Toy Model For Hyperfine Measurement

Adriano Del Vincio, Germano Bonomi,
Simone Stracka

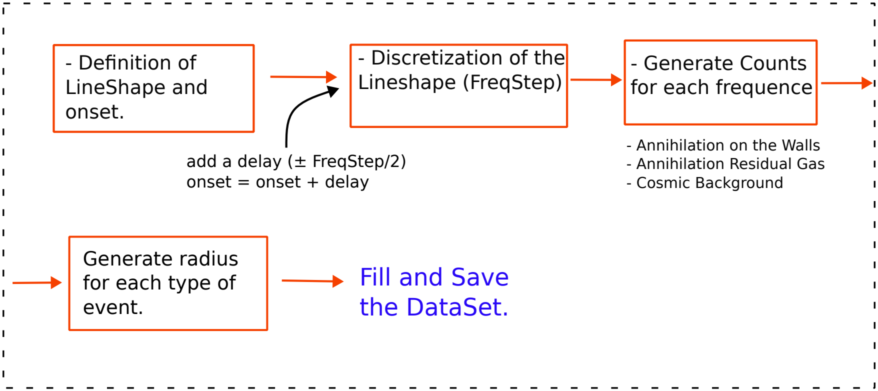
University of Brescia, INFN Pisa

December 5, 2023

Scheme of the Simulation

Scheme of the Monte Carlo Toy for generating the events

Loop from 0 to Ntrials
Inner Loop: from 0 to Repetition



- Definition of LineShape and onset.

- Discretization of the Lineshape (FreqStep)

add a delay (± FreqStep/2)
onset = onset + delay

- Generate Counts for each frequence

- Annihilation on the Walls
- Annihilation Residual Gas
- Cosmic Background

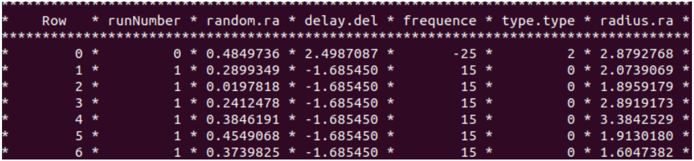
Generate radius for each type of event.

Fill and Save the DataSet.

In this simulation, the data are created and analyzed using *RDataFrame* framework.

A brief introduction about the Monte Carlo

We have developed a Monte Carlo Toy that produces two `.root` dataset files. The variables are columns of values that are shown in the figure below:



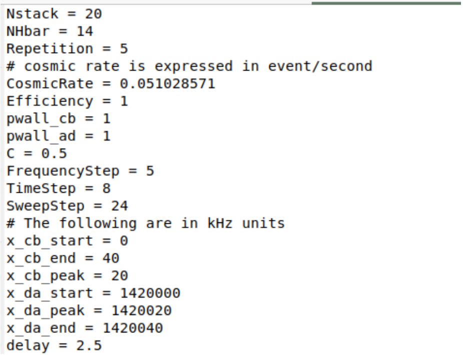
```
************************************************************************************
*    Row   * runNumber * random.ra * delay.del * frequence * type.type * radius.ra *
************************************************************************************
*        0 *         0 * 0.4849736 * 2.4987087 *       -25 *         2 * 2.8792768 *
*        1 *         1 * 0.2899349 * -1.685450 *        15 *         0 * 2.0739069 *
*        2 *         1 * 0.0197818 * -1.685450 *        15 *         0 * 1.8959179 *
*        3 *         1 * 0.2412478 * -1.685450 *        15 *         0 * 2.8919173 *
*        4 *         1 * 0.3846191 * -1.685450 *        15 *         0 * 3.3842529 *
*        5 *         1 * 0.4549068 * -1.685450 *        15 *         0 * 1.9130180 *
*        6 *         1 * 0.3739825 * -1.685450 *        15 *         0 * 1.6047382 *
```

Figure: Structure of the dataset.

- *runNumber*: identifies which run the event belong to (from 0 to $Repetition - 1$)
- *random*: values uniform distributed from 0 to 1, can be used to randomize the selection or for sub-sampling in the data
- *delay*: store the onset delay
- *frequence*: the frequency of the event
- *type*: type of the event: 0 annihilation on the walls, 1 residual gas annihilation, 2 cosmic event
- *radius*: radius of the annihilation vertex.

A brief introduction about the Monte Carlo

The Annihilation on the walls are generated as function of the frequency, using the two line-shapes of the transitions (c → b) and (d → a). The Annihilation on the residual gas and the cosmic background are generated uniformly on the frequency spectrum. All the parameters of the simulation are loaded from the `ToyConfiguration.txt` file. The parameters are chosen to reproduce the runs 4b.

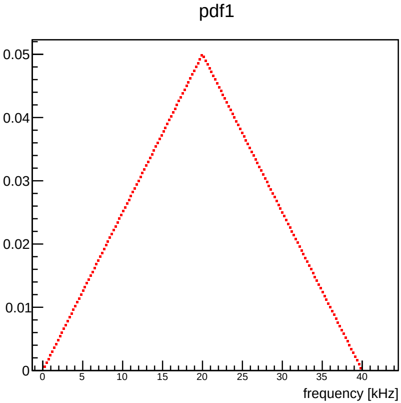


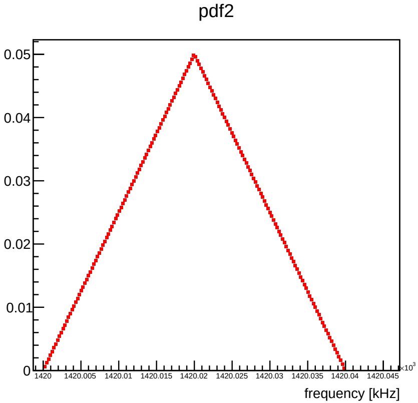
```
Nstack = 20
NHbar = 14
Repetition = 5
# cosmic rate is expressed in event/second
CosmicRate = 0.051028571
Efficiency = 1
pwall_cb = 1
pwall_ad = 1
C = 0.5
FrequencyStep = 5
TimeStep = 8
SweepStep = 24
# The following are in kHz units
x_cb_start = 0
x_cb_end = 40
x_cb_peak = 20
x_da_start = 1420000
x_da_peak = 1420020
x_da_end = 1420040
delay = 2.5
```

Figure: Parameters of the Toy.

Triangular Line-shape Pdfs

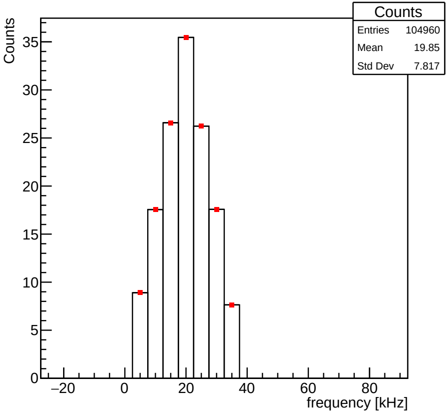
For this first use of the toy, we have chosen simple line-shapes, triangular with a symmetric rise and fall.

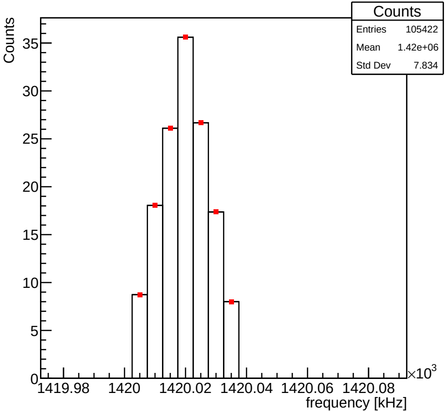




Triangular Line-shapes Simulation

We sample at the given frequency step of 5 kHz the Pdfs, to simulate the experimental line-shapes. We applied the onset finding algorithm to this distribution.



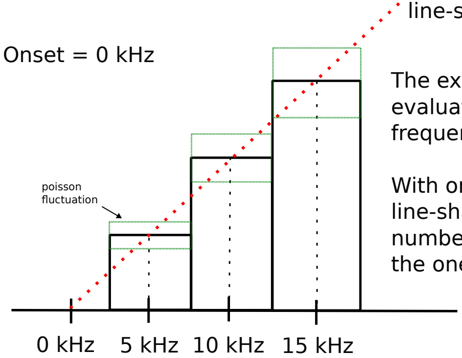


The onset is fixed ad $f = 0$ kHz. In this sample the cosmic background is set to zero.

A simple Onset finding Algorithm

The first algorithm that is tested is quite simple: **the onset is identified by the first bin with a content over a given threshold ($> N\mu_{cosmic}$)** [1]

Before showing the plot with the simulated data, it is useful to remind how this algorithm deals with the frequency step:



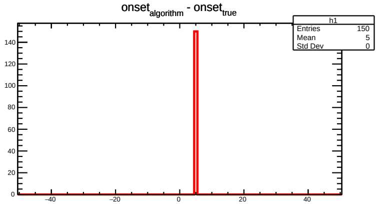
The expected bin content is given evaluating the lineShape at each frequency step (+ 5 kHz, +10kHz ...).

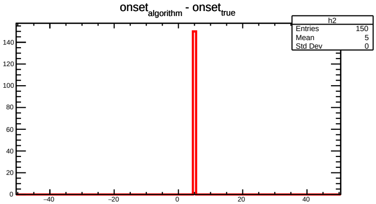
With onset fixed at 0, and a linear line-shape, the first bin with an expected number of counts different from 0 will be the one at 5 kHz (bias).

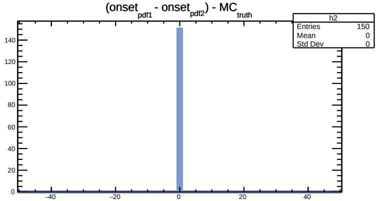
[1] Where the μ_{cosmic} is computed from the Poisson distribution of the cosmic counts expected per bin.

Consistency Check 1

We have tested the algorithm with a dataset without cosmic background and delay fixed to zero. The algorithm identifies the onset at frequency 5 kHz.

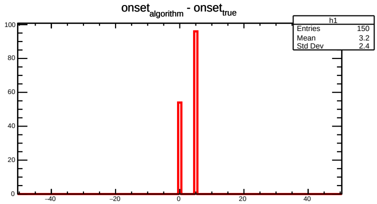




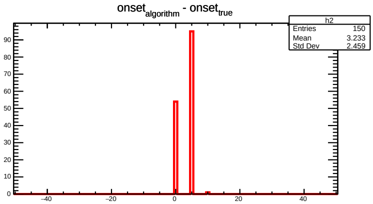


Consistency Check 2

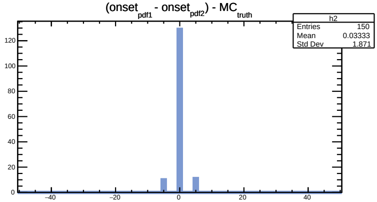
We have tested the algorithm with a dataset without cosmic background. The delay is uniform distributed in $-2.5\,\text{kHz}$ and $2.5\,\text{kHz}$.



$\text{onset}_{\text{algorithm}}$ - $\text{onset}_{\text{true}}$



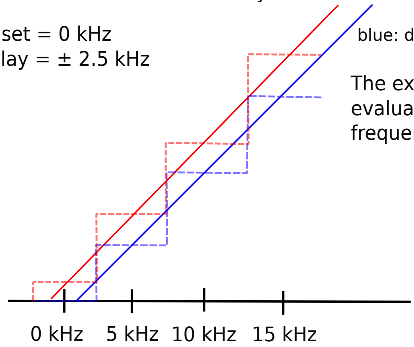
$\text{onset}_{\text{algorithm}}$ - $\text{onset}_{\text{true}}$



$(\text{onset}_{\text{pdf1}}$ - $\text{onset}_{\text{pdf2}})$ - MC_{truth}

Consistency Check 2

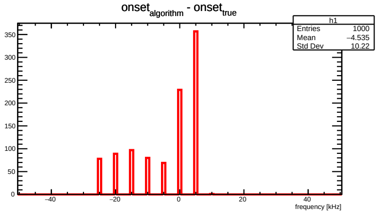
With the delay, two bins (*frequency* = 0 kHz and *frequency* = 5 kHz) are populated.

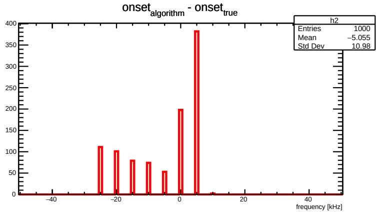


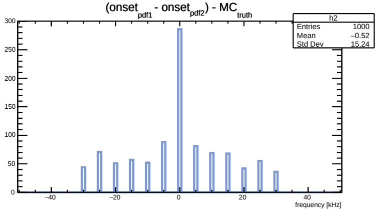
The expected bin content is given evaluating the lineShape at each frequency step (+ 5 kHz, +10kHz ...).

algorithm test: threshold $> 3\mu_{cosmic}$

In this case, we have applied the algorithm to simulated data with cosmic background (using a rate of $0.051\,\frac{event}{s}$, from passcut1). Each bin has an expected cosmic background of $dwelltime \cdot rate = 0.408$. The delay is uniform distributed in $-2.5\,\text{kHz}$ and $2.5\,\text{kHz}$.

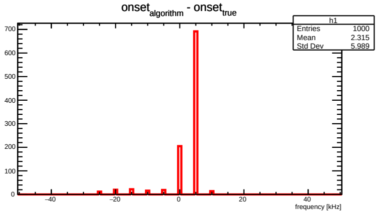




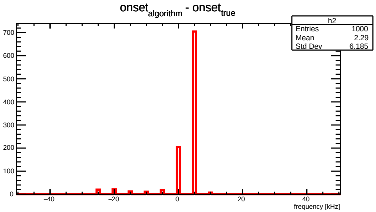


algorithm test: threshold $> 5\mu_{cosmic}$

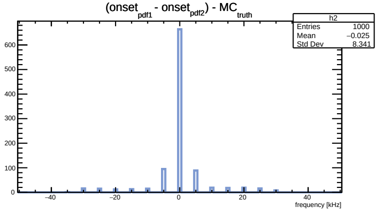
In this case, we have applied the algorithm to simulated data with cosmic background (using a rate of $0.051\,\frac{event}{s}$, from passcut1). Each bin has an expected cosmic content of $dwelltime \cdot rate = 0.408$. The delay is uniform distributed in -2.5 kHz and 2.5 kHz.



$onset_{algorithm}$ - $onset_{true}$



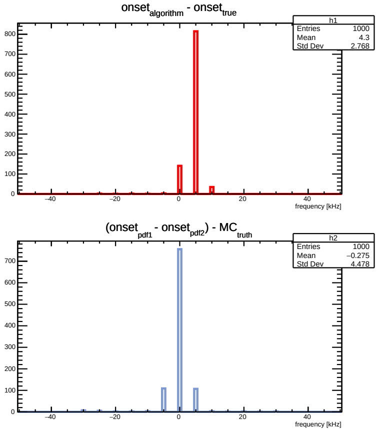
$onset_{algorithm}$ - $onset_{true}$



$(onset_{pdf1}$ - $onset_{pdf2})$ - MC_{truth}

algorithm test: threshold $> 8\mu_{cosmic}$

In this case, we have applied the algorithm to simulated data with cosmic background (using a rate of $0.051\frac{event}{s}$, from passcut1). Each bin has an expected cosmic content of $dwelltime \cdot rate = 0.408$. The delay is uniform distributed in $-2.5\,\text{kHz}$ and $2.5\,\text{kHz}$.



Next step

- different line-shapes (e.g. quadratic rise, etc.)
- different onset-finding algorithm
- simulation of repetition/runs with Bfield drift.

Improvements of the week (24/11/30 - 24/12/07)

- Implemented the onset-finding algorithm of 2017 ($\textit{first} > 0, \textit{second} > 1$)
- Simulation with a lineShape following the `run 69373` (lineShape with high statistics).
- Implementation and test of different onset finding algorithms.

Fit to the data of run 69373

The lineShape is fitted using a Cruijff function, which takes into account the asymmetry of the left-right tails, $(model = N \cdot exp(\frac{-(x-x_0)^2}{2\sigma_{0,1}+k_{0,1}(x-x_0)^2}))$

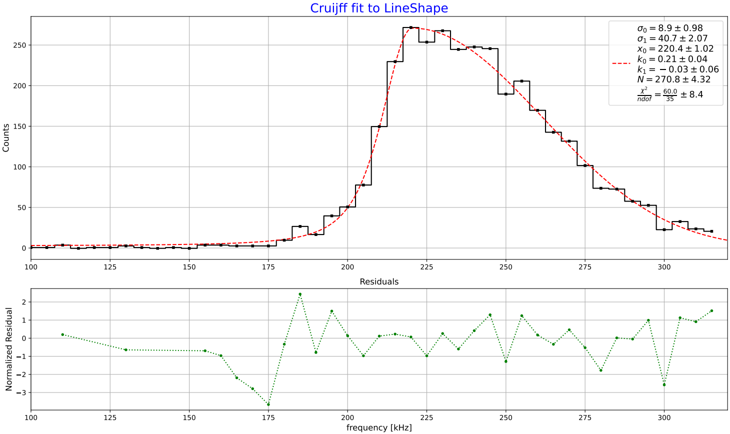


Figure: On top plot, the black line represents data and the red line the fit with the Cruijff function.

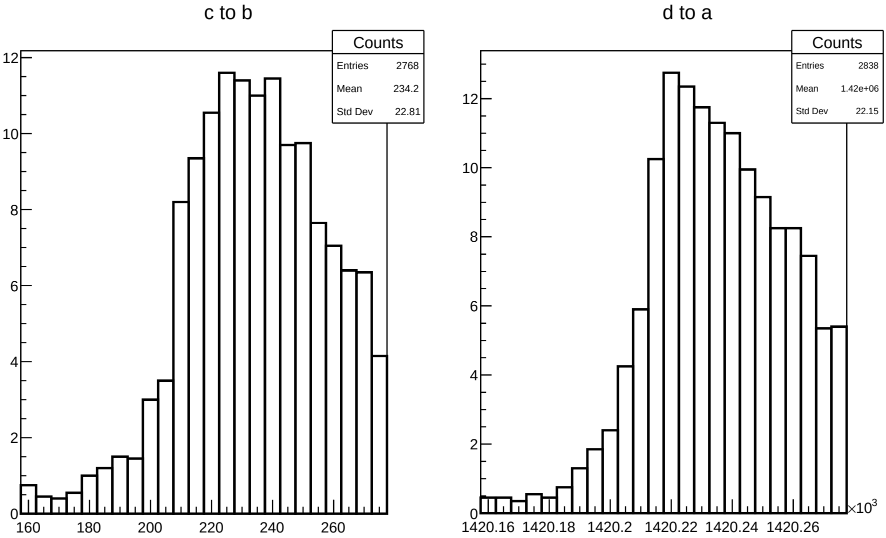
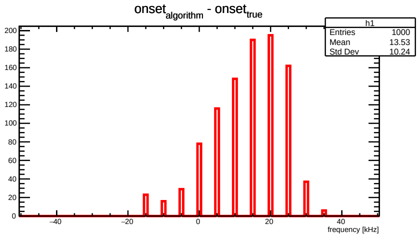
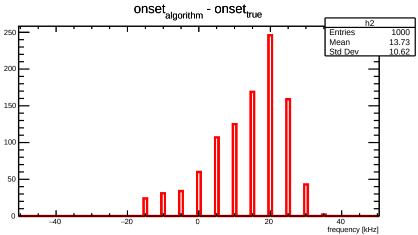
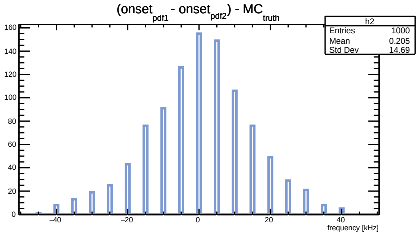


Figure: Histogram of the generated lineshape for 20 run. The Lineshape is scaled by a factor of 20.

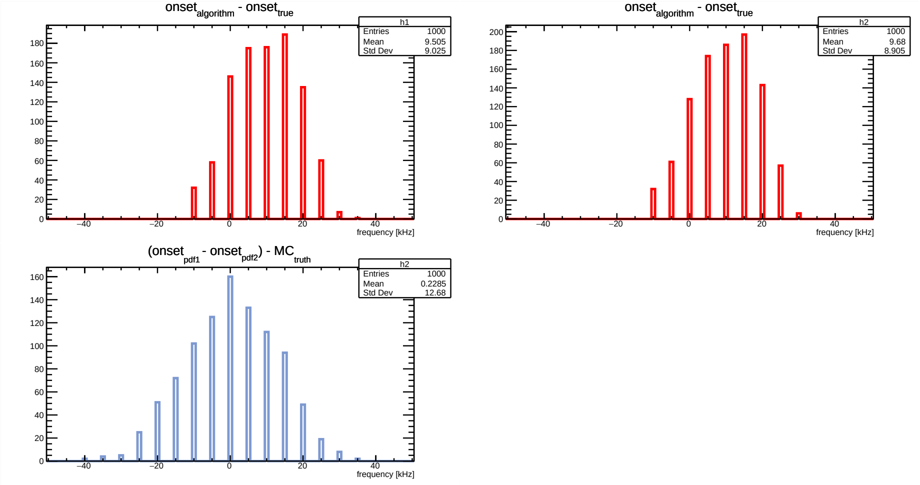
test 2017 algorithm







test 2017 algorithm (reversed)


onset_algorithm - onset_true
h1
Entries 1000
Mean 9.505
Std Dev 9.025
frequency [kHz]
onset_algorithm - onset_true
h2
Entries 1000
Mean 9.68
Std Dev 8.905
frequency [kHz]
(onset_pdf1 - onset_pdf2) - MC_truth
h2
Entries 1000
Mean 0.2285
Std Dev 12.68
frequency [kHz]