x+1)$$
kurz $M(x) = B(x)M(x+1)$
mit $B(x) = aexp(\frac{1}{2}(\frac{1}{a^2})^x(\frac{1}{a^2} - 1))$
$$Highres = \frac{-fM(x+1)+M(x)}{fM(x+2)-M(x+1)-fM(x+1)+M(x)}$$
$$= \frac{-fM(x+1)+B(x)M(x+1)}{f\frac{1}{B(x+1)}M(x+1)-M(x+1)-fM(x+1)+B(x)M(x+1)}$$
$$= \frac{B(x)-f}{\frac{f}{B(x+1)}-(1+f)+B(x)}$$ hier fallen die Amplituden raus

calc

Nullstelle:

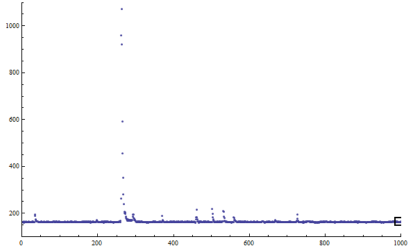
$B(x) - f = 0$

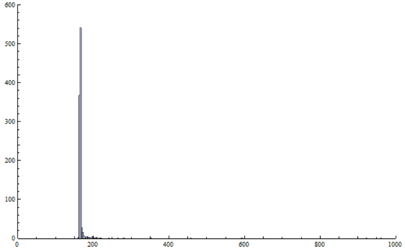
$x = ln(\frac{2ln(f/a)}{\frac{1}{a^2}\text{-}1})/ln(\frac{1}{a^2})$

$=>$Parametrisierung: $(Highres(x), x + Highres(x))$mit

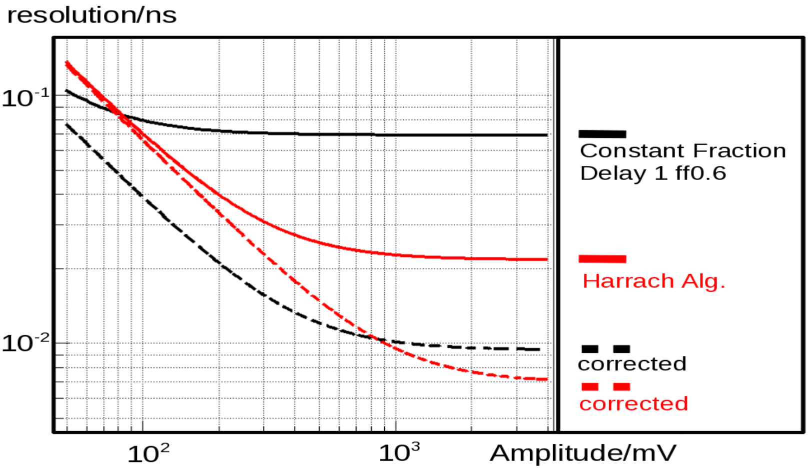
$x \in \left[ln(\frac{2ln(f/a)}{\frac{1}{a^2}\text{-}1})/ln(\frac{1}{a^2}), ln(\frac{2ln(f/a)}{\frac{1}{a^2}\text{-}1})/ln(\frac{1}{a^2}) - 1\right]$

baseline


1000
800
600
400
200
0
200
400
600
800
1000
E

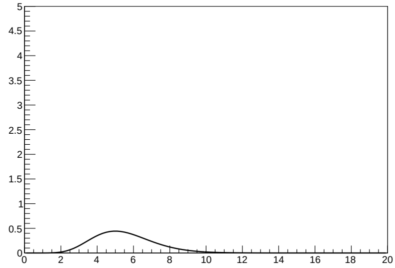

600
500
400
300
200
100
0
0
200
400
600
800
1000

Resolution vs Amplitude



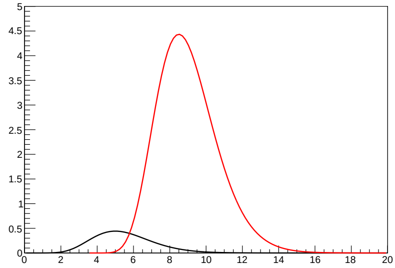
Harrach algorithm

- designed for poisson pulses
- $Ct^{\frac{t_m}{\tau}} e^{-\frac{t}{\tau}}$



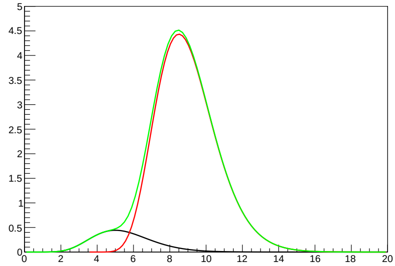
Harrach algorithm

- and pile up situations



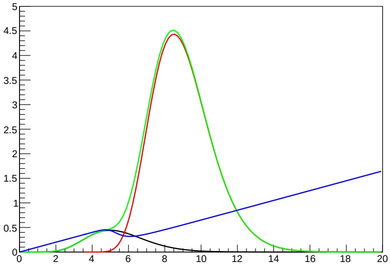
Harrach algorithm

- to get estimates of the time of each pulse



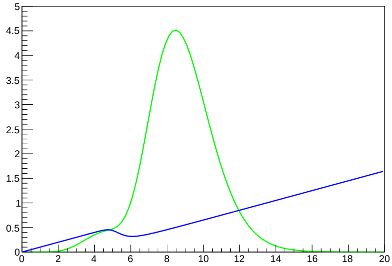
Harrach algorithm

- arises from a model of the phase derivative of the pulse
- in practice one has to form the expression $\frac{f(t)}{f'(t)+\frac{f(t)}{\tau}}$(blue)
- so one has to form a discrete derivative (like cf)



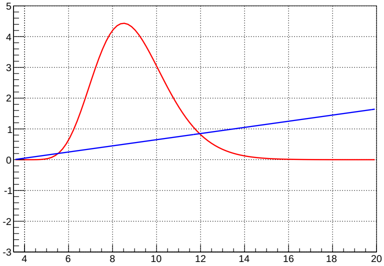
Harrach algorithm

- without knowing the real pulse shape, τ is treated as a free parameter in the simulation



Harrach algorithm

- later results are for one pulse



Harrach algorithm

- produce zero crossing to get time estimate