UC Irvine
UC Irvine Previously Published Works

Title

A novel trigger based on neural networks for radio neutrino detectors

Permalink

https://escholarship.org/uc/item/1s83g2zx

Authors

Anker, A
Paul, MP
Baldi, P
et al.

Publication Date

2022-03-18

Copyright Information

Peer reviewed

A novel trigger based on neural networks for radio neutrino detectors

Astrid Anker[a,*] **and Manuel P. Paul**[a] **on behalf of the ARIANNA Collaboration**
(a complete list of authors can be found at the end of the proceedings)

[a]*Department of Physics and Astronomy, University of California, Irvine, CA 92697, USA*

E-mail: aanker@uci.edu

The ARIANNA experiment is a proposed Askaryan detector designed to record radio signals induced by neutrino interactions in the Antarctic ice. Because of the low neutrino flux at high energies, the physics output is limited by statistics. Hence, an increase in sensitivity significantly improves the interpretation of data and offers the ability to probe new parameter spaces. The trigger thresholds are limited by the rate of triggering on unavoidable thermal noise fluctuations. The real-time thermal noise rejection algorithm enables the thresholds to be lowered substantially and increases the sensitivity by up to a factor of two compared to the current ARIANNA capabilities. A deep learning discriminator, based on a Convolutional Neural Network (CNN), is implemented to identify and remove a high percentage of thermal events in real time while retaining most of the neutrino signals. We describe a CNN that runs on the current ARIANNA microcomputer and retains 95% of the neutrino signals at a thermal rejection factor of 10^5. Finally, the experimental verification from lab measurements are conducted.

*37*th *International Cosmic Ray Conference (ICRC 2021)*
July 12th – 23rd, 2021
Online – Berlin, Germany

---

*Presenter

1.   Introduction

Extreme-high-energy (EHE) neutrino astronomy expands the opportunity to learn more about the fierce processes of astronomical objects [1]. Radio-based detectors such as ARIANNA leverage the $O(1\,\mathrm{km})$ attenuation lengths of radio signals in ice to search for the distinct Askaryan pulses produced by neutrino induced particle showers. The radio technique enables cost-efficient instrumentation for monitoring large detection volumes. However, because of the low flux of EHE neutrinos, event rates are still small for a large array of hundreds of radio detector stations. Thus, improving the sensitivity of the detectors is one of the primary objectives. One way to increase the sensitivity is to lower the trigger threshold so that smaller neutrino signals are recorded by the detector. The problem with this is that the trigger rate is already dominated by unavoidable thermal noise fluctuations, and the detector has a limited data transmission rate. However, if thermal noise is identified and rejected in real time, the trigger thresholds can be lowered while maintaining the same data rate, thus increasing the sensitivity of the detector. This proceeding demonstrates that deep learning techniques can be implemented in the current ARIANNA data acquisition system to reject thermal noise in real time. An extended version of this work is to be published [2].

2.   The ARIANNA experiment

The ARIANNA detector is an array of autonomous radio detector stations located in Antarctica at Moore's Bay and the South Pole. In general, the stations are composed of log periodic dipole antennas (LPDA's) and dipoles in groups of 4 or 8 antennas. The radio signals are digitized and captured using a custom-made chip design known as the SST [3]. The analog trigger system imposes requirements on individual waveforms to maintain the sensitivity to Askaryan pulses; a high and low threshold must occur within 5 ns, and multiple antennas channels (at least 2 of 4) must meet the high-low threshold within 30 ns of one another. Once a station has triggered, the digitized waveforms (containing 256 samples for each channel) are piped into an Xilinx Spartan 4 FPGA, and then further processed and stored to an internal 32 GB memory card by an MBED LPC 1768 microcontroller before being transmitted out of Antarctica.

For this study, the data transmission rate for Iridium communications is the target trigger rate because for most remote Antarctic locations, satellite is more reliable. Stations that rely solely on Iridium are expected to operate at trigger rates from ~ 0.3 mHz to keep losses due to data transfer, f_{trans}, below 3%. The trigger thresholds of ARIANNA are adjusted to a certain multiple of the Signal to Noise Ratio (SNR), defined here as the ratio of the maximum absolute value of the amplitude of the waveform to the V_{RMS}^{noise}. Currently, the stations are set to trigger above 4.4 SNR to reach the constrained trigger rate of order 1 mHz.

3.   Expected gain in sensitivity

NuRadioMC [4] is used to simulate the increase in sensitivity if 99.99% of all thermal noise fluctuations are rejected in real time, which allows to increase the low-level trigger rate by four orders of magnitude while maintaining the same data rate. The sensitivity of the ARIANNA detector at Moore's Bay is simulated using the relation between trigger threshold and trigger rate from [5]. The
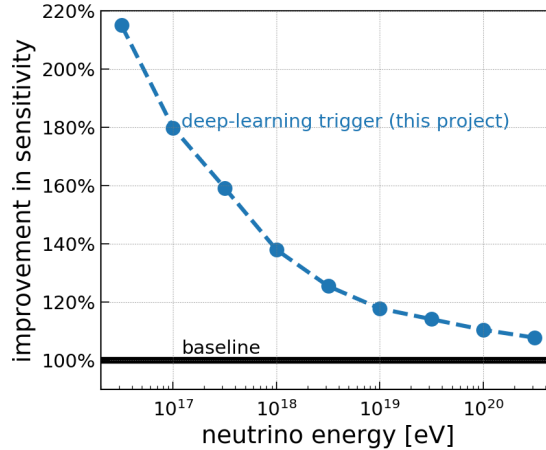
Figure 1: Expected improvement in sensitivity to high-energy neutrinos with the deep-learning trigger developed in this work. The baseline is the standard ARIANNA high/low trigger with a 2 out of 4 antennas coincidence requirement for the nominal bandwidth of 80 MHz to 800 MHz at a thermal noise trigger rate of 10 mHz. The blue curve shows the sensitivity for a trigger threshold corresponding to a trigger rate of 100 Hz and otherwise the same simulation settings.

resulting gain in sensitivity is shown in Fig. 1 and increases by almost a factor of two at energies of 10^{17} eV. The improvement decreases towards higher energies because fewer of the recorded events are close to the trigger threshold but at 10^{18} eV there is still an increase in sensitivity of 40%.

4. Thermal noise rejection using deep neural networks

To implement a deep learning filter, the network structure needs to be optimized for fast and accurate classification. The two metrics are signal efficiency (defined here as the ratio of correctly identified signal events to the total number of signal events) and noise rejection factor (defined here as $\frac{1}{(1-N_{ratio})}$, where N_{ratio} is the ratio of correctly identified background events to the total number of background events). For the trained network, ideally it would be able to reach 5 orders of magnitude thermal noise rejection while providing a high signal efficiency (at or above 95%). This would enable the trigger threshold to be lowered significantly – thus increasing the sensitivity to EHE neutrinos – while keeping a low event rate of 0.3 mHz. Typically using a more complex network structure would yield higher efficiencies, but this would also create a slower network. These two constraints need to be optimized as the deep learning architecture is developed.

4.1 Generation of training data sets

NuRadioMC [4] is used to simulate a representative set of the expected simulated neutrino events for the ARIANNA detector on the Ross ice shelf. The generated neutrino interactions are distributed uniformly in the ice around the detector with random incoming direction and with an energy spectrum following an astrophysical and cosmogenic neutrino flux expectation; the astrophysical flux measurement by IceCube with a spectral index of $\gamma = 2.19$ [6] is combined with a GZK neutrino flux model based on Auger data for a 10% proton fraction [7].

The resulting radio signals are simulated in the four LPDA antennas of the ARIANNA station by convolving the electric-field pulses with the antenna response, and the rest of the signal chain is approximated with an 80 MHz to 800 MHz band-pass filter. An event is recorded if the signal pulse crosses a high and low threshold of 3.6 times the V_{RMS}^{noise} within 5 ns in at least two LPDAs within 30 ns. At such a low trigger threshold, noise fluctuations can fulfil the trigger condition at a non-negligible rate. Therefore, the signal amplitude is required to be at least 2.8 times the V_{RMS}^{noise} before adding noise. In total 200,000 events are generated and this is called the *signal data set* in the following. The training data set for thermal noise fluctuations is obtained by simulating thermal noise in the four LPDA antennas and saving only those events where a thermal noise fluctuation fulfills the trigger condition described above. In total one million events are generated and this is called the *noise data set* in the following.

4.2 Network structures and training

All of the networks are created with Keras [8]. To find the best network, the network size is optimized for the number of Floating Point Operations (FLOP). The number of FLOPS can be approximated by counting the amount of nested loop iterations required to classify incoming data. The efficiencies of two different input data sizes are studied: the two channels with the maximum amplitude with 2 x 256 samples, and only 100 samples around the position of the maximum signal of the channel with the highest amplitude (typically the dominant neutrino signal spans over less than 50 samples).

The Fully Connected Neural Network (FCNN) used in this baseline test is a fully connected single hidden layer with a node size of 64 for the 100 samples and 128 for the 512 samples, into a sigmoid activation. The Convolutional Neural Network (CNN) structure consists of 5 filters with 10x1 kernels each and a ReLU activation, into a dropout of 0.5, max pooling with size 10x1, and then a flattening step to reshape the data and a sigmoid activation. The signal and noise event classification distributions fall between 0 and 1, where close to 0 is noise data and close to 1 is signal data. Once trained, with the 100 input sample CNN mentioned above, the sigmoid probability/threshold cut distribution on the left side of Fig. 2 is obtained. From this distribution, the amount of signal efficiency vs. noise rejection can be varied by choosing different threshold cut values. Training and testing these networks with each input data size yields the signal efficiency vs. noise rejection plot on the right side of Fig. 2.

Since all of the networks have efficiencies above the required 95% for signal at 10^5 noise rejection, the next consideration is the amount of FLOPS required for each network because this directly impacts the processing time. Typically, CNN's have less parameters overall due to their convolutional nature, which focuses on smaller features within a waveform; comparatively, the FCNN considers the whole waveform to make its prediction, so it requires more node connections. Next, the FLOPS and processing times are investigated for each network on two different devices.

5. Processing time on devices

There are several time scales that impact the physics capabilities of the ARIANNA detector: the time to readout and evaluate an event using a deep learning filter, T_{min}, the fraction of operational time lost (deadtime), f_{sys}, by the detector due to the readout and processing of triggered events
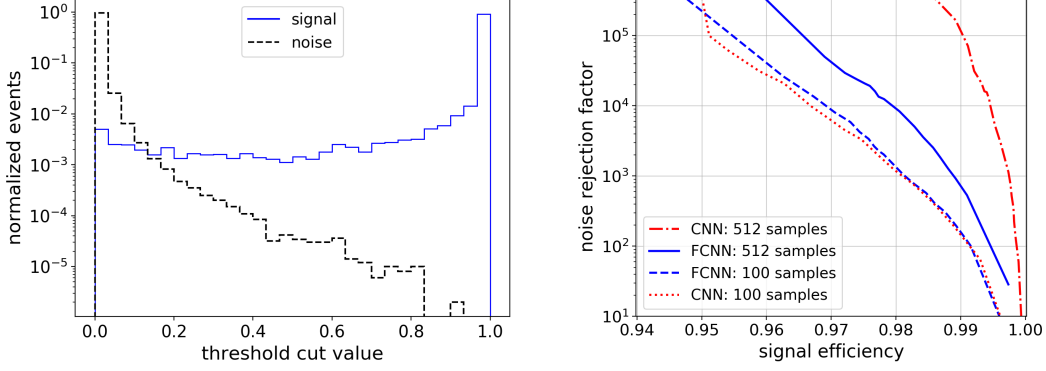
Figure 2: Histogram of the threshold cut value for signal and noise classification of the 100 sample CNN (left). Signal efficiency vs. noise rejection factor for networks with two different input data sizes (right). The CNN's have the structure of one convolutional layer containing 5 10x1 kernels. The FCNN's have one fully connect layer with node size 64 for 100 samples input data and node size 128 for 512 samples input data.

which occur randomly in time at a mean rate, R_{trig}, and finally the fractional loss in operational time due to the transmission of randomly occurring data over the Iridium satellite network, f_{trans}. The ARIANNA pilot station cannot acquire neutrino events while the data acquisition system is processing or transmitting events. Operational livetime, L, is the calendar time of nominal operation, T, corrected for the processing and transmission time loss, or L=T(1-f_{sys}-f_{trans}).

The time to readout and evaluate a triggered event, T_{min}, is modeled to contain three terms: (1) the time to transfer the data from the waveform digitizers to the microcomputer, T_{read}, (2) the time to reformat and calibrate the raw data for the deep learning evaluation, T_{ref}, and (3) the time to evaluate the event with deep learning, T_{dl}. Assuming the event storage is sufficiently rare and fast, it does not appreciably impact T_{min}. The readout time is given by T_{min}=T_{read}+T_{ref}+T_{dl}. T_{ref} and T_{dl} depend on the device whereas T_{read} depends on the details of the data acquisition system. T_{read} is assumed to be negligible here when evaluating new device platforms. In section 6.1, it will not be considered negligible when studying the readout speed.

5.1 Processing time

Two devices are explored for their processing time: a Raspberry Pi compute module 3+ microcomputer and an MBED LPC1768 ARM microcontroller. The MBED is the current device installed in ARIANNA, and it is implemented through custom C code. The Raspberry Pi is a microcomputer with a Raspbian operating system, which is based on Debian. Since the optimal networks found in the previous section are small and shallow, a custom C code is written that implements the trained neural networks on the MBED and Raspberry Pi. To test the prediction capabilities and the classification time in both devices, a simulated event is read in and either matrix multiplied by the array of weights and biases in the FCNN case or convolved with the weights and bias filters in the CNN case. Two different methods for measuring the processing time are used because the MBED is the only device integrated into the ARIANNA hardware. The processing time of the Raspberry Pi is measured by looping over the classification code 100 times with the clock function in C, and this interval divided by 100 is the average processing time per event. The

processing time of the MBED is measured by probing the ARIANNA digitizer which receives a reset pulse from the MBED when it is ready for a new event. Therefore by measuring the time between pulses, T_{min} and $T_{read} + (T_{ref}$ or $T_{dl})$ can be found when enabling or disabling T_{dl} or T_{ref}. The processing time T_{dl} and reformatting time T_{ref} for both devices are shown in Table 1 for a given model along with the FLOPS of each network. While Table 1 shows that the relationship is not completely linear, FLOPS provide a reasonable proxy to estimate the relative speeds of specific deep learning models. The two best performing networks are the CNN and FCNN with 100 input data samples, though their differences are minor. Thus, the 100 sample CNN is chosen for the experimental verification processes and is implemented into the current MBED software in the next section.

| Variable	model	FLOPS	MBED	Raspberry Pi
	FCNN 512 samples	61,568	45 ms	2.5 ms
	FCNN 256 samples	16,448	13 ms	1.0 ms
T_{dl}	CNN 256 samples	12,815	9.4 ms	0.95 ms
	FCNN 100 samples	6,464	4.7 ms	0.46 ms
	CNN 100 samples	5,045	3.7 ms	0.39 ms
T_{ref}	all networks		1.3 ms	0.095 ms

Table 1: Processing times per event, T_{dl} and the number of Floating Point Operations (FLOPS) of various models, and the reformatting time per event, T_{ref}, for an MBED and a Raspberry Pi.

6. Laboratory verification of deep-learning trigger

The performance of the deep learning algorithm is verified next with real data taken by the ARIANNA hardware. This section discusses the methods to test the algorithm on the ARIANNA DAQ board and the limitations of the hardware. Real-time data taking with LPDA antennas in the lab is challenging due to the lingering radio frequency noise present in the environment. Thus, for in-lab tests of the deep learning implementation, an analog "post LPDA antenna" radio neutrino pulse is created and injected into the ARIANNA hardware, bypassing the antenna, to verify the simulated results.

To accomplish this, neutrino signals are simulated in all detector antennas by the standard simulation tool, NuRadioMC. The neutrino template is then installed into an Agilent Tech. 81160A arbitrary pulse generator to produce an analog version of a typical neutrino waveform. This waveform is then input into a standard amplifier from the ARIANNA pilot station, which adds realistic amplifier noise. The noisy signal is then routed to the input of the ARIANNA DAQ board for data taking. A similar process is done to the simulated template. The injected pulse is set to trigger the ARIANNA digital acquisition board. Once digitized by the SST chip, the FPGA passes the digitized data to the MBED where the channel with the largest signal fluctuation is run through the deep learning algorithm and assigned a prediction probability. The deep learning performance on the ARIANNA hardware is also verified on a local computer, and both methods show similar results.

To test the CNN and verify that the simulated hardware components are comparable between measured and simulated data, a histogram of the probability distribution is plotted. This distribution
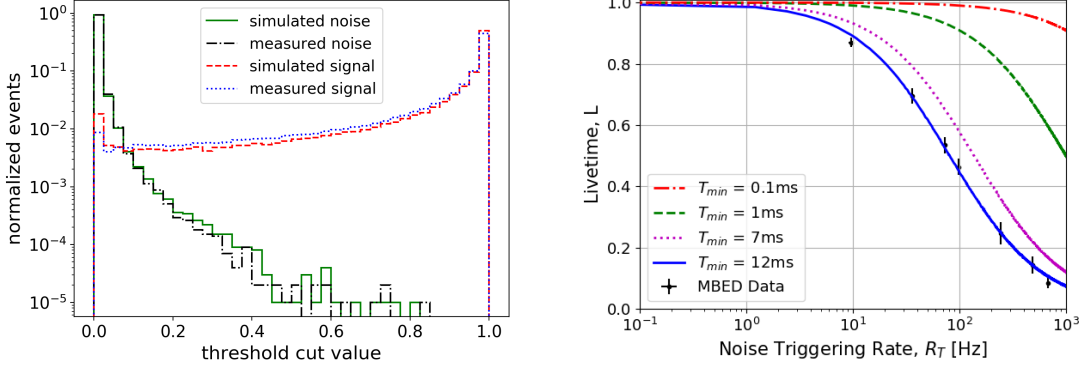
Figure 3: (left) Histogram of the threshold cut values of simulated and measured signal template and noise. (right) Livetime, L, as a function of Noise Trigger Rate, R_T, for three assumptions for the instrumental deadtime. Also plotted are black data points from the experimental verification study.

is obtained by using the previously trained full neutrino spectrum CNN to predict signal and noise from simulated and measured events. Running the network on a local computer allows for direct comparison between the expected outcome from simulated data to the experimental outcome of the ARIANNA hardware. As seen on the left plot of Fig. 3, the simulated data are in good agreement with the measured data, and errors associated are likely due to differences in SNR distributions and environmental effects such as strong radio pulses leaking into the cables.

This significant change in triggering rate opens the question regarding the current hardware and if it can handle triggering effectively at rates five orders of magnitude higher than before. The main limitation is the MBED, which contains most of the data packaging code, the communication, and oversees the operation of the entire station.

6.1 ARIANNA hardware performance for the CNN filter

This section explores the hardware performance of the ARIANNA pilot station. The readout rate is limited by the MBED because of the transferring of data, the packaging of events, and the running of on board analysis programs, including the deep learning algorithm. Taking the values from Table 1, and now including $T_{read} = 7.3$ ms (which was experimentally obtained with the same procedure as T_{dl}), the MBED has the event readout time $T_{min} = 12.3$ ms. To study the effect of T_{min} on operational livetime, L, the ARIANNA triggering system is simulated. T_{min} is directly related to L. First the effect of the deadtime on the triggering system is simulated by writing code that takes an exponential distribution of times, dt, representing the time between incoming triggers of events with a mean of R_{trig}^{-1}, where R_{trig} is the noise trigger rate. The simulation uses the distribution of times to calculate a fractional livetime, $1 - f_{sys}$, as a function of R_{trig}. To calculate f_{sys}, the program calculates over a time, T, the average portion of time spent in deadtime due to R_T.

To confirm the simulation results for the fractional livetime as a function of noise trigger rate, the ARIANNA DAQ board is injected with signal-like pulses periodically at a rate of 100 mHz while varying the noise triggering rate by changing the trigger threshold. In the right plot of Fig. 3, the results of the experiment are seen as the threshold is lowered, which in turn increases the noise rate and decreases the live time. This experiment has some potential error since neutrinos are

measured in the detector at random instead of periodic. This result motivates the consideration for improvements in hardware, such as replacing the MBED with a faster processing device.

7. Summary and Future plans

Due to the low neutrino flux at extreme high energies, the physics output is limited by statistics. Probing new parameter spaces is made possible by implementing deep learning techniques to increase the ARIANNA detector's sensitivity. A CNN was utilized on an MBED microcontroller to discriminate between thermal noise fluctuations and neutrino signal; this allowed the trigger rate to be increased by five orders of magnitude while still only saving neutrino candidates at the constrained 0.3 mHz rate. This CNN retained 95% neutrino signal at a thermal rejection factor of 10^5. The simulations study was verified in lab measurements that found an excellent agreement between the measured and simulated distributions. A more complete paper, currently in preparation, will provide additional details [2]. In addition, it will show that the DL procedure increases the operational livetime due to background events generated by "high wind" periods.

We consider several improvements to the deep learning application described in this paper as the processing speed of the current hardware is limited to 10 Hz. First, the ARIANNA hardware that sends the digitized data from the FPGA to the microprocessor can be parallelized to increase readout speeds. Second, upgrading the computing capabilities (in particular the FPGA and microprocessor) is expected to increase processing speeds significantly, leading to an increase of trigger rates to 100 Hz.

Acknowledgement

References

[1] M. Ackermann *et al. Bull. Am. Astron. Soc.* **51** (2019) 185.

[2] A. Anker *et al.*, "Improving sensitivity of the arianna detector by rejecting thermal noise with deep learning." In preparation, 2021.

[3] S. A. Kleinfelder, E. Chiem, and T. Prakash *2014 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)* (Nov, 2014) .

[4] C. Glaser, D. García-Fernández, A. Nelles, J. Alvarez-Muñiz, S. W. Barwick, D. Z. Besson, B. A. Clark, A. Connolly, C. Deaconu, K. D. de Vries, and et al. *The European Physical Journal C* **80** no. 2, (Jan, 2020) .

[5] C. Glaser and S. W. Barwick *JINST* **16** no. 05, (May, 2021) T05001.

[6] C. Haack for the IceCube collaboration *PoS(ICRC2017)1005* .

[7] A. van Vliet, R. A. Batista, and J. R. Hörandel *Physical Review D* **100** no. 2, (Jul, 2019) .

[8] F. Chollet *et al.*, "Keras." https://keras.io/, 2015.

Full Authors List: ARIANNA Collaboration

Astrid Anker[1], Pierre Baldi[2], Steven W. Barwick[1], Jakob Beise[3], Hans Bernhoff[4], Dave Z. Besson[5,6], Nils Bingefors[3], Maddalena Cataldo[7], Pisin Chen[8], Daniel García Fernández[9,7], Geoffrey Gaswint[1], Christian Glaser[3], Allan Hallgren[3], Steffen Hallmann[9], Jordan C. Hanson[10], Spencer R. Klein[11], Stuart A. Kleinfelder[12], Robert Lahmann[7,1], Jiayi Liu[1], Mitchell Magnuson[5], Stephen McAleer[2], Zach Meyers[9], Jiwoo Nam[8], Anna Nelles[9,7], Alexander Novikov[5,6], Manuel P. Paul[1], Christopher Persichilli[1], Ilse Plaisier[9,7], Lilly Pyras[9], Ryan Rice-Smith[1], Joulien Tatar[13], Shih-Hao Wang[8], Christoph Welling[9,7], Leshan Zhao[1]

[1]Department of Physics and Astronomy, University of California, Irvine, CA 92697, USA.
[2]Department of Information and Computer Science, University of California, Irvine, CA 92697, USA.
[3]Uppsala University Department of Physics and Astronomy, Uppsala SE-752 37, Sweden.
[4]Uppsala University Department of Engineering Sciences, Division of Electricity, Uppsala SE-752 37, Sweden.
[5]Department of Physics and Astronomy, University of Kansas, Lawrence, KS 66045, USA.
[6]National Research Nuclear University MEPhI (Moscow Engineering Physics Institute), Moscow 115409, Russia.
[7]ECAP, Friedrich-Alexander Universität Erlangen-Nürnberg, 91058 Erlangen, Germany.
[8]Department of Physics and Leung Center for Cosmology and Particle Astrophysics, National Taiwan University, Taipei 10617, Taiwan.
[9]DESY, 15738 Zeuthen, Germany.
[10]Whittier College Department of Physics, Whittier, CA 90602, USA.
[11]Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA.
[12]Department of Electrical Engineering and Computer Science, University of California, Irvine, CA 92697, USA.
[13]Research Cyberinfrastructure Center, University of California, Irvine, CA 92697, USA.