JMonitor: A collection of applications for detector monitoring

B. Strandberg

November 23, 2026

Abstract

The JMonitor software is a collection of applications for monitoring the performance of the detector. A brief overview of the applications and their cross-use ise provided.

Scope

This document is meant to be used in conjunction with the Doxygen documentation of Jpp [1]. The parameter interfaces to the applications are (will be) documented in the source code, see e.g. JMonitorK40 application in Doxygen. Here the basic principles and some examples are provided. This document is work in progress, contributions are welcome.

1 JMonitorK40

JMonitorK40 is an application that is used to determine the parameters of PMTs in DOMs in-situ. It takes a raw sea data file as input and outputs a .root file with histograms. JMonitorK40 requires s detector file (.detx) associated with the raw sea data file to work properly. JMonitorK40 output is an input to JFitK40 - the two applications are intertwined and both are required to extract the parameters of the PMTs.

1.1 Getting the detector file

There are several ways to acquire the detector file. One option is to use the example application $JPP_DIR/examples/JDB to fetch it from the database. Jpp examples are not compiled by default, so one needs to do

```
cd $JPP_DIR/examples/JDB
make
```

To acquire the detector file, one can then use

```
$JPP_DIR/examples/JDB/JDB -u your_db_username -!  your_db_password -D 29 2800 >
my_detfile.detx
```

to fetch the detector file of detector ID 29 (ORCA 1 line), run number 2800.

1.2 Output histograms

The application will create three histograms in the `JMonitorK40` output file per each DOM defined in the detector file provided to `JMonitorK40`. If no data is found for the DOM in the raw sea data file, the histograms will be empty. The histograms are described below and are filled in the application by looping over the timeslice data.

Histogram DOM-ID.2S

This is a 2D histogram that contains the PMT hit coincidence data. In each KM3NeT DOM there are 31 PMTs, thus there are $31 \times 30/2 = 465$ PMT pairs. Per each PMT pair the quantity $t_{\mathrm{m}} - t_{\mathrm{n}}$ is recorded, where $t_{\mathrm{m(n)}}$ stands for the hit time of $\mathrm{PMT}_{\mathrm{m(n)}}$.
The x-axis of the 2D histogram enumerates the PMT pairs, the y-axis holds the coincidence data. The PMT pairs are sorted by the opening angle. The higher the index, the smaller the opening angle between the PMTs. The data along the y-axis is the raw coincidence histogram of each PMT pair, accumulated by looping over the timeslices.

Histogram DOM-ID.1B

This is a 1D histogram that counts the coincidences from background events per each PMT pair. The background events can be calculated from 'random' data, 'rates' data or 'tails' data, see Sect. 1.3. This histogram provides the background count per one bin along the x-axis of a coincidence histogram of one PMT pair. For example: take the pair 465 of the output histogram DOM-ID.2S. In the backround subtraction (performed in the `JFitK40` routine) a loop over the bins of the spectrum of the pair 465 is performed and the value stored in the DOM-ID.1B at x = 465 is subtracted for each bin.

Histogram DOM-ID.1L

This is a 1D histogram that counts the livetime of each PMT pair, in seconds. The livetime of each PMT pair is counted by accumulating the durations of the timeslices during which both PMTs of the pair were recording data (were not in HRV, were within rate cuts etc). The livetimes are necessary to convert the raw coincidences in histograms DOM-ID.2S and DOM-ID.1B to coincidence rates. This is basically the simple procedure of dividing the number of counts with the time duration during which the counts were accumulated. The conversion is performed in `JFitK40`.

Other histograms

There are three additional histograms. The `weights_hist` stores the width of the coincidence window used in the analysis (parameter `-T`) and the bin width of the coincidence histograms (typically 1 ns). Additionally, there are histograms PMT_first and PMT_second. These histograms store the PMT ID's that correspond to the 465 pairs. This information is helpful for post-analysis of the JMonitor/JFit results.

1.3 Different background estimation methods

The `JMonitorK40` application can estimate the background coincidences in several ways. Background estimation method is selected through the parameter `-b`, which can have values

- `-b randoms`,
- `-b rates`,
- `-b tails`.

'Randoms' refers to a background estimation that uses hits well outside the defined coincidence window. Such hits are only recorded in L0 timeslices, as in other timeslices some coincidence window (typically 10–25 ns) is implemented at trigger level. Hence the 'randoms' background estimation should be used only with L0 data.
'Rates' refers to a background estimation that uses the individual hit rates of the PMTs that make the pair. This estimation is based on Poission statistics and is the most universal of the three.
'Tails' refers to a background estimation that uses the tail regions of the coincidence spectrum of the PMT pair. The 'tails' estimation works best for sufficiently large coincidence window ($\sim$25 ns).

1.4 Examples

Example command 1:

```
JMonitorK40 -T 24 -M 2+8 -V 100+9900 -a my_detfile.detx -f
KM3NeT_00000029_00003186.root -o JMonK40_out.root -C JDAQTimesliceL1 -b tails -B 19+24
```

This command specifies a window of 24 ns (`-T 24`) for the coincidence of two PMTs. For example if $\mathrm{PMT_m}$ has a hit at t_m and $\mathrm{PMT_n}$ has a hit at t_n, the histogram corresponding to the PMT pair m, n will only be filled with $t_\mathrm{m} - t_\mathrm{n}$ if $|t_\mathrm{m} - t_\mathrm{n}| \leq 25$ ns. It is the responsibility of the user to check that the coincidence window is not larger than permitted by the trigger settings (may be improved in the future).
The multiplicity range (`-M 2+8`) is set to 2–8. This means that the coincidence histogram of a PMT pair m, n is filled if the DOM has ≥ 2 and ≤ 8 hits in the time window of 24 ns (given through parameter `-T`). Note that for this and other parameters the range is provided through the '+' sign.
The hit rate veto (`-V 100+9900`) is set to 100–9900 Hz. The hits of PMTs that have a hit rate outside the provided band are ignored.
The application takes as input the detector file `my_detfile.detx` and the raw sea data file `KM3NeT_00000029_00003186.root` and outputs a file `JMonK40_out.root`.
The application is set to look for L1 timeslices (`-C JDAQTimesliceL1`).
The application is set to estimate the background from the tail regions of the coincidence spectrum of each PMT pair (`-b tails`). The tail region is specified to 19–24 ns (`-B 19+24`). In the code the estimation uses the tails at both ends, i.e. both the range [19, 24] ns and the range [-19, -24] ns.

Example command 2:

```
JMonitorK40 -T 24 -M 2+8 -V 100+9900 -a my_detfile.detx -f
KM3NeT_00000029_00003186.root -o JMonK40_out.root -C JDAQTimesliceL1 -b rates
```

Exactly as example 1, but in this case the background coincidences of each PMT pair are estimated from the rate data of the PMTs.

2 JFitK40

`JFitK40` application fits the 2D histograms in the output of `JMonitorK40`. The 1D background histogram is used to subtract the background from the coincidence spectrum of each PMT pair, the 1D livetime histogram is used for normalisation. Both the background subtraction and the normalisation are performed in `JFitK40`, `JMonitorK40` merely prepares the data for this application. `JFitK40` requires the same detector file as input that was provided to `JMonitorK40`.

2.1 Output histograms

Per each DOM defined in the detector file, the JFitK40 application will look for a set of histograms in the JMonitorK40 output. After background subtraction and normalisation the 2D histogram that holds the coincidence data is fitted. Several output histograms are created per each DOM. The creation of some of the listed histograms is controlled through the parameter interface.

Histogram DOM-ID.1B

This is the DOM-ID.1B histogram from the JMonitorK40 output after livetime normalisation, which converts from background counts to background rates.

Histogram DOM-ID.2S

This is the DOM-ID.2S histogram from the JMonitorK40 output after livetime normalisation and background subtraction. The livetime normalisation converts from coincidence counts to coincidence rates. This histogram is fitted in the JFitK40 application.

Histogram DOM-ID.2F

This is a 2D histogram that results from subtracting the fit result from the fitted DOM-ID.2S histogram. For example if the fit function is a perfect representation of the data, this histogram is filled with 0s.

Histogram DOM-ID.1t0

This is a 1D histogram that contains the time offsets of the PMTs in the DOM from the fit result.

Histogram DOM-ID.1sigma

This is a 1D histogram that contains the transit time spreads of the PMTs in the DOM from the fit result.

Histogram DOM-ID.1QE

This is a 1D histogram that contains the relative PMT efficiencies of the PMTs in the DOM from the fit result.

2.2 Examples

Example command 1:

```
JFitK40 -A -N -P pmt_file.txt -a my_detfile.detx -f JMonK40_out.root -o
JFitK40_out.root -w -!  "-1 0" -c
```

The application takes the JMonitorK40 output and a detector file (same as provided to JMonitorK40) as inputs and outputs the file JFitK40_out.root.
The detector file provided as input will be overwritten (the flag -A) by using the time offsets from the fit result.
Saving of the histogram DOM-ID.2F is optional and set through the flag -N.
Optionally one can provide a PMT file to the application (-P pmt_file.txt), the PMT efficiencies from the fit result will be written to the PMT file. The PMT file can then be used with JTriggerEfficiency.
Writing of the histograms DOM-ID.t0, DOM-ID.1sigma and DOM-ID.1QE is optional and set through the flag -w.

The fit is constrained to PMT pairs that have an opening angle smaller than 90° between them through the flag `-c`. This is helpful when e.g. analysing L2 data.
Optionally the time offset of PMT 0 in all DOMs is fixed to 0 through the option `-! "-1 0"`. Offsets of several PMTs can be fixed, e.g. `-! "-1 0 -1 20"` fixes the time offsets of PMTs 0 and 20 in all DOMs. Time offsets of PMTs of specific DOMs can be fixed, e.g. `-! "-1 0 -1 20 808432835 15"` fixes the time offsets of PMTs 0 and 20 for all DOMs and the time offset of PMT 15 in DOM 808432835.

References

[1] `http://pi1241.physik.uni-erlangen.de:8080/view/Jpp/job/Jpp_trunk/doxygen/#`