PROCEEDINGS OF SCIENCE

Evaluation of an FPGA-based fast machine-learning trigger for neutrino telescopes

Francesca Capel,[a,*] **Christian Spannfellner,**[b,c,*] **Christian Haack**[d] **and Janik Prottung**[b]

[a] *Max Planck Institute for Physics,*
Föhringer Ring 6, 80805 Munich, Germany
[b] *Technical University Munich, School of Natural Sciences, Physics Department ECP/E49,*
James-Franck-Strasse 1, 85748 Garching, Germany
[c] *ORIGINS Excellence Cluster,*
Boltzmannstraße 2, 85748 Garching, Germany
[d] *Erlangen Centre for Astroparticle Physics*
Nikolaus-Fiebiger-Str. 2, 91058 Erlangen, Germany

E-mail: capel@mpp.mpg.de, christian.spannfellner@tum.de

Energetic neutrinos provide a view into the underlying processes of astrophysical particle accelerators, but their weakly interacting nature makes them challenging to detect. Current experiments instrument large volumes of ice or water with 3D grids of photomultiplier tubes (PMTs) to capture the Cherenkov light produced by interactions of high-energy neutrinos. Such detectors must be located in remote locations deep underwater or in ice to reduce atmospheric background signals. These challenging conditions impose strict limits on the power and bandwidth available for data transfer to the surface, and triggers are used to maintain manageable rates. We evaluate the potential of fast, intelligent machine-learning triggers that can be implemented on low-power field-programmable gate arrays (FPGAs). We aim to make the most of the given hardware with improved discrimination of signal and background and therefore improved sensitivity to low-energy events. In particular, we focus on the case of underwater neutrino detectors and the efficient discrimination of track-like signals from the bioluminescence background. We develop a machine-learning trigger by using the planned P-ONE experiment as a case study and implement a software testbench to compare its performance to a less complex trigger design based on coincident detections.

38th International Cosmic Ray Conference (ICRC2023)
26 July - 3 August, 2023
Nagoya, Japan

*Speaker

PoS(ICRC2023)1104

1. On-FPGA trigger systems

FPGAs (field-programmable gate arrays) are integrated circuits consisting of programmable logic blocks and interconnects. These building blocks can be configured to perform various functions tailored to hardware designs or needs. Compared to other front-end processing units, the advantage of FPGAs lies in their re-programmability, allowing to update the firmware as a design evolves - ultimately avoiding the need to replace hardware-specific chips or circuit boards. While attractive, the widespread use of FPGAs in scientific experiments has been prohibited by the required knowledge of specialized hardware description languages, like VHDL and Verilog. Recent progress in high-level synthesis (HLS) approaches helped to overcome this barrier. These HLS approaches translate between more commonly used programming languages, such as C++ and Python, and hardware description languages. Specifically, the open-source hls4ml[1] software package allows for the implementation of machine learning algorithms, e.g. classifiers based on neural networks. Particle physics experiments at the LHC [1] and the BELLE II [2] experiment have implemented successfully such on-FPGA machine learning algorithms as trigger systems. More advanced trigger algorithms open up the possibility of increasing the detector sensitivity and allow physics analyses at lower energy thresholds. Specifically, astrophysical neutrino detectors, constructed at remote locations in the deep-sea or antarctic, often face power and bandwidth constraints, where novel trigger algorithms could allow more effective data acquisition.

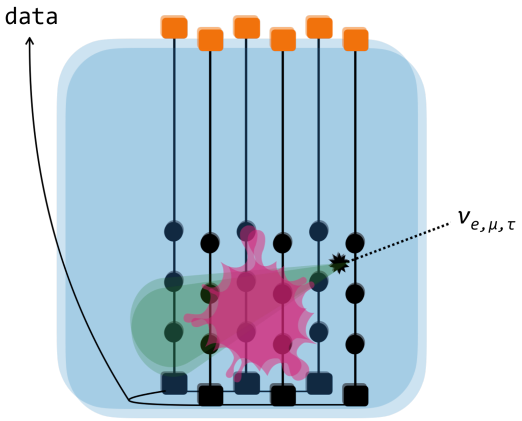

Figure 1: Artistic illustration of a neutrino detector. Neutrino telescopes instrument large volumes with photosensitive instruments, commonly referred to as optical modules. Maritime detectors are exposed to higher background rates induced by bioluminescence and ^{40}K decay.

2. Neutrino detector trigger schemes

Astrophysical neutrino detectors use Earth itself as a detector by instrumenting vast volumes of a transparent medium, like ice or water, with photomultiplier tubes (PMTs). The PMTs are used to detect the Cherenkov light, which emerges from particle interactions within or in the vicinity of the detector volume. Depending on the initial energy of the neutrino and its flavour at the time of interaction, a cascade, track-like, or double-bang event can be observed in the instrumented detector volume. The spatial, time, and intensity information of the PMTs is used in turn to reconstruct the initial energy and direction of the astrophysical neutrino. However, neutrino telescopes are exposed to background events induced by atmospheric muons and neutrinos, as well as dark counts from the PMTs, which overlay the signal. Underwater telescopes often have to face even more complex background patterns due to bioluminescence and more continuous background originating from radioac-

[1] https://github.com/fastmachinelearning/hls4ml

tive decay of potassium (see fig. 2). Current-generation neutrino detectors typically use time-over-threshold (ToT) multiplicity triggers to separate background events from the regular data stream. The trigger algorithms themselves are based on the varying spatial and temporal profile of the background compared to an astroparticle interaction. Low-energy bioluminescence and potassium events are expected to be observed only locally around one or a few optical modules. After applying the trigger algorithm, the selected data is broadcast to a central station for offline analysis. New advancements in graph neural networks (GNNs) allow their effective use for offline classification and reconstruction of neutrino events [4]. Expanding on these concepts, similar approaches can be investigated for online trigger systems, promising enhancements in the signal to background discrimination and increased detector sensitivity. In particular, low-energy neutrino events could be more effectively detected without the need for a more densely-instrumented detector volume. The networks can also be trained to differentiate the dynamic bioluminescence signals and the track or cascade morphology of astrophysical signals by combining the data from several optical modules. The challenge is to accelerate event classification from $\sim$ ms timescales, typical of current offline methods, to $\sim$ μs scales for real-time implementation (see also [5, 6]).

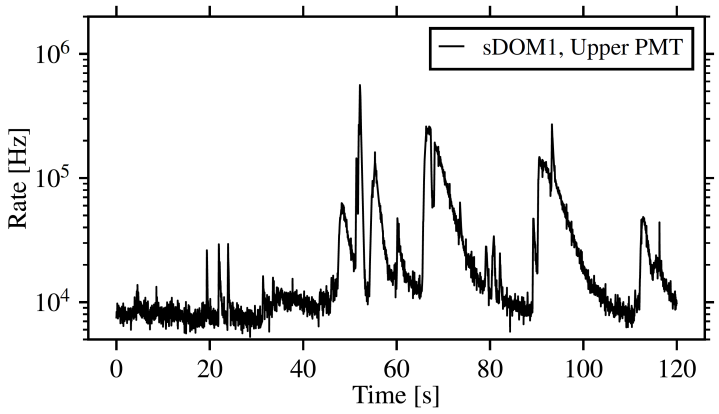

Figure 2: PMT rate measured at the Cascadia Basin, offshore of Vancouver Island, Canada. The rate has been measured in 30 ms intervals. Continuous background of the order of 10^4 Hz, originating from diffuse bioluminescence and ^{40}K decay. Higher rates are induced by bioluminescence bursts, lasting a few seconds [3].

3. Software framework

To evaluate potential trigger designs, we develop a software framework that enables us to explore different detector geometries, simulate realistic event signatures and generate suitable training datasets. Our framework is based on the Ananke[2] and Olympus[3] Python packages that form part of the P-ONE software development efforts. Ananke provides a fast, uniform and flexible data format that we can use to specify the details of a detector layout, response and necessary simulation inputs. This package can also be used to store simulated datasets and facilitate the merging or redistribution of simulated events to build large datasets appropriate for the training and assessment of trigger algorithms. The Olympus package simulates signal and background events and their detection for a given detector configuration. Neutrino events are specified by an initial time, position and energy, and events are propagated through the detector medium depending on their energy and

[2] https://github.com/pone-software/ananke
[3] https://github.com/pone-software/olympus

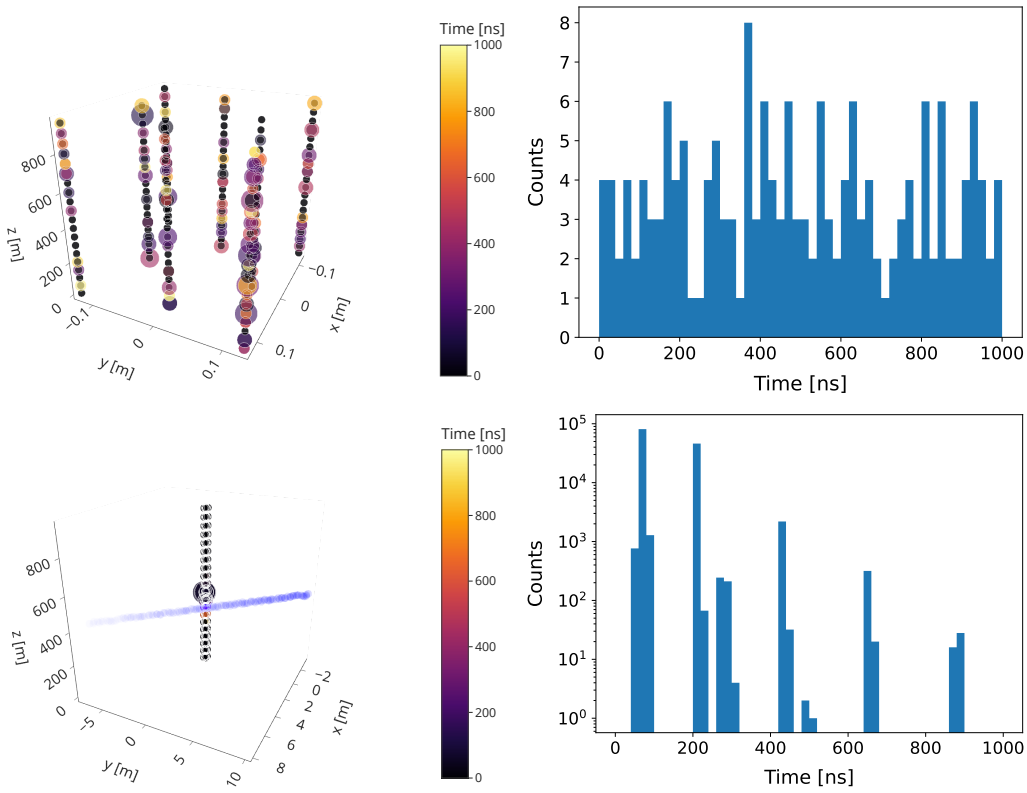

Figure 3: Example detected events generated with Ananke and Olympus. The upper panels show noise from a merged dataset of single-containing electrical and bioluminescence components. PMTs from a single-line detector are spaced out for visibility. The lower panels show a 2.2 TeV cascade event in a single-line detector in addition to the noise signal. The left panels show the event view with optical modules shown as coloured circles. The colour of these circles represents the arrival time, and the size gives the normalised number of hits. Blue points show the light sources, with lighter shading for earlier times and darker shading for later times. The right panels show the number of hits in all PMTs as a function of time.

interaction type. A Cherenkov light source is defined at each point where secondary particles are created, and the subsequent photons are propagated towards the detector. This detailed full Monte Carlo approach is implemented using the Hyperion[4] package. However, it is too computationally expensive to permit the simulation of many events as needed to develop effective training datasets. To reduce the computational burden while maintaining per-PMT detector information, we developed an alternative approximate approach that assumes that Cherenkov light travels directly from the interaction source to the PMT without interacting [7]. We validated this approach against the more detailed method and confirmed that the results were satisfactory for our simulation framework. We also implemented two different types of background events: electrical noise and bioluminescence. Electrical noise is stochastic, uncorrelated and based on the properties of the PMTs. We simulate the number of PMT hits from electrical noise using a Poisson distribution with a given rate, and the events are distributed evenly in time. The bioluminescence contribution is more complex to model. We use observations from the STRAW-a and STRAW-b experiments [3, 8], as well as the

[4] https://github.com/pone-software/hyperion

experience from the development of the KM3Net observatory [9] to motivate realistic simulations using the Fourth Day open-source software package [10]. This software enables the simulation of bioluminescence signals for 2D geometries, so we further assume rotational invariance of the signal around the detector modules and precompute a large database of expected signals for a given detector configuration. These events can then be randomly selected and merged with other events as required. In addition to the two types of background that we consider here, we also expect a further contribution from the radioactive decay of ^{40}K that will essentially form a noise floor. As we expect the bioluminescence signal to dominate the background events, we omit this contribution for now and plan to explore its impact in future work.

We implement a simple single-string detector within our framework and apply the abovementioned methods to generate a large training dataset. Our detector is motivated by the planned P-ONE prototype design and has 20 modules, each containing 16 PMTs, and separated by a 50 m vertical distance [11]. PMTs are assumed to have a dark noise rate of 1.6×10^{-4} ns^{-1}, an efficiency of 42%, a radius of 0.21 m, and an opening area radius of 37.5 mm. Fig. 3 shows examples of the different types of events generated. We combine generated events to create a training dataset containing 30 000 cascade events in the energy range from 100 GeV to $10^{5.5}$ GeV and 100 000 intervals of electrical and bioluminescence noise. A further cut is made to exclude the cascade events that result in a large number of hits in the detector, to focus on the more challenging detection cases. This results in a final number of 2 290 cascade events. One third of the total dataset is set aside for validation purposes.

To set up a GNN model, we use the publicly available GraphNet framework to be trained with our prepared dataset [4]. In particular, we implement the ConvNet network architecture with three convolutional layers, a dropout ratio of 0.3 and 8 nodes per intermediate layer. For the training, we use 16 epochs with a batch size of 8. Our preliminary results show excellent discrimination of cascade events from noise. We evaluate the receiver operating characteristic (ROC) curve for our validation dataset and find an area under the curve, AUC = 0.9995. We interpret this early result as an encouraging proof of principle but also an indication that our simulated datasets likely underestimate the background contributions. In future, we plan to build on the developed simulation capabilities to generate more realistic training datasets with a focus on neutrino events at the threshold of detection and improved noise modelling. We also intend to use this testbench to compare the performance of the implemented GNN with more standard PMT-level and module-level coincidence triggers.

4. Hardware framework

The implementation of on-FPGA neural networks will be done by FPGA evaluation boards, i.e. Zynq UltraScale+ evaluation kits. For the translation of the developed trigger algorithms, the existing hls4ml[5] synthesis tool for graph and convolutional neural networks can be explored. The on-FPGA trigger systems, however, will require optimization efforts in order to run efficiently and with low latency, as online buffering capacities are limited. Neural networks can be optimized by compression, quantization, and parallelization [12]. Compression is an effort to reduce the redundant number of synapses and neurons of a given neural net without impacting its overall

[5] https://github.com/fastmachinelearning/hls4ml

performance. Quantization aims to optimize the precision of the calculations, for e.g. weighting factors, by effective use of data types with reduced sizes. Parallelization summarises the effort to find the balance between resource usage and algorithm throughput by varying the amount of parallel and sequential multiplications for a given layer [12]. These optimization principles will be applied to the developed and translated trigger algorithms, where their performance can be tested with the purchased evaluation kits. We plan to perform the first hardware implementations of trigger algorithms in late summer this year.

5. Conclusion and outlook

The hardware and software testbenches currently in preparation will allow the development of trigger systems tailored to the needs of underwater neutrino detectors. We will start with simple trigger designs, gradually increasing their complexity and optimization to harness the available FPGA resources. With larger complexity, i.e. the implementation of larger arrays of optical modules, the neural network trigger performance might exceed that of regular multiplicity trigger schemes. After the development of optimized trigger algorithms, performance tests with data from maritime neutrino detectors, such as KM3NeT [13] or P-ONE [14], can be anticipated. Finally, boundary constraints induced by the available power, bandwidth, and latency budgets need to be considered to optimize for realistic implementations of on-FPGA trigger systems in the scope of an astrophysical neutrino observatory. Ultimately, changes in the detector efficiency and geometry induced by ocean dynamics, i.e. sedimentation, marine growth, and deep-sea currents, could be included in the data stream and captured by the neural network. Such aspects could so far not be included with time-over-threshold multiplicity trigger systems and promise improved online classification and discrimination. This approach, however, requires an online data acquisition structure that allows the combination of environmental data and physics data, i.e. Cherenkov hits through particle interactions, in real-time.

Acknowledgements

We acknowledge financial support from the Excellence Cluster ORIGINS, which is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC-2094-390783311.

References

[1] J. Duarte et al., *Journal of Instrumentation* **13** (2018) P07027.

[2] S. Neuhaus et al., *Journal of Physics: Conference Series* **608** (2015) 012052.

[3] N. Bailly et al., *The European Physical Journal C* **81** (2021) 1071 [2108.04961].

[4] R. Abbasi et al., *Journal of Instrumentation* **17** (2022) P11003.

[5] N. Choma et al., *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)* (2018) 386.

[6] F.J. Yu et al., *Trigger-level event reconstruction for neutrino telescopes using sparse submanifold convolutional neural networks*, 2023.

[7] J. Prottung, *Graph neural network based trigger algorithm for the Pacific Ocean Neutrino Experiment (P-ONE)*, June, 2023.

[8] K. Holzapfel et al., *PoS* **ICRC2023** 1166.

[9] H.v. Haren et al., *Astroparticle Physics* **67** (2015) 1.

[10] S. Meighen-Berger et al., *Bioluminescence modeling for deep sea experiments*, 2021.

[11] C. Spannfellner et al., *PoS* **ICRC2023** 1219.

[12] Y. Iiyama et al., *Frontiers in Big Data* **3** (2021) 55.

[13] A. Margiotta, *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **766** (2014) 83.

[14] M. Agostini et al., *Nature Astronomy* **4** (2020) 913.

PoS(ICRC2023)1104