

# WPJET4 Gamma Camera Upgrade (GCU) WPJET4 Gamma Spectrometer Upgrade (GSU)

<b>Additional report</b>	<b>Software utilities for offline pile-up recovery for Gamma Camera and Gamma Spectrometers</b>
--------------------------	---

## 1. Introduction

### 1.1 Glossary

**Waveform** – a digital representation of an analog signal as read out from a radiation detector by digital data acquisition (DAQ) system. It is an array of integer numbers corresponding to subsequent instantaneous values of signal amplitude as sampled by the analog-to-digital converter.

**Sample** – value of the analog signal amplitude at a given instant of time, or more precisely, amplitude integrated over short period of time from  $t$  to  $t+dt$ , where  $dt$  is short in comparison to signal duration.

**Sampling frequency** – frequency at which the digital DAQ is digitizing (sampling) the incoming analog signals. The sampling frequency determines the sample duration. For example, at 200 MHz sampling frequency each sample is 5 ns long (i.e., each sample represents the signal amplitude integrated over  $dt = 5$  ns).

### 1.2 Signal pile-up

Electric signal from a radiation detector produced as a result of energy deposition by radiation particle has a certain shape and duration as illustrated in Figure 1 (left). The shape and duration of the signal is characteristic of the radiation type, detector type and electronic signal processing chain while the maximum amplitude as well as the area under the signal are generally proportional to energy deposited in the detector. As seen in Figure 1 (left), a signal produced by the scintillation detector does not have a symmetric, Gaussian-like shape but is rather characterized by a fast rising edge and a longer falling edge. Fortunately, the knowledge of an exact mathematical formula describing the shape of the signal is not needed as the characteristic signal shape can be precisely established from simple measurements.

If a second radiation particle hits the detector within the duration of the signal induced by a preceding particle then this second signal will add to the first one as illustrated in Figure 1 (right). This phenomenon is called “pile-up”.

While the resulting signal is an arithmetic sum of two individual signals separated in time by an offset, it is no longer simple to deduce the energy of contributing radiation particles. Thus, pile-up events need to be identified. Once identified such an event may be rejected what would, however, lead to loss of data. Alternatively, one may try to recover pile-up event, i.e., identify individual overlapping signals and extract the information carried by each one.

Here we report on software utilities developed for off-line pile-up recovery for gamma-ray

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	1 of 21

diagnostics.

The implemented algorithm is described in more detail in Appendix 1. Generally speaking, it belongs to a class of pile-up recovery algorithms that exploit first-order derivative method to search for signal candidates and least-squares fitting using a known reference signal to recover individual signal candidates. The algorithm is somewhat related to least-squares fit algorithms described in e.g., [1, 2], however, it is more robust. In contrast to algorithms discussed in the cited literature the algorithm developed here is able to identify, and to a large extent recover, events that are not recognized as pile-up by the first-order derivative method (see Sec. 4).

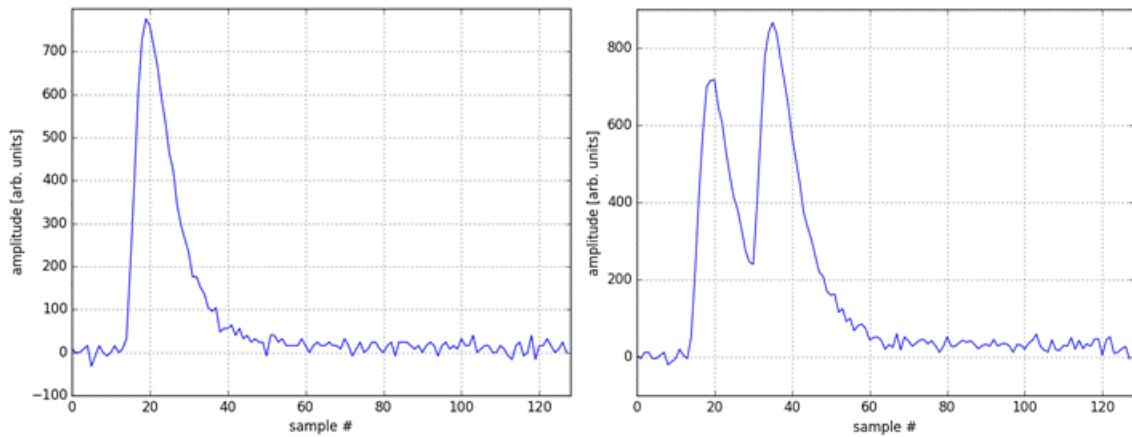


Figure 1. Example of a waveform of a single signal as produced by an individual gamma ray, i.e., without a pile-up (left) and an example of pile-up of two signals (right).

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	2 of 21

## 2. The software utilities

Production versions of the pile-up recovery utilities are in the directory /work/padrich/pileup\_recovery/ on the JET analysis cluster.

All programs are written in Python and **require python 3** to run.

### 2.1 The main program

The main pile-up recovery utility is the program: **pileup\_recovery.py**

The **mandatory arguments** are:

- shot** JET pulse number (e.g. 95173),
- diag** the diagnostic name (allowed are: 'KN3G', 'KM6S', 'KM6T', 'kn3g', 'km6s', 'km6t'),
- det** detector number within the diagnostic,
- pwidth** number of samples in the waveform (the “PWIDTH” parameter of the DAQ. E.g., 128),
- TimeThr** the time threshold in seconds. Events that have timestamp corresponding to  $\text{time} > \text{time threshold}$  are skipped,
- pattern** full path to the file with a data describing the reference signal (see Sec. 2.2 below for description of an utility to produce the reference signal),
- output\_dir** path to a directory where results will be written (e.g., /work/username).

Example of the command to run the pile-up recovery from the command line for the Gamma Camera detector 5 and JET pulse #95173 is:

```
python /work/padrich/pileup-recovery/pileup_recovery.py -shot 95173 -diag KN3G  
-det 5 -pwidth 128 -TimeThr 55  
-pattern /work/padrich/shot_9999027_ch5_pattern.txt -output_dir /work/padrich
```

The program accesses the appropriate JPF data using the [GETDAT API](#). The settings of the DAQ, such as sampling frequency, acquisition threshold and pretrigger time are retrieved using the [SAL API](#). It is foreseen to unify all data access using SAL as soon as SAL server functionality is extended to reliably handle the transfer of large data sets<sup>1</sup>. This future extension will allow using the pile-up recovery program also outside the JET computing cluster.

The program sequentially processes all events that occurred at  $\text{time} < \text{TimeThr}$ . For each event a number N waveforms corresponding to recovered individual gamma-rays are produced. N can be 0 or positive integer and depends on how many gamma-rays were detected and recovered for that event. All recovered waveforms are stored in a binary segmented mode file.

---

<sup>1</sup> <https://git.ccf.ac.uk/simple-access-layer/jet-sal-server/issues/15>

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	3 of 21

This file is the primary output of the program. This file has exactly the same data structure as the original JPF and thus may be further processed with preexisting user programs for processing binary segmented mode data. It is written in a directory specified on input with the `-output_dir` argument. The file naming convention is detailed further below.

In addition, a second binary segmented mode file is produced that contains rejected events and, in case of fully or partly recovered events, residues<sup>2</sup> remaining after pile-up recovery. This secondary output file also has the same data structure as the segmented binary mode files produced by the DAQ. It may be useful for educational purposes and for diagnostic of pile-up recovery.

For control, diagnostic and education purposes two additional files are produced. One is a plain ascii text file that contains event-by-event list of recovered energies and times as well as a handful of useful diagnostic information. Structure of this text output file is detailed further below.

Second is a graphics file in the png format that contains plots of:

- initially estimated event multiplicity<sup>2</sup> and actual number of reconstructed gamma-rays vs. event ID,
- baseline level of accepted event vs. event ID,
- calibrated time of recovered gamma-ray vs. ID of the recovered gamma-ray,
- histogram of uncalibrated gamma-ray energy resulting from pile-up recovery,
- histogram of Pearson's chi-squared of all the fits performed<sup>2</sup>,
- histogram of Pearson's p-value of all the fits performed<sup>2</sup>.

These plots are useful to get a quick overview of the course and results of pile-up recovery.

### Naming convention of the output files

Names of the output files start with a sequence `[shot]_[diag]_Det_[det]` followed by, respectively:

1. `“-pileup_recovered.bin”`
2. `“-pileup_residues_and_rejected.bin”`
3. `“-pileup_recovery_diagnostics.txt”`
4. `“-pileup_recovery_summary_plot.png”`

For the example mentioned above, the output files would be:

```
95173_KN3G_Det_5-pileup_recovered.bin
95173_KN3G_Det_5-pileup_residues_and_rejected.bin
95173_KN3G_Det_5-pileup_recovery_diagnostics.txt
95173_KN3G_Det_5-pileup_recovery_summary_plot.png
```

An example of the graphics output file is shown in Figure 2.

<sup>2</sup> See Appendix 1

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	4 of 21

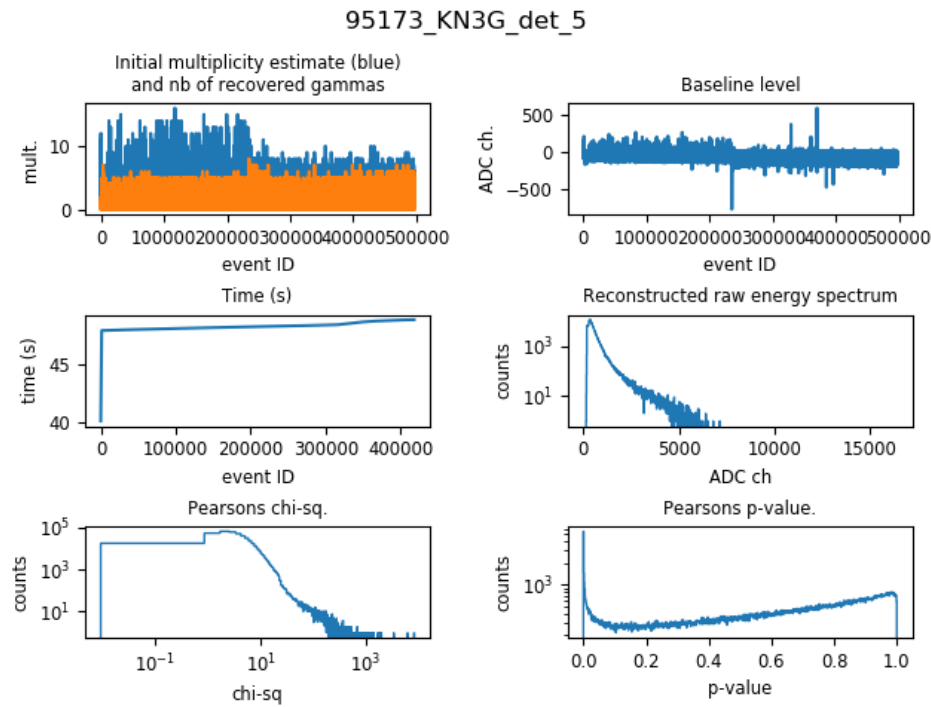


Figure 2. Example of a summary plot produced by `pileup_recovery.py` for KN3G detector no. 5 and JET pulse no. 95173

### Structure of the text output file

The structure of the text output file is illustrated with an example in Figure 3. The file starts with a brief preamble (lines that begin with #) containing explanatory notes on the structure of the data record and on meaning of recovery status codes (described in detail below). Data records follow the preamble.

Each data record corresponds to a single event from the input JPF. The number of records is equal to the number of processed events from the input JPF. Records are separated with newline. Each record begins with an event ID that is the sequence number of the event in the JPF input file. The event ID is followed by:

- calibrated event time (in seconds),
- maximum waveform amplitude before baseline restoration,
- calculated baseline level,
- standard deviation of the calculated baseline level,
- recovery status code,
- initial multiplicity estimated by means of counting the number of times the waveform's first-order derivative changes sign from positive to negative,
- number of recovered gamma-rays (can differ from the initial multiplicity estimate),
- IDs of all recovered gamma-rays – those correspond to event sequence numbers in the first binary output file (i.e., the one to which the waveforms of individual recovered gamma-rays are written),
- uncalibrated energies of all recovered gamma-rays,
- offsets (in number of samples; may be fractional) of the individual recovered

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	5 of 21

- gamma-ray waveforms within the original unprocessed waveform,
- calibrated time of individual recovered gamma-rays (in seconds). Time stamp corresponding to the value of the calibrated time is written in the first binary output file along the individual waveform.

Recovery status codes are:

- 0 - Full recovery. An event was processed and it looks like all gamma-rays were identified and successfully recovered.
- 1 - Event rejected without attempting a recovery. Rejection caused by inability to reliably establish the baseline level (i.e., standard deviation of baseline samples is above a predefined threshold). This is the most frequent reason for event rejection. In most cases the problem is due to a tail of previous signal or a small<sup>3</sup> signal present at the beginning of a frame that obscures baseline restoration.
- 2 - Event recovery attempted but broken at some point due to poor quality of a fit (i.e., Pearson's p-value below predefined threshold).
- 3 - Event recovery attempted but broken at some point due to occurrence of negative amplitudes within a fit range. This occurs very rarely but was observed to result in run-time errors in the fitting routine.
- 4 - Event recovery attempted but broken at some point due to occurrence of negative chi-squared. Although this sounds like an impossibility, there is a known bug in `scipy.stats.chisquare()` function that in certain, extremely rare, circumstances produces this error.
- 5 - Event recovery attempted but broken at some point due to a runtime error in `scipy.optimize.curve_fit()`. The `curve_fit()` function of the `scipy` library is used to fit the reference signal against the measured signal. On rare occasions the function returns with an error status.
- 6 - Event recovery attempted but broken at some point due to undetermined fit range. This happens when there are more than two signals overlapping within a very narrow time window.
- 7 - Event rejected without attempting a recovery because all amplitudes in the waveform happened to be below the threshold once the baseline was restored back to 0. There is a tiny fraction of odd events that are rejected because of this.
- 8 - Event rejected because at the end of, otherwise successful, recovery it turned out that energies of all recovered gamma-rays are below the acquisition threshold. This is rather rare, however, happens for close enough pile-up of two low-amplitude signals. None of them would trigger the acquisition alone but due to pile-up the trigger threshold could be exceeded and hence such a useless data acquired.

---

<sup>3</sup> i.e., below the threshold

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	6 of 21

```

# Event_ID Event_Time Max_Amplitude Baseline Baseline_Std Recovery_Status Initial_Multiplicity_Estimate N_gammas_recovered ID_1 ID_2 ... E_1 E_2 ... T_offset_1 T_offset_2 ... T_1(s) T_2(s) ...
# (sec) before baseline restoration
#
#
#
# Recovery status codes are:
# 0 = full recovery
# 1 = can not reliably establish the baseline level. Event rejected.
# 2 = recovery broken due to poor quality of fit (Pearsons p-value below threshold)
# 3 = recovery broken due to occurrence of negative amplitudes within fit range
# 4 = recovery broken due to occurrence of negative chi-sq (known bug in scipy.stats.chisquare)
# 5 = runtime error in scipy.optimize.curve_fit()
# 6 = recovery broken due to undetermined fit range (e.g. more than two signals overlapping within extremely short time window)
# 7 = all amplitudes below threshold after baseline restoration. Event rejected.
# 8 = energies of all recovered gammas below threshold. Event rejected.
#
1 48.074464055 832 -77.0 13.2 0 1 1 924.3 35.6 48.074464233
2 48.082390555 344 -77.0 7.9 0 1 1 2 422.8 34.2 48.082390726
3 48.083763015 376 -82.0 11.8 0 1 1 3 459.6 34.7 48.083763189
4 48.099573185 328 -71.0 12.6 0 1 1 4 380.6 34.4 48.099573357
5 48.103174195 344 -74.0 12.2 0 1 1 5 419.2 34.2 48.103174366
6 48.107251065 368 -69.0 14.7 0 1 1 6 435.3 34.2 48.107251836
7 48.108548610 784 -77.0 11.2 0 1 1 7 770.3 35.7 48.108548789
8 48.109056240 288 -73.0 15.7 0 1 1 8 322.8 33.3 48.109056407
9 48.110626080 488 -73.0 14.7 0 1 1 9 349.5 35.1 48.110626175
10 48.111920540 256 -75.0 11.3 0 1 1 10 323.2 35.3 48.111920706
.
.
.
496474 48.990583640 416 -19.0 101.1 1 2 0
496475 48.990584645 280 13.0 55.7 1 1 0
496476 48.990585755 480 -163.0 9.7 0 3 0 418485 418486 418487 639.1 269.3 268.9 34.9 70.3 87.2 48.990585929 48.990586107 48.990586191
496477 48.990586450 248 -146.0 12.6 0 1 1 418488 345.8 32.4 48.990586612
496478 48.990587575 328 -168.0 12.8 0 1 1 418489 487.2 34.1 48.990587745
496479 48.990588960 1640 -170.0 19.7 0 1 1 0
496480 48.990589625 312 -138.0 11.1 0 1 1 418490 434.2 33.9 48.990589794
496481 48.990591805 880 -124.0 50.5 1 1 0
496482 48.990593295 1152 -163.0 10.5 0 2 3 418491 418492 418493 1096.0 688.9 673.4 35.8 104.3 106.0 48.990593474 48.990593816 48.990593825
496483 48.990594470 1224 -109.0 22.0 1 6 0

```

Figure 3. Example of the text output file (an excerpt).

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	7 of 21

The text output file ends with a summary, as illustrated with an example in Figure 4, including information on execution time as well as statistical data on pile-up recovery efficiency such as:

- total number of processed events (labeled as `Processed event count`),
- total number of fully recovered events (labeled as `Fully recovered event count`),
- total number of partly recovered events (labeled as `Partly recovered event count`),
- total number of recovered gamma-rays (labeled as `Recovered individual gammas count`),
- total number of rejected events (labeled as `Rejected event count`).

At the very end, a table is produced showing the aggregate numbers of fully and partly recovered as well as rejected events broken down to initial multiplicity estimates and recovery/rejection status codes.

An example of such a table is shown in Figure 4. Each row contains data on events of given initial gamma-ray multiplicity estimate. Data in columns include:

- the first column (labeled as `Multiplicity`) is the initial gamma-ray multiplicity estimate,
- the second column (labeled as `Nevt total`) is the total number of events that were initially identified to have gamma-ray multiplicity as indicated in the first column,
- the third column (labeled as `Nb fully recovered`) is the total number of events of the given multiplicity that were fully recovered,
- columns 4 to 8 (labeled individually with recovery status codes from 2 to 6 and with a common top label `Nb and status of partly recovered events`) hold total number of events that were partly recovered, categorized accordingly to recovery status codes that indicate the reason for ending the recovery prematurely.
- columns 9 to 16 (labeled individually with rejection status codes from 1 to 8 and with a common top label `Nb and status of rejected events`) hold total number of events that were rejected, categorized accordingly to status codes that indicate the reason for rejection. An event with recovery status between 2 and 6 is classified as rejected if no gamma-rays were recovered despite an undertaken attempt.

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	8 of 21





## Batch mode execution.

Depending on the number of events collected for a given detector during a JET pulse, the pile-up recovery may be a lengthy process, taking longer than one hour. Therefore, it is recommended to run the pile-up recovery in batch mode using the job queuing system.

For a guide on using the job queuing system at JET please consult, e.g., <http://w3.freia.hpc.l/batchjobs.html>

There are example job command files in the directory `/work/padrich/pileup-recovery`

Since the software requires python 3, it is recommended to load the appropriate python module from the level of user's `.bashrc` file (i.e., to include a command like “`module swap python/3.7`” in one's `.bashrc`). If python 3 is loaded from `.bashrc`, the batch job can be submitted with a command like:

```
llsubmit batch_pileup-95173_kn3g_5.job
```

The content of the batch command file (`batch_pileup-95173_kn3g_5.job`) in the example above is:

```
# @ executable = python
# @ arguments = /work/padrich/pileup_recovery/pileup_recovery.py -shot 95173
               -diag KN3G -det 5 -pwidth 128 -TimeThr 55 -pattern
               /work/padrich/shot_9999027_ch5_pattern.txt -output_dir /work/padrich
# @ input = /dev/null
# @ output = /work/padrich/ll_out-95173_kn3g_5.txt
# @ error = /work/padrich/ll_err-95173_kn3g_5.txt
# @ initialdir = /work/padrich/pileup_recovery
# @ notify_user = padrich
# @ notification = complete
# @ queue
```

Alternatively, if one does not want to mess with `.bashrc`, the batch job could be submitted in the same way but with a different batch command file, as in the example below:

```
llsubmit batch_pileup_script-95173_kn3g_5.job
```

while content of the batch command file (`batch_pileup_script-95173_kn3g_5.job`) is now different:

```
# @ executable = pileup_batch.sh
# @ input = /dev/null
# @ output = /work/padrich/ll_out-95173_kn3g_5.txt
# @ error = /work/padrich/ll_err-95173_kn3g_5.txt
# @ initialdir = /work/padrich/pileup_recovery
# @ notify_user = padrich
# @ notification = complete
# @ queue
```

And the executable script `pileup_batch.sh` consists of these lines:

```
export MODULEPATH=/usr/local/modules/default
```

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	10 of 21

```
source /etc/profile.d/modules.sh
module swap python/3.7
python /work/padrich/pileup_recovery/pileup_recovery.py -shot 95173 -diag
KN3G -det 5 -pwidth 128 -TimeThr 55 -pattern
/work/padrich/shot_9999027_ch5_pattern.txt -output_dir /work/padrich
```

## 2.2 A utility to produce the reference signal

The reference signal (or “the pattern”) is constructed by averaging a large number of waveforms acquired with low activity calibration source(s). Under such conditions every event consists of a waveform of a single gamma-ray signal.

An **important assumption** is made here, that on average the shape of the signal is independent of the gamma-ray energy and only the maximum amplitude varies with energy.

The program to produce the reference signal is: **signal\_pattern\_factory.py**

The **mandatory arguments** are:

- input** full path to the input file (i.e., a binary, segmented mode calibration data file),
- output** full path to the output file (a text file to save the reference signal),
- pwidth** number of samples in the waveform (the “PWIDTH” parameter of the DAQ. E.g., 128),
- TimeThr** the time threshold in seconds. Events in the input file that have timestamp corresponding to time > TimeThr are not taken into account for production of the reference signal,
- minAmp1** minimum amplitude of a waveform to be considered useful for reference signal production,
- maxAmp1** maximum amplitude of a waveform to be considered useful for reference signal production.

Example of the command to produce the reference signal for the Gamma Camera detector 1, based on calibration data in /work/ncruz/CalibrationData/KN3G/9999027/shot\_9999027\_ch1.bin, is:

```
python /work/padrich/pileup-recovery/signal_pattern_factory.py -input
/work/ncruz/CalibrationData/KN3G/9999027/shot_9999027_ch1.bin -output
shot_9999027_ch1_pattern.txt -pwidth 128 -TimeThr 1200 -minAmp1 1000 -maxAmp1 6000
```

It should not take long time to produce a reference signal. The above command may be executed directly from the command line.

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	11 of 21

### 3. Validation of the pile-up recovery software

The algorithm and the software were validated using data measured in controlled conditions with high activity  $^{137}\text{Cs}$  source.

The measurements were done using 1'' by 1''  $\text{LaBr}_3(\text{Ce})$  scintillator coupled to a photomultiplier tube equipped with an active base. The signal from the PMT was digitized using CAEN DT5730 desktop digitizer.

A sample of data was first taken with low activity  $^{137}\text{Cs}$  source. This sample was free of pile-up events and served to produce the reference signal characteristic of the detector setup. Another set of data was taken with the same setup and with high activity  $^{137}\text{Cs}$  source (count rate of about 80k counts/s). This data served to validate the software and to assess the efficiency of the method.

Figure 5 shows energy spectrum (uncalibrated) obtained for the high activity  $^{137}\text{Cs}$  source without applying the pile-up recovery software (left side) and the energy spectrum resulting from application of the pile-up recovery software on the same data (right side). Here the energy was calculated as the area under the signal.

There were in total 4000000 events in the data set from which 3965379 were fully recovered and 4274 partly recovered. From those 4000000 events 180939 events were initially classified as multiplicity<sup>4</sup> 2 pile-ups and 6234 events were classified as multiplicity higher than two. From the events initially classified as multiplicity 2, 180348 events were fully recovered what corresponds to 99.7% success rate, and 24 events were rejected what corresponds to 0.01% rejection rate (the remaining events were partly recovered, i.e., only one gamma-ray was recovered).

From the events initially classified as multiplicity higher than two, 5751 events were fully recovered, what corresponds to 92.2% success rate and 317 events were rejected, what corresponds to 5.1% rejection rate.

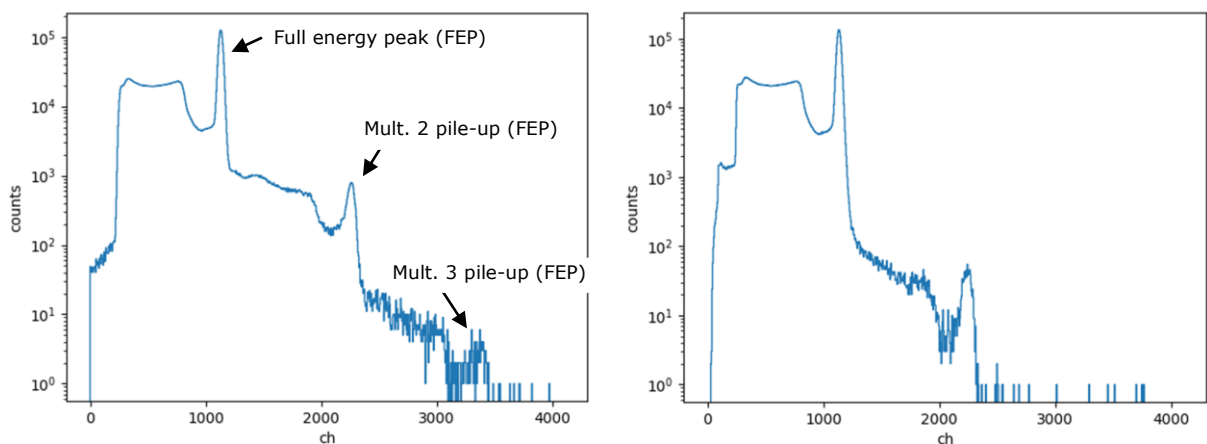


Figure 5. Uncalibrated energy spectrum measured with high activity  $^{137}\text{Cs}$  source. Left side: without applying pile-up recovery software; full energy peak as well as multiplicity 2 and 3 full energy pile-up peaks are indicated. Right side: spectrum resulting from the same data but first processed with the pile-up recovery software.

<sup>4</sup> See Appendix 1

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	12 of 21

Analysis of the spectra obtained without and with pile-up recovery reveals that virtually all events with multiplicity higher than 2 are correctly recognized as pile-ups and recovered or rejected. Furthermore, by integrating the counts in FEP peaks of double fold pile-up (around channel 2250 in Figure 5) it can be concluded that about 95% of multiplicity 2 pile-up events are correctly classified and treated while the remaining 5% is mistreated as multiplicity 1 events. Those are the events in which both gamma-rays are so close in time that the resulting signal is not distinguishable from the signal produced by a single gamma-ray and thus it is not possible to resolve the individual gamma-rays (see below).

#### 4. Evaluation of the algorithm's resolving power

The resolving power of the algorithm, i.e., its ability to correctly identify and recover two overlapping signals in function of their separation in arrival time, was studied using artificially synthesized pile-up events.

The synthetic data were produced by mixing two multiplicity 1 events measured with one of the detectors of the JET Gamma Camera using a calibration source. In this procedure the second waveform was shifted in time by a random offset and added to the first one. Such an event was then stored along with energies of both original events as well as with the time offset introduced between them.

The pile-up recovery software was applied on such artificially produced data. The output was analyzed in terms of recovery success rate as a function of time offset. Figure 6 shows the result of this analysis. On the X axis is the time offset between maximum amplitudes of both waveforms given in ns. The recovery success rate in percent is given on the Y axis.

When applied on this synthetic data the algorithm is 100% efficient as long as maxima of both waveforms are at least 12 ns away. This corresponds to the rise time of the signal from Gamma Camera detectors. On the other hand, at 200 MHz sampling frequency, 12 ns correspond to about 2.5 digitizer samples.

Based on the above, the pile-up recovery algorithm developed here is able to correctly classify and recover all multiplicity 2 events as pile-up as long as two gamma-rays arrive to the detector with time shift no less than the duration of the signal rise time.

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	13 of 21

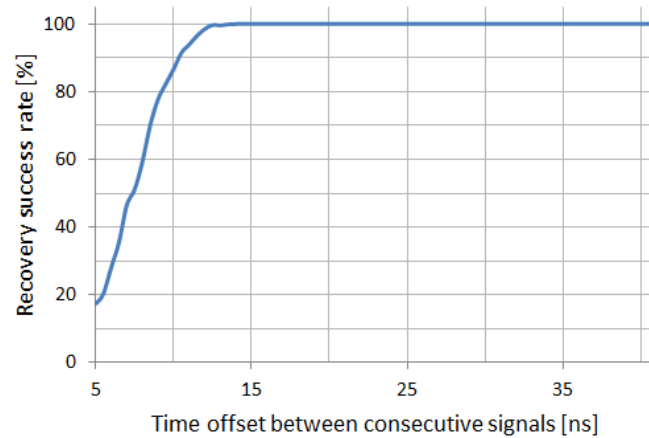


Figure 6. Resolving power of the pile-up recovery algorithm and software developed here.

## 5. Limitations

Efficiency of the pile-up recovery algorithm and software developed here depends on and is limited by:

- 1) the quality of the reference signal,
- 2) the degree to which a waveform collected under conditions of a real JET pulse could be described as linear combination of reference waveforms,
- 3) the fulfillment of the condition that the shape of the signal is independent of the gamma-ray energy.

Concerning the first point, the reference signal should be as free of random noise as possible. This can be practically achieved by averaging a large enough number of calibration waveforms (say 1000 or more).

Concerning the second point, the more noisy and distorted (e.g., unstable baseline) the experimental data the less efficient the pile-up recovery no matter how perfect the reference signal.

Concerning the last point, it is rather obvious that the fit of a reference signal to a measured one is going to be successful only if the shapes of both are the same (apart from the random noise in the data). This is true as long as the signal shape is independent of the gamma-ray energy.

Especially, for the case of Gamma Camera, i.e., small scintillators readout by MPPCs, it is at this point not clear to what extent this condition is fulfilled. This should be verified with more detailed studies motivated by a known nonlinearity of the MPPC response at higher energies.

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	14 of 21

## 6. Summary and conclusions

1. A pile-up recovery algorithm relying on fitting the measured data with expected characteristic signal pattern was developed. The new algorithm is more elaborate and advantageous to algorithms known in the literature. It is able to certain extend (see sec. 4) recover events in which two gamma-rays are so closely spaced in time that are not recognized as pile-up by the first-order derivative method.
2. The algorithm was implemented in Python 3. Software tools for processing segmented mode data from gamma-ray diagnostics were developed. It could be used to process data from Gamma Camera and Gamma Spectrometers.
3. The algorithm and the software were validated with data measured in laboratory conditions with a high activity  $^{137}\text{Cs}$  source.  
Based on analysis of reconstructed gamma-ray energy spectra it is deduced that about 95% of multiplicity 2 events are correctly identified as pile-up. The full recovery success rate of those events is 99.7% and the rejection rate is 0.01%.  
Furthermore, virtually all events of multiplicity higher than 2 are correctly classified as pile-up and treated with overall full recovery success rate of 92.2% and with the rejection rate of 5.1%.
4. Application of the algorithm on the artificially synthetized pile-up data revealed that the algorithm is able to correctly classify and recover with close to 100% success rate a multiplicity 2 event as pile-up as long as two signals are separated in time by no less than their characteristic rise time. In case of JET Gamma Camera this corresponds to about 12 ns or at 200 MHz sampling rate, to the duration of about 2.5 samples of the digitizer.
5. The real-life efficiency of the pile-up recovery is limited with fulfillment of conditions discussed in Sec. 0. Clearly, the more noisy and distorted the measured data the less efficient the pile-up recovery. The efficiency would also drop if for any reason the shape of the signal would be dependent on the gamma-ray energy. In the latter case, a modification of the algorithm to account for the energy dependence of the reference signal would possibly be viable.
6. The experience gained so far indicate that in case of run-away experiments, the single most important cause of event rejection during pile-up recovery is inability to reliably establish the baseline level. The number of evens rejected due to this can be substantial and by far exceeds combined number of events rejected due to any other reason.

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	15 of 21

## 7. References

1. I. I. Mahmoud, M. S. ElTokhy and H. A. Konber, *Pileup recovery algorithms for digital gamma ray spectroscopy*, [JINST 7 \(2012\) P09013](#)
2. X. L. Luo, V. Modamio, J. Nyberg, et al. (2018) *Pulse pile-up identification and reconstruction for liquid scintillator based neutron detectors*, [Nuclear Instruments and Methods in Physics Research, Section A 897 \(2018\) 59-65](#).

### **The report was prepared by**

Przemysław Adrich on behalf of all GCU and GSU project teams.

*The present document is of intellectual property of the JET Enhancements (JET4) Work Package (in particular NCBJ). This document cannot be neither copied nor distributed without right permission from EUROfusion Consortium. The MTCD2@NCBJ device was prepared by Eastern Wall Technologies Company.*

*This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 and 2019-2020 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.*

*This scientific work was partly supported by Polish Ministry of Science and Higher Education within the framework of the scientific financial resources in the years 2015-2019 allocated for the realization of the international co-financed project.*

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	16 of 21



## Appendix 1.

### Description of the implemented pile-up recovery algorithm

The pile-up identification and recovery relies on comparison of a measured signal with a known reference signal. The reference signal is an idealized, normalized waveform expected from a detector as a response to a single gamma-ray event. Generally, the exact analytical function describing the shape of the reference signal is not known. Fortunately it does not need to be known because the reference signal is easily produced as an average of large number of pile-up free waveforms obtained from measurements with low activity calibration sources.

In what follows we classify events according to their multiplicity, i.e., the number of individual gamma-ray signals present in the measured waveform. An event with a single gamma-ray has multiplicity = 1, a double fold pile-up, i.e., an event consisting of two gamma-rays observed in a single detector, has multiplicity = 2 and so on.

Event multiplicity is estimated by means of counting the number of times the waveform's first-order derivative changes sign from positive to negative. This is a quick way of obtaining a relatively good multiplicity estimate, however, it is not sensitive to very close pile-ups, i.e., events in which two or more signals are so shortly spaced in time that the first-order derivative is not changing the sign between the maxima of individual signals. Examples of different multiplicity waveforms are presented in Figure 7 along with their first order derivative and multiplicity estimated in a way described. For each example the number of recovered gamma-rays is also indicated as "N gammas recovered". Note that the initially estimated multiplicity can be lower than actual number of gamma-rays recovered from the data.

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	17 of 21

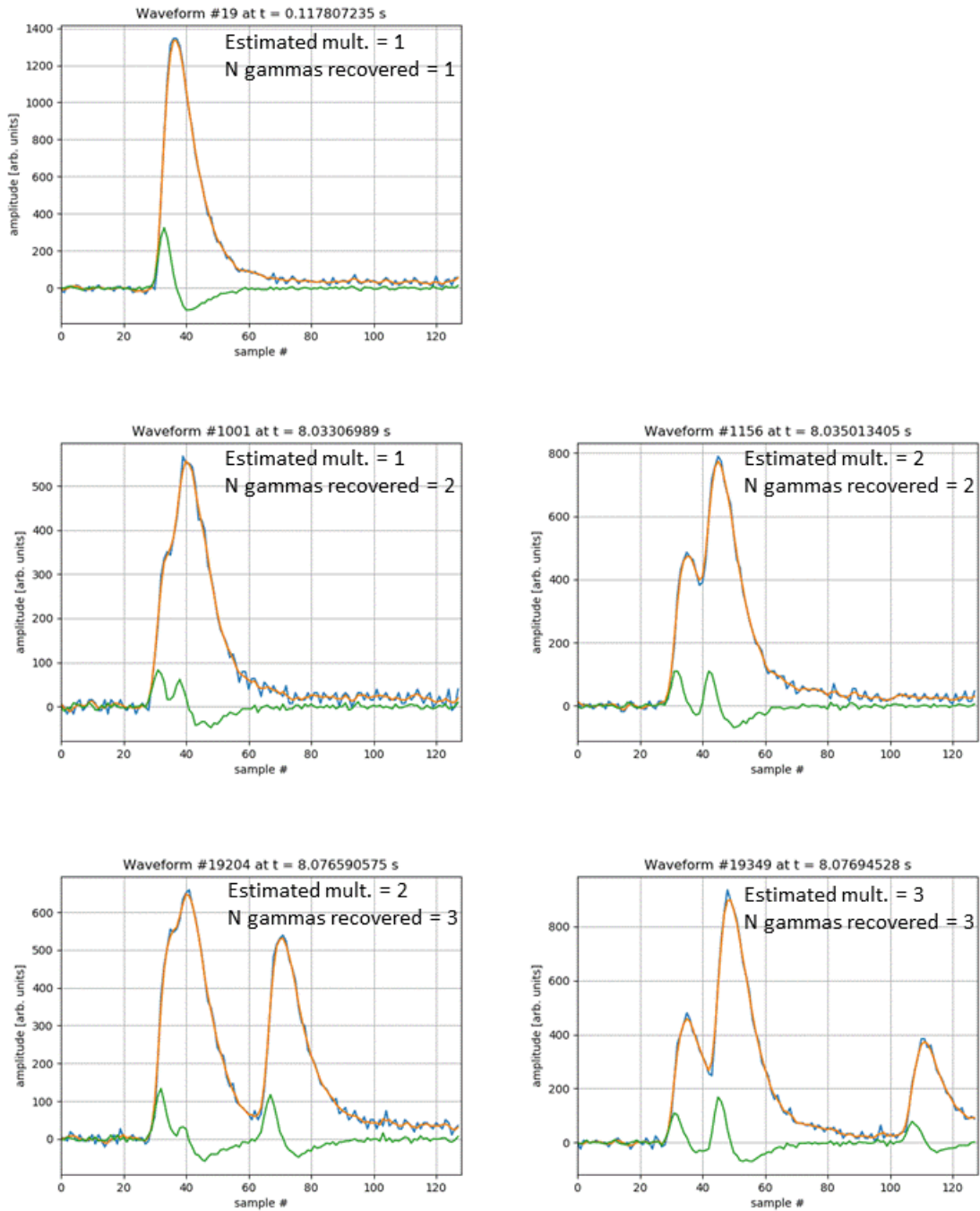


Figure 7 Examples of waveforms corresponding to pile-ups of different multiplicities. Blue line is the original waveform as digitized by the DAQ, orange is the waveform smoothed by means of three-point moving average and green is the first-order derivative of the smoothed waveform. The multiplicity is estimated by counting the number of times the first-order derivative changes its sign from positive to negative. Only the sign changes that occur while the waveform's amplitude is above the acquisition threshold are counted.

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	18 of 21

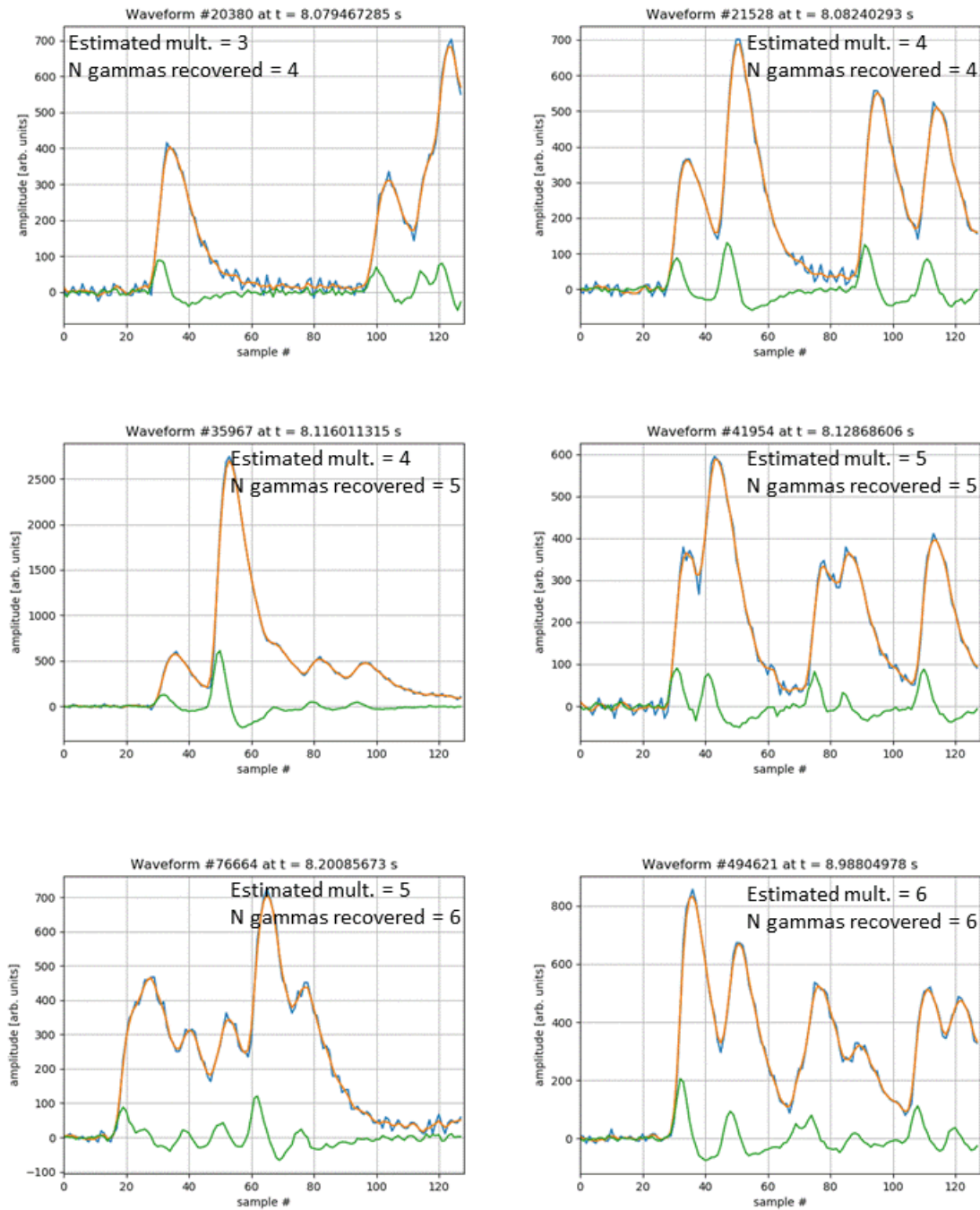


Figure 7 cont. Examples of waveforms corresponding to pile-ups of different multiplicities. Blue line is the original waveform as digitized by the DAQ, orange is the waveform smoothed by means of three-point moving average and green is the first-order derivative of the smoothed waveform. The multiplicity is estimated by counting the number of times the first-order derivative changes its sign from positive to negative. Only the sign changes that occur while the waveform's amplitude is above the acquisition threshold are counted.

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	19 of 21

The reference signal is constructed by averaging a large number of waveforms measured with low activity calibration source(s). Under such conditions every event consists of a single gamma-ray signal. An **important assumption** is made here, that the shape of the signal is independent of the gamma-ray energy and only the amplitude varies with energy. An example of a reference signal waveform constructed for one of the JET Gamma Camera detectors is shown in Figure 8.

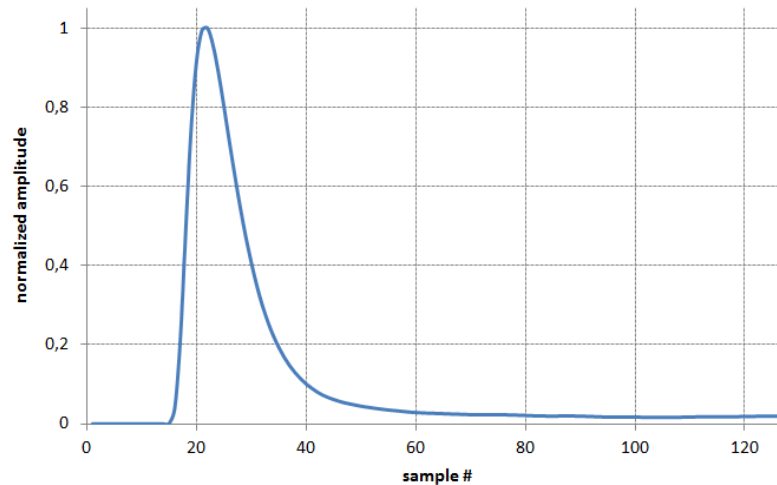


Figure 8. Example of a reference signal. An idealized response of the detector to a single gamma-ray.

The reference signal has a characteristic rise time, fall time and width. One would expect very similar characteristics for all multiplicity = 1 waveforms. Thus, a second step of the algorithm is the comparison of rise time, fall time and width of the first detected signal (by first-order derivative) with values expected for a “clean” single fold signal (i.e., the reference signal). If the differences are small, one can conclude the measured signal resembles the reference one. In order to extract the precise time and energy of the corresponding gamma-ray, the reference signal is fitted against the measured signal. More precisely, the fitted function has the form

$$amplitude(t) = A * reference\_waveform(t - offset),$$

where  $A$  and  $offset$  are the free parameters of the fit and  $reference\_waveform(t - offset)$  is the value of the reference signal waveform (as seen in an example in Figure 8) interpolated at  $(t - offset)$ .

The fit is validated by means of Pearson’s chi-squared goodness of fit test (often referred to as simply the chi-squared test). It is perhaps the most widely used goodness of fit test. The test was specifically designed for testing categorical data. The test does not require the probability distribution to be continuous (in opposite to other popular statistical tests, e.g., Kolmogorov–Smirnov test) and does not require the data to follow a Gaussian distribution (as opposite to normality tests, e.g., Shapiro–Wilk test).

If the Pearson’s p-value is above a given threshold the fit is accepted. The energy (given as the value of the fit parameter  $A$ ) as well as time (offset of maximum amplitude converted to absolute time) are stored. The waveform resulting from the fit (i.e., the waveform  $amplitude(t)$ ) is also stored as well as subtracted from the original waveform. The multiplicity

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	20 of 21

is then recalculated and the procedure is repeated as long as multiplicity > 0.

The algorithm described so far is known in the literature and was already applied in the field of gamma-ray spectroscopy. This algorithm, however, is not able to properly recover events in which two gamma-rays arrive to a detector in such a short time span that the analysis of the first-order derivative results in (wrong) attribution of multiplicity = 1. Such events are normally rejected since the quality of the fit is poor. As long as the count rate is relatively low one would not suffer significant data loss due to this limitation. However, with growing count rates the algorithm would throw away a growing fraction of data. Since very high count rates are expected during DT campaign at JET, the algorithm developed here goes beyond the first-order derivative method. Namely, if the first detected signal is distinctly different from the reference signal (i.e., has significantly longer rise time, fall time or width) or a fold-1 fit as described above fails, then it is assumed that the signal is in fact composed of two gamma-rays closely spaced in time and not resolved by the first-order derivative method. In such a case, a four parameter fit is attempted in which a function of the form

$$amplitude(t) = A_1 * reference\_waveform(t - offset_1) + A_2 * reference\_waveform(t - offset_2)$$

is fitted against the measured waveform.

The fit is validated in the same way as in the case of a single fold fit (Pearson's chi-squared test) and if accepted the results are stored and the waveform  $amplitude(t)$  is subtracted from the original waveform. The multiplicity is then estimated anew and the procedure is repeated as long as multiplicity > 0. Only if this double fold fit fails the event is ultimately rejected.

Subsequent plots in Figure 9 illustrate steps of the pile-up recovery.

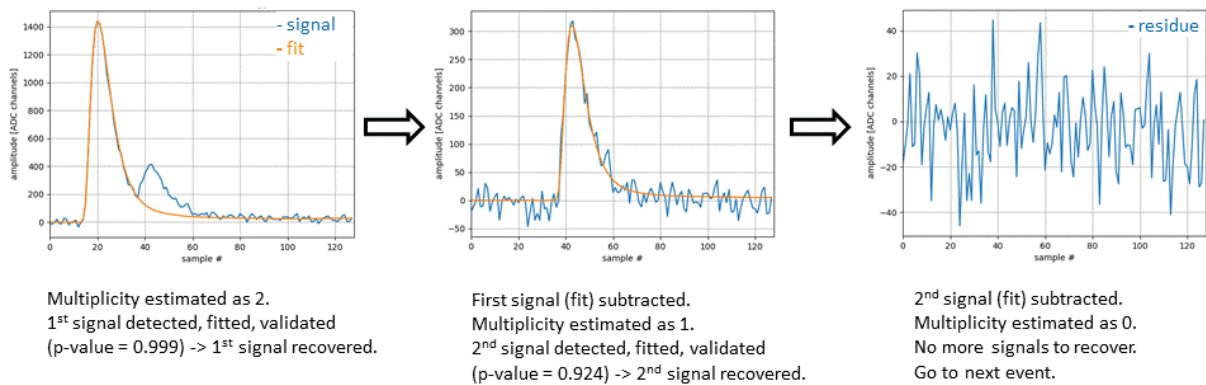


Figure 9. Illustration of the major steps of the pile-up recovery algorithm.

WPJET4	GCU&GSU @ NCBJ	Date	Page
GCU&GSU	PILE-UP RECOVERY	December 2019	21 of 21