1-high-level-processing

June 24, 2020

1 HIGH LEVEL ANALYSIS

[1]:
```
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
from scipy import stats
import straxen
from multihist import Histdd, Hist1d

# Print out exactly what versions we are using, for reference / troubleshooting:
import sys
import os.path as osp
print(f"Python {sys.version} at {sys.executable}\n"
      f"Straxen {straxen.__version__} at {osp.dirname(straxen.__file__)}")
```

```
Python 3.6.10 |Anaconda, Inc.| (default, Mar 25 2020, 23:51:54)
[GCC 7.3.0] at /opt/XENONnT/anaconda/envs/XENONnT_development/bin/python
Straxen 0.9.0 at
/opt/XENONnT/anaconda/envs/XENONnT_development/lib/python3.6/site-
packages/straxen
```

[2]:
```
st = straxen.contexts.xenon1t_dali()
run_id = '180215_1029'
```

[3]:
```
peaks = st.get_array(run_id,['peaks','peak_basics'])
```

2 PEAK PROCESSING: peak_processing.py

2.1 Peak Basics

Stiamo guardando il codice del plugin peak_processing

Domanda sulla funzione compute, prende dei piccoli chunck di picchi e prende peaks che è un array, compute_center_times si potrebbe migliorare, si potrebbe implementare diversi modi per calcolarlo.

[4]:
```
#Per vedere le opzioni dei picchi:
st.show_config('peak_basics')
```

[4]:

	option	default
0	n_top_pmts	253
1	diagnose_sorting	False
2	peaklet_gap_threshold	350
3	peak_left_extension	30
4	peak_right_extension	
5	peak_min_pmts	
6	peak_split_gof_threshold	
7	peak_split_filter_wing_width	
8	peak_split_min_area	
9	peak_split_iterations	
10	diagnose_sorting	
11	gain_model	
12	tight_coincidence_window_left	
13	tight_coincidence_window_right	
14	n_tpc_pmts	
15	hit_min_amplitude	
16	hev_gain_model	
17	baseline_samples	
18	tail_veto_threshold	
19	tail_veto_duration	
20	tail_veto_resolution	
21	tail_veto_pass_fraction	
22	tail_veto_pass_extend	
23	pmt_pulse_filter	
24	save_outside_hits	
25	n_tpc_pmts	
26	check_raw_record_overlaps	
27	allow_sloppy_chunking	
28	hit_min_amplitude	
29	pax_raw_dir	
30	stop_after_zips	
31	events_per_chunk	
32	samples_per_record	
33	s1_max_rise_time	
34	s1_max_rise_time_post100	
35	s1_min_coincidence	
36	s2_min_pmts	
37	s2_merge_max_area	
38	s2_merge_max_gap	
39	s2_merge_max_duration	

```
4                                                  200
5                                                    4
6   (None, ((0.5, 1), (3.5, 0.25)), ((2, 1), (4.5,…
7                                                   70
8                                                   40
9                                                   20
10                                               False
11                                             <OMITTED>
12                                                  50
13                                                  50
14                                             <OMITTED>
15                             pmt_commissioning_initial
16                                       (disabled, None)
17                                                  40
18                                                   0
19                                            3000000
20                                               1000
21                                               0.05
22                                                  3
23                                               None
24                                             (3, 20)
25                                             <OMITTED>
26                                                  True
27                                                False
28                             pmt_commissioning_initial
29                                        /data/xenon/raw
30                                                   0
31                                                  50
32                                                 110
33                                                  60
34                                                 150
35                                                   3
36                                                   4
37                                                5000
38                                                3500
39                                               15000

                                                current  \
0                                                   127
1                                              <OMITTED>
2                                              <OMITTED>
3                                              <OMITTED>
4                                                   30
5                                                    2
6                                              <OMITTED>
7                                              <OMITTED>
8                                              <OMITTED>
```

```
9                                          <OMITTED>
10                                         <OMITTED>
11  (to_pe_per_run, https://raw.githubusercontent…
12                                         <OMITTED>
13                                         <OMITTED>
14                                               248
15                                       XENON1T_SR1
16  (to_pe_per_run, https://raw.githubusercontent…
17                                         <OMITTED>
18                                            100000
19                                         <OMITTED>
20                                         <OMITTED>
21                                         <OMITTED>
22                                         <OMITTED>
23  (0.012, -0.119, 2.435, -1.271, 0.357, -0.174, …
24                                             (3, 3)
25                                               248
26                                             False
27                                              True
28                                       XENON1T_SR1
29                                         <OMITTED>
30                                         <OMITTED>
31                                         <OMITTED>
32                                         <OMITTED>
33                                         <OMITTED>
34                                         <OMITTED>
35                                         <OMITTED>
36                                         <OMITTED>
37                                         <OMITTED>
38                                         <OMITTED>
39                                         <OMITTED>

                            applies_to  \
0                         (peak_basics,)
1                               (peaks,)
2                   (peaklets, lone_hits)
3                   (peaklets, lone_hits)
4                   (peaklets, lone_hits)
5                   (peaklets, lone_hits)
6                   (peaklets, lone_hits)
7                   (peaklets, lone_hits)
8                   (peaklets, lone_hits)
9                   (peaklets, lone_hits)
10                  (peaklets, lone_hits)
11                  (peaklets, lone_hits)
12                  (peaklets, lone_hits)
13                  (peaklets, lone_hits)
```

```
14                  (peaklets, lone_hits)
15                  (peaklets, lone_hits)
16  (records, veto_regions, pulse_counts)
17  (records, veto_regions, pulse_counts)
18  (records, veto_regions, pulse_counts)
19  (records, veto_regions, pulse_counts)
20  (records, veto_regions, pulse_counts)
21  (records, veto_regions, pulse_counts)
22  (records, veto_regions, pulse_counts)
23  (records, veto_regions, pulse_counts)
24  (records, veto_regions, pulse_counts)
25  (records, veto_regions, pulse_counts)
26  (records, veto_regions, pulse_counts)
27  (records, veto_regions, pulse_counts)
28  (records, veto_regions, pulse_counts)
29                        (raw_records,)
30                        (raw_records,)
31                        (raw_records,)
32                        (raw_records,)
33              (peaklet_classification,)
34              (peaklet_classification,)
35              (peaklet_classification,)
36              (peaklet_classification,)
37                         (merged_s2s,)
38                         (merged_s2s,)
39                         (merged_s2s,)
```

```
                                                help
0                                  Number of top PMTs
1   Enable runtime checks for sorting and disjoint…
2        No hits for this many ns triggers a new peak
3          Include this many ns left of hits in peaks
4         Include this many ns right of hits in peaks
5   Minifnmum contributing PMTs needed to define a…
6   Natural breaks goodness of fit/split threshold…
7   Wing width of moving average filter for low-sp…
8   Minimum area to evaluate natural breaks criter…
9      Maximum number of recursive peak splits to do.
10  Enable runtime checks for sorting and disjoint…
11  PMT gain model. Specify as (model_type, model_…
12  Time range left of peak center to call a hit a…
13  Time range right of peak center to call a hit …
14                                 Number of TPC PMTs
15  Minimum hit amplitude in ADC counts above base…
16  PMT gain model used in the software high-energ…
17  Number of samples to use at the start of the p…
18  Minimum peakarea in PE to trigger tail veto.Se…
```

```
19              Time in ns to veto after large peaks
20  Time resolution in ns for pass-veto waveform s…
21  Pass veto if maximum amplitude above max * fra…
22  Extend pass veto by this many samples (tail_ve…
23  Linear filter to apply to pulses, will be norm…
24  Save (left, right) samples besides hits; cut t…
25                                 Number of TPC PMTs
26  Crash if any of the pulses in raw_records over…
27  Use a default baseline for incorrectly chunked…
28  Minimum hit amplitude in ADC counts above base…
29                    Directory with raw pax datasets
30         Convert only this many zip files. 0 = all.
31                Number of events to yield per chunk
32                      Number of samples per record
33            Maximum S1 rise time for < 100 PE [ns]
34            Maximum S1 rise time for > 100 PE [ns]
35  Minimum tight coincidence necessary to make an S1
36       Minimum number of PMTs contributing to an S2
37     Merge peaklet cluster only if area < this [PE]
38  Maximum separation between peaklets to allow m…
39  Do not merge peaklets at all if the result wou…
```

[5]:
```
st.data_info('peak_basics')
```

[5]:
```
            Field name Data type  \
0                 time     int64
1              endtime     int64
2          center_time     int64
3                 area   float32
4           n_channels     int16
5              max_pmt     int16
6        max_pmt_area   float32
7      range_50p_area   float32
8      range_90p_area   float32
9   area_fraction_top   float32
10              length     int32
11                  dt     int16
12          rise_time   float32
13  tight_coincidence     int16
14                type      int8

                                            Comment
0         Start time of the peak (ns since unix epoch)
1           End time of the peak (ns since unix epoch)
2   Weighted center time of the peak (ns since uni…
3                                 Peak integral in PE
4             Number of PMTs contributing to the peak
```

```
5            PMT number which contributes the most PE
6   Area of signal in the largest-contributing PMT…
7   Width (in ns) of the central 50% area of the peak
8   Width (in ns) of the central 90% area of the peak
9   Fraction of area seen by the top array (NaN fo…
10             Length of the peak waveform in samples
11         Time resolution of the peak waveform in ns
12       Time between 10% and 50% area quantiles [ns]
13                    Hits within tight range of mean
14                    Classification of the peak(let)
```

Calcolo di **range_50p_area** e **range_90p_area**, viene presa la larghezza del picco e poi usato :,

[6]:
```
print(peaks['width'][:,5])
print(peaks['width'][:,9])
```

```
[196.87552  83.1763  231.50201 … 383.8594  268.03873 167.85959]
[ 208.07713   97.32471  246.52963 … 3063.1973   275.9225   195.40831]
```

Calcolo di **area_fraction_top**

[7]:
```
n_top = st.config['n_top_pmts']
area_top = peaks['area_per_channel'][:, :n_top].sum(axis=1)
m = peaks['area'] > 0
#r['area_fraction_top'][m] =
area_top[m]/peaks['area'][m]
```

[7]:
```
array([0.4595095 , 0.36514154, 0.        , …, 0.4225748 , 1.        ,
       1.        ], dtype=float32)
```

Calcolo di **rise_time**

[8]:
```
-peaks['area_decile_from_midpoint'][:,1]
```

[8]:
```
array([ 14.280781,  86.96509 ,  16.90953 , …, 234.96863 ,  39.32416 ,
       186.01207 ], dtype=float32)
```

2.2 Peak Positions

La ricostruzione della posizione, prende la somma totale di più PMT inseme, ma non considera la posizione dei singoli PMT.

Al momento la posizione dei picchi è ricostruita usando un solo metodo, si può introdurre un nuovo algoritmo per fare questo.

[9]:
```
st.data_info('peak_positions')
```

[9]:
```
  Field name Data type                                         Comment
0          x   float32  Reconstructed S2 X position (cm), uncorrected
1          y   float32  Reconstructed S2 Y position (cm), uncorrected
2       time     int64               Start time since unix epoch [ns]
3    endtime     int64       Exclusive end time since unix epoch [ns]
```

[10]:
```
st.register(straxen.plugins.peak_processing.PeakPositions)
```

[10]:
```
straxen.plugins.peak_processing.PeakPositions
```

Viene caricato un file json dove sono inseriti i PMT non funzionanti, poi vengono presi soltanto i PMT che hanno un'area del picco maggiore di un certo valore definendo un mask e poi viene usata la funzione predict per trovare la posizione.

2.3 Peak Proximity

L'obiettivo è quello di controllare picchi nelle vicinanze, si può cambiare la finestra che si guarda con get_window_size con OverlapWindowPlugin, guardare qui per più dettagli https://strax.readthedocs.io/en/latest/developer/overlaps.html#overlap-window-plugins.

[11]:
```
st.data_info('peak_proximity')
```

[11]:
```
           Field name Data type  \
0        n_competing     int32
1   n_competing_left     int32
2     t_to_prev_peak     int64
3     t_to_next_peak     int64
4  t_to_nearest_peak     int64
5               time     int64
6            endtime     int64

                                          Comment
0  Number of nearby larger or slightly smaller peaks
1  Number of larger or slightly smaller peaks lef…
2  Time between end of previous peak and start of…
3  Time between end of this peak and start of nex…
4  Smaller of t_to_prev_peak and t_to_next_peak [ns]
5                  Start time since unix epoch [ns]
6          Exclusive end time since unix epoch [ns]
```

3 EVENT PROCESSING: event_processing.py

Ci sono 5 classi nel codice e alla fine c'è **Event Info** con : 1. Events 2. EventsBasics 3. EventsPositions 4. CorrectedAreas 5. EnergyEstimate

3.1 Events

Dipende da peak_basics e peak_proximity. Ci sono argomenti start e end, che se usati vengono presi come intervalli nel quale cercare gli eventi (può essere utile se ad esempio ci sono dei periodi in cui non ho dati).

[12]:
```
st.data_info('events')
```

[12]:

	Field name	Data type	Comment
0	event_number	int64	Event number in this dataset
1	time	int64	Event start time in ns since the unix epoch
2	endtime	int64	Event end time in ns since the unix epoch

Il trigger è cercato con un minimo (100) è un massimo.

3.2 Events Basics

Dipende da events, peak_basics, peak_positions, peak_proximity e calcola le proprietà di base per ogni evento:

[13]:
```
st.data_info('event_basics')
```

[13]:

	Field name	Data type
0	time	int64
1	endtime	int64
2	n_peaks	int32
3	drift_time	int32
4	s1_index	int32
5	alt_s1_index	int32
6	s1_time	int64
7	alt_s1_time	int64
8	s1_center_time	int64
9	alt_s1_center_time	int64
10	s1_endtime	int64
11	alt_s1_endtime	int64
12	s1_area	float32
13	alt_s1_area	float32
14	s1_n_channels	int32
15	alt_s1_n_channels	int32
16	s1_n_competing	float32
17	alt_s1_n_competing	float32
18	s1_range_50p_area	float32
19	alt_s1_range_50p_area	float32
20	s1_area_fraction_top	float32
21	alt_s1_area_fraction_top	float32
22	alt_s1_interaction_drift_time	int32
23	alt_s1_delay	int32
24	s2_index	int32
25	alt_s2_index	int32

26	s2_time	int64
27	alt_s2_time	int64
28	s2_center_time	int64
29	alt_s2_center_time	int64
30	s2_endtime	int64
31	alt_s2_endtime	int64
32	s2_area	float32
33	alt_s2_area	float32
34	s2_n_channels	int32
35	alt_s2_n_channels	int32
36	s2_n_competing	float32
37	alt_s2_n_competing	float32
38	s2_range_50p_area	float32
39	alt_s2_range_50p_area	float32
40	s2_area_fraction_top	float32
41	alt_s2_area_fraction_top	float32
42	alt_s2_interaction_drift_time	int32
43	alt_s2_delay	int32
44	s2_x	float32
45	s2_y	float32
46	alt_s2_x	float32
47	alt_s2_y	float32

	Comment
0	Start time since unix epoch [ns]
1	Exclusive end time since unix epoch [ns]
2	Number of peaks in the event
3	Drift time between main S1 and S2 in ns
4	Main S1 peak index in event
5	Alternate S1 peak index in event
6	Main S1 start time since unix epoch [ns]
7	Alternate S1 start time since unix epoch [ns]
8	Main S1 weighted center time since unix epoch …
9	Alternate S1 weighted center time since unix e…
10	Main S1 end time since unix epoch [ns]
11	Alternate S1 end time since unix epoch [ns]
12	Main S1 area, uncorrected [PE]
13	Alternate S1 area, uncorrected [PE]
14	Main S1 count of contributing PMTs
15	Alternate S1 count of contributing PMTs
16	Main S1 number of competing PMTs
17	Alternate S1 number of competing PMTs
18	Main S1 width, 50% area [ns]
19	Alternate S1 width, 50% area [ns]
20	Main S1 fraction of area seen by the top PMT a…
21	Alternate S1 fraction of area seen by the top …
22	Drift time using alternate S1 [ns]

```
23            Time between main and alternate S1 [ns]
24                         Main S2 peak index in event
25                    Alternate S2 peak index in event
26           Main S2 start time since unix epoch [ns]
27      Alternate S2 start time since unix epoch [ns]
28  Main S2 weighted center time since unix epoch …
29  Alternate S2 weighted center time since unix e…
30             Main S2 end time since unix epoch [ns]
31        Alternate S2 end time since unix epoch [ns]
32                       Main S2 area, uncorrected [PE]
33                  Alternate S2 area, uncorrected [PE]
34                  Main S2 count of contributing PMTs
35             Alternate S2 count of contributing PMTs
36                    Main S2 number of competing PMTs
37               Alternate S2 number of competing PMTs
38                        Main S2 width, 50% area [ns]
39                   Alternate S2 width, 50% area [ns]
40  Main S2 fraction of area seen by the top PMT a…
41  Alternate S2 fraction of area seen by the top …
42                  Drift time using alternate S2 [ns]
43            Time between main and alternate S2 [ns]
44  Main S2 reconstructed X position, uncorrected …
45  Main S2 reconstructed Y position, uncorrected …
46  Alternate S2 reconstructed X position, uncorre…
47  Alternate S2 reconstructed Y position, uncorre…
```

Calcola le proprietà facendo un loop sui 2 picchi (S1 e S2), c'è anche la possibilità di fare il loop

[ ]:
```
runs = st.select_runs(run_mode='kr*', available='event_info')
run_id = '181027_2044'
st.is_stored(run_id,'event_info')
events = st.get_df(run_info['name'], 'event_info')
events = st.get_df(run_info['name'], 'event_info')
```

[ ]:
```

```