

EVENT INDEX BASED CORRELATION ANALYSIS FOR THE JUNO EXPERIMENT

Tao Lin^a (on behalf of the JUNO collaboration)

Institute of High Energy Physics, Chinese Academy of Sciences, Beijing, 100049, China

E-mail: ^alintao@ihep.ac.cn

The Jiangmen Underground Neutrino Observatory (JUNO) experiment is mainly designed to determine the neutrino mass hierarchy and precisely measure oscillation parameters by detecting reactor anti-neutrinos. The total event rate from DAQ is about 1 kHz and the estimated volume of raw data is about 2 PB/year. But the event rate of reactor anti-neutrino is only about 60/day. So one of the challenges for data analysis is to select sparse physics signal events in a very large amount of data, whose volume can not be reduced by using the traditional data streaming method. In order to improve the speed of data analysis, a new correlated data analysis method has been implemented based on event's index data. The index data contain the address of events in the original data files as well as all the information needed by event selection, which are produced in event pre-processing using the JUNO's Sniper-based offline software. The index data are subsequently selected by using refined selection criteria with Spark so that the volume of index data is further reduced. At the final stage of data analysis, only the events within the time window are loaded according to the event address in the index data. A performance study shows that this method achieves a 14-fold speedup compared to correlation analysis by reading all the events. This contribution will introduce detailed software design for event index-based correlation analysis and present performance measured with a prototype system.

Keywords: JUNO, time correlation, analysis tool

Tao Lin

Copyright © 2021 for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

1. Introduction

The Jiangmen Underground Neutrino Observatory (JUNO) experiment under construction in southern China, will have a rich physics program, besides neutrino mass ordering and precise measurement of oscillation parameters [1,2]. The JUNO detector is located 700 m deep underground. As shown in Figure 1, it contains a central detector, water Cherenkov detector and top tracker. The innermost of the central detector contains 20 kton of liquid scintillator, surrounded by 17,612 20inch PMTs and 25,600 3inch PMTs.

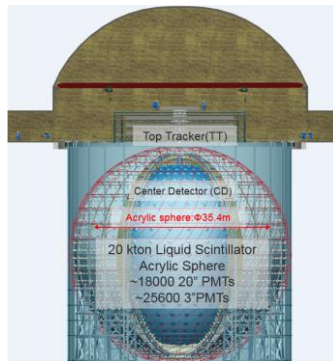


Figure 1. Schematic view of the JUNO detector

As one of the important systems, the JUNO offline software [3] is used to process the 2PB/year data coming from the detector. As shown in Figure 2, the offline software includes an underlying framework, external libraries and several applications. The applications consist of event generators, simulation, calibration, reconstruction and analysis tools. In order to support all these applications, SNiPER (Software for Non-collider Physics ExpeRiment) [4] is adopted as the underlying data processing framework.

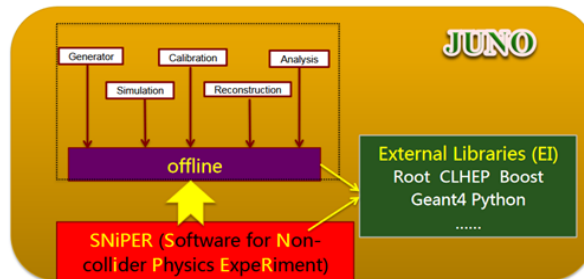


Figure 2. The architecture of the JUNO offline software

The challenges in the analysis are the rare signals and the time correlation, which are quite different from collider experiments. The total event rate is about 1 kHz, while the event rate of reactor antineutrinos is about 60 per day. Therefore, most of the events are backgrounds for the analysis. If there is no time correlation between the events, then all the background could be discarded. However, a neutrino is detected via the inverse beta decay (IBD) process, producing a prompt signal positron and a delayed signal neutron with an average neutron capture time of 200 μs . Hence all the events in the same time window are needed. Due to the time correlation, it is difficult to use the big data technologies. In order to improve the speed of data analysis and the ability to analyze data interactively [5], an event index-based method has been proposed. The key idea of this method is to reduce the I/O by loading the events within the time window on demand according to the event address in the selected index. In this paper, the design and implementation of this method will be shown.

2. Design and Implementation

2.1 The event index-based method and the analysis event index

An event index contains an address of the event in the original data as well as the necessary information needed by the event selection. There are three stages in the event index-based method, as shown in Figure 3. The first stage is the generation of event index. The event index data are produced in event pre-processing by the SNIpER framework. The second stage is the reduction of the event index using big data technologies, such as Spark. The event index is selected by using refined criteria and the volume of the event index is reduced. The third stage is the time correlation analysis in the correlation analysis framework. The event addresses are loaded from the event index and the events are loaded automatically according to these addresses.

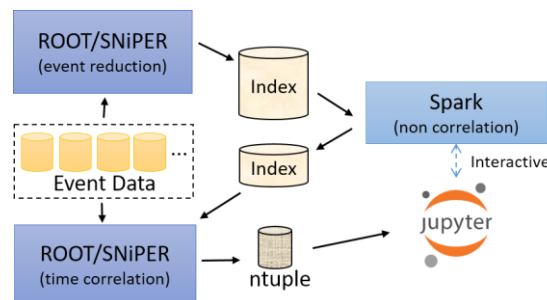


Figure 3. The schematic view of the event index-based method

As already mentioned, the event index contains two parts. The first part is the address of the event in the data. The address contains a reference to the event data file and a reference to the entry in the event data file. This part will be used by the analysis framework internally. The second part is the user defined event level information. This part will be used for the single event selection. In this study, the reconstructed energy, the reconstructed vertex and the event time are stored in the second part.

The file formats of the event index could be in the plain text format, the ROOT [5] format and the HDF5 [6] format, which are supported by both ROOT and the big data technologies. A data frame-like structure of the event index can be easily analyzed and processed by these technologies. In the current implementation, the event index in plain text format are written by ROOT and then processed by Spark and ROOT.

2.2 The event index-based correlation analysis framework

The event index-based correlation analysis framework has been developed based on SNIpER. One of the essential features of SNIpER is the ability to manage multiple events in the same time window using an event buffer service. With the modular design of the framework, there is no impact on the user developed event selection algorithms whether using event index or not. The only changes are the event loop and the event buffer service with event index support, as shown in Figure 4.

For the analysis without event index, the event loop is driven by the ROOT-based event data. The events will be read by the ROOT I/O and put into the event buffer at the beginning of each event. The event selection algorithms will access the event data from the event buffer. When the processing of the current event is done, the framework will read and analyze the next events until all the events are processed.

For the analysis with event index, the event loop is driven by the event index, instead of the event data. At the beginning of each event processing, the index-based event buffer service will load an index via the index I/O. Then according to the reference to the file, the service will check whether the file is loaded or not. If the file is not loaded, then the ROOT I/O will be used to open the file. When the file is ready, the event will be then loaded according to the entry number from the event index. The next step is loading the other events in the same time window via the ROOT I/O. When all the events in the time window is ready, the event selection algorithms can access them. In the next

event processing, the framework will load the next index, instead of the next event. By using this method, a fraction of background events can be skipped.

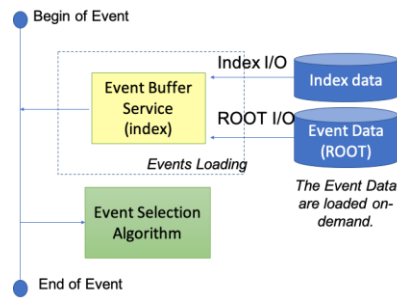


Figure 4. The data flow of the correlation analysis framework

3. Performances

In order to evaluate the performances of the correlation analysis framework, two cases are studied: one case is only considering the I/O without time correlation analysis and another case is considering the time correlation analysis. The performance of event loading with different ratios is shown in Figure 5. The complete event index will be read by the index I/O and then the event selection will be randomized according to the ratios. If an event index is selected, then the corresponding event data is loaded. In this case, the other events in the same time window are not loaded. In order to reduce the uncertainty, all the measurements are repeated 30 times. As shown in the figure, even though there are overheads, the event index can speed up the event loading by reducing the ROOT I/O.

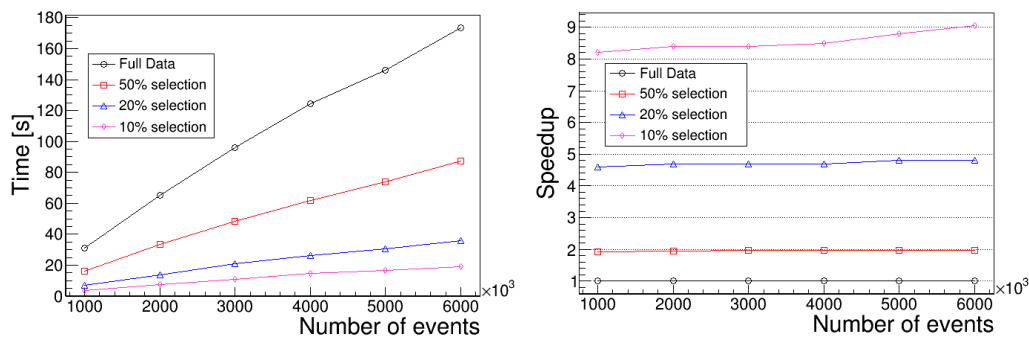


Figure 5. The performance of the event loading with different ratios

As the neutrino events are rare, the radioactivity background samples in liquid scintillator are generated to mimic the IBDs. All the isotopes in the decay chains are considered. The intervals between two events are sampled according to the event rates. The fiducial volume cut and energy cut are applied in the selection of single events. Then the energy cut, time interval cut and distance cut are applied in the selection of correlated events. In the test, there are about 2.5% of events selected in the event index according to the selection criterion and about 5% of events loaded from the event data. The performance of the time correlation analysis is shown in Figure 6. Compared to the analysis without index data, there is about 14-fold speedup. The method can provide further speedup if less events are selected in the future.

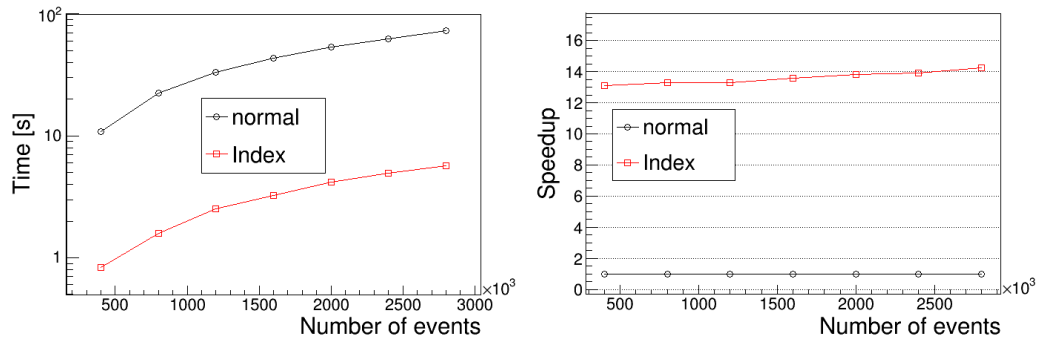


Figure 6. The performance of the correlation analysis compared to the normal analysis

4. Conclusion

In this study an event index-based correlation analysis method has been developed and applied to the JUNO analysis. By reducing the I/O of event data, this method could improve the speed of the data analysis. The speedup is about 14 when 5% of events are really loaded. In order to further speed up the analysis, the parallelized version is still under development.

5. Acknowledgement

This work is supported by National Natural Science Foundation of China (NSFC 11805223) and Xie Jialin Fund.

References

- [1] An F.P. et al. [JUNO Collaboration]. Neutrino Physics with JUNO // J.Phys.G 43 (2016) 3, 030401
- [2] Abusleme A. et al. [JUNO Collaboration]. JUNO Physics and Detector // accepted by Progr. Part. Nucl. Phys. arXiv 2104.02565
- [3] Huang X.T., Li T., Zou J.H., Lin T., Li W.D., Deng Z.Y., Cao G.F. Offline Data Processing Software for the JUNO Experiment // PoS ICHEP2016 (2017), 1051
- [4] Zou J.H., Huang X.T., Li W.D., Lin T., Li T., Zhang K., Deng Z.Y., Cao G.F. SNIPEr: an offline software framework for non-collider physics experiments // J.Phys.Conf.Ser. 664 (2015) 7, 072053
- [5] Lin T. [JUNO Collaboration] Jupyter-based service for JUNO analysis // EPJ Web Conf. 245 (2020) 07011
- [6] Brun R. and Rademakers F. ROOT - An Object Oriented Data Analysis Framework // Nucl. Inst. & Meth. in Phys. Res. A 389 (1997) 81-86
- [7] The HDF Group, Hierarchical data format version 5 // <http://www.hdfgroup.org/HDF5>