



Development of a Deep Neural Network for the data analysis of the NeuLAND neutron detector

C.A. Douma^{a,*}, E. Hoemann^b, N. Kalantar-Nayestanaki^{a,c,*}, J. Mayer^b, for the R³B Collaboration

^a KVI-CART, University of Groningen, Groningen, The Netherlands

^b Institut für Kernphysik, Universität zu Köln, Köln, Germany

^c Present address: Nuclear Energy Group, ESRIG, University of Groningen, Groningen, Netherlands

ARTICLE INFO

Keywords:

Neutron detection
R³B
NeuLAND
Machine Learning
Neural Networks

ABSTRACT

A new Machine Learning algorithm for shower-head identification in the NeuLAND neutron detector is presented. The new algorithm uses densely-connected Deep Neural Networks (DNNs) to properly classify events and clusters, which allows accurate reconstruction of the 4-momenta of the detected neutrons. As data-events recorded with NeuLAND vary quite a lot in size, and not all emitted neutrons always produce signals in the detector, careful pre- and post-processing of the data turned out to be required for letting the DNNs be successful in their classifications. However, after properly implementing these procedures, the new algorithm offers a better efficiency than previously-used algorithms in virtually all investigated scenarios. However, the newly-developed algorithm (as well as previous ones) suffers from systematic uncertainties. These uncertainties mainly arise from the physics lists used in the Geant4 simulations to train the DNNs. They are particularly large for the neutron energy range around 200 MeV and for NeuLAND configurations of few double-planes (slimmed down version of the detector). The accuracy improves with a larger number of double-planes. Furthermore, both model improvements and accurate benchmarks are needed for the currently used Geant4 physics lists to reduce the systematic uncertainties of the new algorithm for high-precision studies. Further improvement of the present DNN algorithm is also needed, especially for experiments that require high precision in the neutron scattering angle reconstruction. However, it seems unlikely that this improvement can be realized using only NeuLAND data.

1. Introduction

The R³B setup (Reactions with Radioactive Relativistic Beams setup) is a multi-purpose experimental setup used to study nuclear structure properties of short-lived isotopes [1]. The setup will be located at the high-energy branch of the Super FRagment Separator (Super-FRS) [2] of the Facility for Antiproton and Ion Research (FAIR) [3] near Darmstadt in Germany.

The heart of the R³B setup consists of a fixed target, at which a secondary beam generated in the Super-FRS is directed. The general goal of the R³B experiment is to provide a kinematically complete reconstruction of all particles participating in the reaction [1], so that the nuclear structure of the beam isotope (which could be very short-lived) can be studied. In order to accomplish this, the fixed target is surrounded by many different detector systems (see Fig. 1).

For most R³B experiments, the target is surrounded by silicon strip vertex detectors [4] to measure the charged-particle tracks close to the target. The CALIFA (CALorimeter for In-Flight detection of gamma-rays and high energy charged pArticles) [5] system encloses the vacuum chamber housing the silicon detectors. CALIFA measures gamma-rays

from the decay of excited nuclei and light-charged particles that are produced at large angles. The heavier charged-particles produced at the reaction are usually (strongly) boosted in the forward direction and travel through the superconducting dipole magnet GLAD (Gsi Large Dipole magnet) [6] downstream of the target. This magnet allows for spectrometric analysis of these charged particles. A complex tracking system of many detectors located both upstream and downstream of GLAD [7] is used to measure the tracks of these heavier charged-particles. Several of these detectors are located inside the vacuum chamber downstream of GLAD, but some are also located at the end of the tube downstream of the vacuum chamber. Finally, neutrons produced at the target, which are generally also very forward-boosted, travel right through GLAD and are detected by NeuLAND (Neu Large-Area Neutron Detector) [8].

NeuLAND [8] is a Time-of-Flight spectrometer for the detection of fast neutrons in the range 200 MeV – 1000 MeV [10]. The detector has a modular design and consists almost entirely of active RP408 scintillator material [11], which is a type of BC408 [12]. The neutrons that should be detected can undergo hadronic scattering within the scintillators

* Corresponding authors.

E-mail addresses: c.a.douma@rug.nl (C.A. Douma), n.kalantar-nayestanaki@rug.nl (N. Kalantar-Nayestanaki).

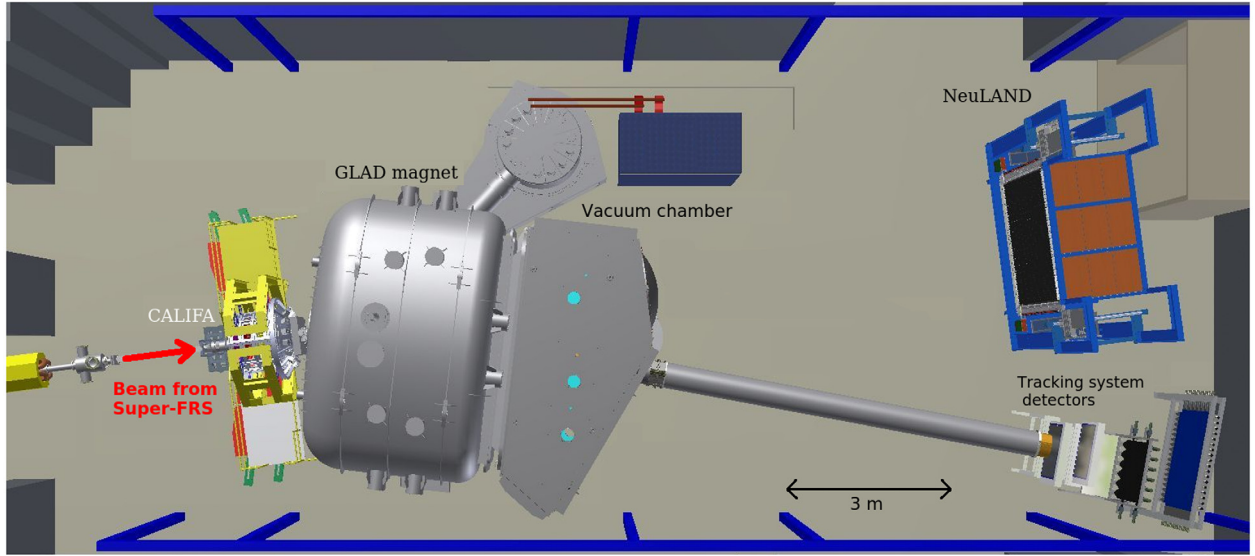


Fig. 1. Overview of the first version of the R^3B setup in Cave C at GSI [9], used with permission.

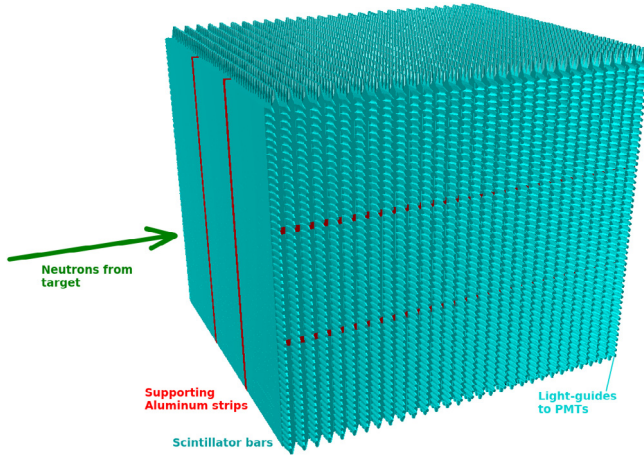


Fig. 2. The full 30 dp NeuLAND geometry, as it was considered in our simulations.

of NeuLAND, which can lead to the production of secondary charged particles. These secondary charged particles are then detected through their scintillation light [10] and their energy and time of arrival are registered by the corresponding electronics.

Each NeuLAND module, called a double-plane (dp), consists of 50 horizontally oriented scintillators of $5\text{ cm} \times 5\text{ cm} \times 250\text{ cm}$, followed by 50 such scintillators in vertical orientation. Each scintillator is read out by two Photo-Multiplier Tubes (PMTs) (one at each endpoint) and is held in place with two 5 cm wide and 0.5 mm thick aluminum strips (see Fig. 2). From these specifications, it follows that each module (dp) has an active area of $2.5\text{ m} \times 2.5\text{ m}$ and a thickness of 10 cm. The NeuLAND design goal is to have 30 dp. However, in the present situation (at the publication date of this work), funding is only secured for 16 dp and only 8 dp are currently mounted in the R^3B setup (see Fig. 3).

The problem that we wish to address in this paper, is that a single high-energy neutron can create a complex shower of secondary particles (both charged and uncharged) in NeuLAND, which can produce signals in any number of scintillators between one and a hundred. This raises the issue which of these signals corresponds to the head of the shower; the first signal generated by the particle shower. This shower head is the information we need from NeuLAND [13], because together with the time and position of the reaction at the target, the time and

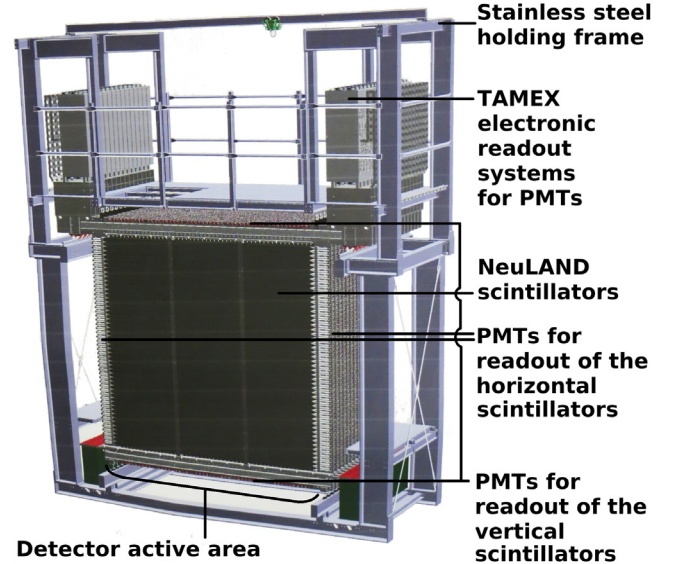


Fig. 3. The current state of the construction of the NeuLAND detector (8 dp) [10]; figure used with permission.

position of the shower head allows us to reconstruct the 4-momentum vector of the neutron [8].

The problem of finding the shower head(s) among all the scintillator signals in NeuLAND is challenging. Especially in the situation where multiple neutrons have to be detected in coincidence, solutions are far from trivial because of two reasons: (1) it is not (always) known a priori how many neutrons have impinged on the detector and (2) showers from distinct neutrons tend to overlap quite often. These problems were already addressed in Refs. [8,13]. However, the solutions proposed there have typical shortcomings in neutron detection, mainly in terms of efficiency (how often one succeeds in correctly identifying the shower heads). A good efficiency for identifying the shower heads is crucial, because this determines the required beam-time for an experiment. For this reason, we propose a newly-developed Machine Learning (ML) algorithm [14] to find the shower heads in NeuLAND.

The methodology of the new ML algorithm is discussed in Section 2. The performance of this algorithm (mainly in terms of efficiency)

and its systematic uncertainties are discussed in Sections 3 and 4. In Section 5, future possibilities for further improvements on top of the new algorithm are discussed. Finally, conclusions and a summary are presented in Section 6.

2. Methodology

2.1. Data generation and preparation

In order to explain the implementation of the ML algorithm in the NeuLAND data-analysis procedure, a brief overview of this procedure is given. This procedure is explained in Fig. 4. Central in this scheme are the NeuLAND “Hits”. These are tuples of 5 numbers: $\{E, t, x, y, z\}$. Within one event, each scintillator that produced a signal is assigned exactly one such a 5-tuple. t and E stand for the energy deposition and the time of the scintillator hit, as reconstructed from the PMT pulses. The position coordinate (either x or y) along the scintillator is also obtained from these PMT pulses. The other two position coordinates are obtained from the geometric position of the scintillator. Together, these 5-tuples (NeuLAND “Hits”) contain all relevant information from NeuLAND.

The “Hits” can either be obtained from the experiment, or from simulation. In both cases, the information from all PMT pulses generated in NeuLAND is first converted to QDC ((Q)charge-to-Digital Convert) values and TDC (Time-to-Digital Convert) values. Next, a “Hit” is produced for each scintillator in NeuLAND where a PMT pulse was generated at both ends. In such a “Hit”, E is obtained as the geometrical mean of the two QDC values of those two PMT pulses [8]. t is obtained as the standard mean of the two corresponding TDC values. The position coordinate along the scintillator is obtained from the difference of those same TDC values.

To obtain the “Hits” from simulation, the PMT pulses were phenomenologically obtained from the outcome of the Monte Carlo transport. The corresponding TDC values were calculated with respect to the beginning of the event, which is exactly known in a simulation. This process is called digitization and it accounts for several experimental effects in the PMT pulses, such as time resolution ($\sigma = 150$ ps), energy resolution ($\sigma = 50$ keV), detection threshold (1 MeV of deposited energy for virtually all experiments), saturation effects, Birk’s law, and light attenuation. For more details on how the PMT pulses were calculated, the interested reader is referred to Sect. 7.2 in Ref. [15]. The resolution effects result in uncertainties in the “Hit”-values of $\sigma = 212$ ps for the t -value and $\sigma = 3$ cm for the position coordinate along the scintillator.

In this work, the simulations were carried out by R3BRoot [16,17], which is an integral framework for simulation and data analysis of the R³B experiment. For our simulations, the NeuLAND geometry, as discussed in Section 1, was used (with a distance of 14 m between NeuLAND and the target position). Individual scintillators were modeled in agreement with Refs. [10,13]. For the physics list, one of the following two reference physics lists was used: QGSP_INCLXX_HP or QGSP_BERT_HP. Both of these physics lists have been benchmarked against experimental data [13] and reality was found to be about halfway between them. As event generator, the traditional NeuLAND simulation files of the ¹³²Sn breakup reaction with a relative energy of 500 keV were used [13]. The relative energy between the neutrons of a single event is defined as the invariant mass of those neutrons together minus the sum of the individual neutron masses: $E_{rel} = |\sum p_i^\mu| - \sum m_i$. The invariant mass is obtained by first adding the 4-momenta of all neutrons within the event, and then taking the length of the resulting 4-momentum vector. As such, the relative energy is a measure for the excitation energy of the recoil nucleus (although not exactly the same).

When the “Hits” are obtained from the experiment, the TDC values are measured with respect to a common-stop signal. However, as not all data channels have exactly the same processing speed and the common-stop signal usually does not correspond to the beginning of the event, the TDC values have to be synchronized before they can be used to

Table 1

Total cross sections in millibarn for a neutron beam at hydrogen and carbon targets according to the physics lists INCLXX and Bertini (obtained from simulation).

Neutron energy:	Hydrogen target INCLXX	Hydrogen target Bertini	Carbon target INCLXX	Carbon target Bertini
200 MeV	76.1	78.3	652	570
600 MeV	81.7	80.7	623	554
1000 MeV	85.2	82.8	691	632

obtain “Hits” that can be compared to the simulation. Since for all NeuLAND experiments, rates are expected to be low enough (< 1 MHz) so that different events do not overlap, this synchronization can be handled event-by-event. Likewise, QDC values have to be calibrated (event-by-event) before they can be used to obtain such “Hits”. This calibration and synchronization procedure is also used to correct for experimental effects that were not considered in the digitization process, such as biases and walk effects. This procedure to first apply such corrections to the experimental data and then obtain “Hits” for further data processing, is what is typically used in NeuLAND experiments such as Ref. [18].

A full discussion of the two considered physics lists QGSP_INCLXX_HP (INCLXX for short) and QGSP_BERT_HP (Bertini for short) used is beyond the scope of this work, but we shall briefly discuss the most relevant differences between them. NeuLAND is a neutron detector of BC408 [8], a CH₂-organic scintillator. Hence, the most important parameters in the physics list are the neutron-to-hydrogen and neutron-to-carbon cross sections. These cross sections have been tabulated in Table 1 for various neutron energies. These cross sections were obtained by directly extracting them from simulations. They are not dominated by one or two specific reaction types, but are a sum of many different reactions.

The Bertini physics list (QGSP_BERT_HP) is based on the Bertini cascade model developed to simulate high-energy particle physics [19]. On the other hand, the INCLXX physics list (QGSP_INCLXX_HP) is an experimental/phenomenological physics list developed for simulating nuclear physics experiments in the intermediate energy range [20]. As INCLXX is dedicated to nuclear physics, it explicitly takes the production of secondary light nuclei into account, such as deuterium, tritium and helium-3. Bertini does not do this, which is the main reason why the neutron-to-carbon cross sections are significantly lower for Bertini. Both Bertini and INCLXX do take the production of secondary alphas into account. Energy spectra of the produced secondary protons do not differ very much. However, as our benchmark in Ref. [13] is about halfway between the two physics lists, we conclude that INCLXX overestimates the production of light nuclei, while Bertini underestimates this.

2.2. Multiplicity determination

After the “Hits” are identified, they are first clustered together according to the procedure outlined in Refs. [13,15]. This procedure assigns “Hits” that have $\Delta x \leq 7.5$ cm, $\Delta y \leq 7.5$ cm, $\Delta z \leq 7.5$ cm and $\Delta t \leq 1.0$ ns to the same cluster. All these clusters (even those of a single “Hit”) are then considered in the subsequent analysis. The next step (see Fig. 4) is to determine the multiplicity of the event (the number of neutrons that came from the target). For this step, we propose the use of ML. We have chosen to use of a Deep Neural Network for this, because this seemed to be the fastest option in terms of CPU processing time. A Convolutional Neural Network (CNN) has also been investigated for the multiplicity determination [21], but while this method was found to have about the same performance (in terms of efficiency) as our work (see Appendix), its CPU processing time was much slower.

The DNN was equipped with three input neurons per scintillator. These input neurons contain the “Hit” energy deposition E , the time of the scintillator “Hit” t and the position coordinate of the “Hit” along

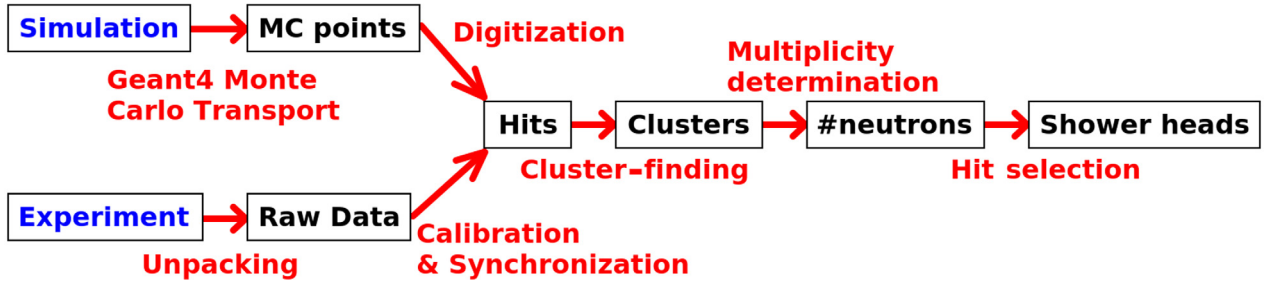


Fig. 4. The Different steps in the NeuLAND data analysis. The upper branch illustrates the steps used for simulated data and the lower branch illustrates the steps used for measured data.

the scintillator (either x or y). For scintillators without a hit, three zeros were given as inputs. This way, all “Hit” information (which is all relevant information NeuLAND has to offer [8]) is fed into the DNN. The information of the other two position coordinates from the “Hits” (which are determined from the geometrical position of the scintillator) is contained in the positions of the input neurons in the network, because the same input neuron always correspond to the same scintillator. Since it was known from Ref. [21] that adding the total energy deposition in the full detector per event and the total number of clusters per event as extra inputs to the DNN improved efficiency, these two quantities were also given to the DNN as inputs. This resulted in 9002 input neurons for the NeuLAND design case (30 dp of 100 scintillators each) [8]. All inputs were linearly scaled to $[0, 1]$ before they were fed into the DNN. t -values (“Hit”-times) above a certain threshold were artificially put to zero. This was done, because a small fraction of the “Hits” ($< 0.1\%$) has very large t -values of the order of 10^4 ns. Such rare and very large t -values were found to be extremely disruptive to the DNN training procedure and, therefore, had to be removed. The proper time threshold depends on the simulated scenario. For the design-case of 30 dp, a neutron energy of 600 MeV and a distance between NeuLAND’s front-end and the target of 14 m, 100 ns was found to be a good threshold (in this situation the neutrons need 72 ns to travel to the rear end of NeuLAND).

As this DNN is supposed to classify a given event according to its multiplicity (the number of neutrons that came from the target), a separate output neuron for each possible multiplicity was implemented. Typical NeuLAND simulations consider multiplicities up to five [8,13], which would result in five output neurons. These output neurons were equipped with the SoftMax activation function [22], as this transforms the network output into a probability per multiplicity. As the best method for multiplicity determination is still unknown at the time of this work, a densely connected network was considered with the ReLU activation function on each hidden neuron, as this a-priori allows any combination of the input data to be considered as a possibility to determine the output. The ReLU activation function is defined as $f(x) = x$ for $x \geq 0$ and $f(x) = 0$ for $x < 0$ [23].

The network was implemented with the Keras [23] user-interface to the TensorFlow [24] framework. Training was done with Supervised Learning (SL) [25] using the ADGRAD [26] minimization algorithm (learning rate 0.001, $\epsilon = 0$ and zero decay) and the Categorical Cross Entropy [27] minimization function. The number of hidden layers was 2, with 9000 (first layer) and 1200 (second layer) neurons each. These parameters (minimization algorithm, minimization function, number of hidden layers and number of neurons per hidden layer) were determined by optimization. This procedure is discussed in detail in the Appendix.

One million simulated events were generated for the training (200.000 per multiplicity). During the training, only events were used where the number of neutrons detected by NeuLAND was identical to the number of neutrons that came from the target. This was done, because it would be incorrect to train the DNN on events where it will never be able to determine the correct answer, which is the case

when not all information is available: the situation where not all neutrons are detected by NeuLAND. The remaining events (that passed this condition) were subdivided into batches of 1000 events and the network weights were updated after each of these batches. The number of one million events and the batch size of 1000 were chosen as such, because it was observed that these numbers were sufficient to reach a saturated network accuracy under almost all circumstances within 2 epochs. Since the multiplicity condition implies that events with higher multiplicities are more often discarded, the undiscarded higher-multiplicity events were given larger weights in the minimization function to prevent the DNN from becoming biased. This bias could result in a loss from roughly 25% (30 dp) to several hundred percent (few dp) in the proper reconstruction of multiplicity-five events.

2.3. “Hit” selection

In the last step in Fig. 4, the “Hit” selection, the actual shower heads have to be identified. For this, we also propose the use of ML. The network design for the “Hit” selection is based on the successes obtained in the multiplicity determination. The network for the “Hit” selection was chosen as a densely connected DNN with 14 input neurons, two output neurons with the SoftMax activation function and 12 hidden layers of 200 neurons with the ReLU activation function each. An event is then sent through this network cluster-by-cluster and a different copy of this network was trained for each multiplicity under consideration. The 14 input neurons contain different cluster properties like Time-Of-Flight, relativistic beta, energy deposition, number of “Hits” in the cluster, etc. (all inspired by Ref. [13]). The two output neurons give the two probabilities of the cluster containing at least one shower head, and of not containing any shower heads at all, respectively.

Using the network output, a score is computed for each cluster as the difference between the two output neurons. Subsequently, all clusters in the event are sorted according to this score. Then, all clusters are discarded, except for the ones with the highest score. The number of clusters kept is decided by the multiplicity DNN. Subsequently, the “Hits” in the clusters which have the shortest TOF define the shower heads. The Supervised Learning for the “Hit” selection was done similarly to the multiplicity determination.

The full simulation and network training as described in all sections above required a computational time of roughly 6 h when the configuration of 30 dp with a neutron beam energy of 600 MeV was used. The generation of simulated data was done using an i7-8750H CPU and 16 GB of RAM memory. The network training was done using a 4 GB GPU (NVIDIA GeForce GTX 1050 Ti 4 GB). The disk-storage requirements for this computation were about 250 GB. The computer code of the new ML algorithm (both the multiplicity determination and the “Hit” section) is available as public-domain software [28].

2.4. Reference algorithms

To evaluate the performance of our new ML algorithm (denoted DNN algorithm in the following), its efficiency is compared to two

other shower-head identification algorithms in the upcoming sections: the TDR algorithm and the “Perfect Tracking” algorithm. Note that from now on, the notion ‘DNN algorithm’ stands for the combination of multiplicity determination and “Hit” selection, both handled by ML. The “Perfect Tracking” algorithm relies on information from the Geant4 simulation. It traces the particle showers created by each individual neutron to find out whether this shower produced any NeuLAND “Hits” (see Fig. 4). If “Hits” are produced, the neutron is assigned a shower head: the first produced “Hit”. Otherwise, the neutron is assigned no shower head at all.

Since the “Perfect Tracking” algorithm relies on the ‘true’ particle showers from the Geant4 simulation, it will always provide the best possible shower-head identification. For this reason, its outcome was used as the ‘correct’ output for the Supervised Learning of the DNNs. However, despite its name the “Perfect Tracking” algorithm does not always result in perfect efficiency. This is because sometimes, a neutron may just not interact (sufficiently) with NeuLAND to produce any “Hits” [13]. Hence, a comparison between the DNN algorithm and the “Perfect Tracking” algorithm can reveal which part of the DNN algorithm efficiency is caused by imperfections in the algorithm and which part is caused by external factors (the lack of proper neutron interactions leading to “Hits”).

The TDR algorithm refers to the algorithm originally proposed in the NeuLAND Technical Design Report (TDR) [8]. This procedure is illustrated in Fig. 5 for a situation of 30 dp and 600 MeV neutrons. Here, the multiplicity is determined by plotting the number of clusters per event versus the total energy deposition in that event and imposing linear “decision” cuts (the envelopes defined by the diagonal black lines in Fig. 5). The slope is the same for all cuts, but the distance between the cuts is different for different multiplicities. The slope and distances are optimized using simulation data. Subsequently, an event is assigned a multiplicity depending on where it is located between the “decision” cuts.

We would like to note that the situation where all neutrons within a single event have (almost) the same energy is actually quite common [8]. In most R^3B experiments, the beam consists of a heavier nucleus of interest with an energy of several hundred MeV per nucleon. This nucleus is then studied in inverse kinematics by impinging it on a very light fixed R^3B -target. As such, the knocked-out neutrons typically have energies of several hundred MeV as well, but with variations of the order of only a few MeV (determined by the shell from which they are removed). Hence, for such experiments, our algorithms can be trained/optimized with mono-energetic neutrons.

The “Hit” selection of the TDR method is exactly as discussed for the DNN algorithm in Section 2.3, only clusters are not sorted according to some DNN-computed score, but according to their so-called R -value, defined as:

$$R = -^{10} \log \left(\frac{|\beta^{\text{cluster}} - \beta^{\text{beam}}|}{E_{\text{dep}}^{\text{cluster}}} \right), \quad (1)$$

where $E_{\text{dep}}^{\text{cluster}}$ is the total energy deposition of all “Hits” in the cluster, β^{beam} is the beam speed divided by the speed of light and β^{cluster} is the presumed neutron speed (divided by the speed of light) between the target reaction and the minimum TOF “Hit” in the cluster. Further details on this method can be found in Refs. [8,13,15].

The TDR algorithm was the first shower-head identification algorithm proposed for NeuLAND [8] and for most situations, its efficiency is better than all other shower-head identification algorithms known today [13], except for the ones that use Machine Learning. For this reason, it serves as an excellent benchmark to judge the efficiency of our new DNN algorithm.

2.5. Sources of uncertainties

Special attention should be paid to the fact that all shower-head identification algorithms discussed so far (including DNN, TDR and

“Perfect Tracking”) [13] rely on the use of simulation data to optimize certain decision parameters. Since simulations generally do not perfectly agree with experimental conditions, this dependence introduces systematic uncertainties to the application of all algorithms. These systematic uncertainties come from three different sources (the three main components in a GEANT4 simulation): the geometry, the event generator and the physics list. The systematic uncertainties from the first two sources can generally be suppressed by closely matching the geometry and the event generator to the experimental conditions, although one should keep in mind that this requires a re-optimization (or re-training) for each new experiment.

However, finding a physics list that closely matches the reality with the desired precision is still an open issue for NeuLAND experiments [10,13]. The alternative to using simulation data would be to perform measurement of reactions such as the break-up of the deuteron which would produce neutrons with known energies and angles once the proton is tagged with a high precision. However, these measurements have to be done for a large number of energies to be useful and as such are not a very viable alternative for the large range of energies desired. Because of the fact that the geometry and the event generator contributions to the systematic uncertainties can generally be made much smaller than the physics list contribution, we have only concentrated on the quantitative study of the physics list contribution to the systematic uncertainties. For this reason, we repeated all simulations for this work four times and changed the physics list between QGSP_INCLXX_HP and QGSP_BERT_HP at both the optimization (or training) and the validation level. The mean of the four outcomes was then considered as the algorithm performance and the largest difference between the mean and each of the four individual outcomes was used as our estimation of the physics list uncertainty.

As reality was determined to be halfway between two physics lists [13], one could, in principle, also compose a realistic training set by mixing events of the two physics lists. However, as the benchmarking of the two physics lists in Ref. [13] was only done for neutron energies of 110 MeV and 250 MeV and only for the NeuLAND detection efficiency, this procedure is unreliable. The appropriate weight factors of the two physics lists are indeed roughly 50% in that situation, but there is no experimental evidence that these weight factors remain the same at higher neutron energies, or for other observables. For this reason, we have decided not to pursue this idea and estimate the physics list uncertainties as described above. However, once appropriate weight factors have been measured over the full range of neutron energies, event mixing may prove to be a powerful tool in the future to reduce the physics-list uncertainties and/or to improve the physics-list models.

3. Performance of multiplicity determination

The efficiency of the NeuLAND multiplicity determination is shown in Figs. 6–8. Results were computed for typical neutron energies of 200 MeV, 600 MeV and 1000 MeV, as is commonly done for NeuLAND simulations [8,13] and for NeuLAND configurations of 8 dp, 12 dp, 16 dp, 23 dp and 30 dp. These configurations were chosen, because the configuration with 8 dp is currently (at the publication date of this work) in use, 16 dp have secured funding at present and 30 dp is the design goal (see Section 1). The configurations of 12 dp and 23 dp were added for upcoming experimental proposals (at the publication date of this work). The efficiency is shown as a percentage of how often the correct number of neutrons in the event (that came from the target) is established. Results are shown as solid lines for the Perfect tracking algorithm, the TDR algorithm and the DNN algorithm (all discussed in Section 2). The physics-list uncertainties of the three algorithms are shown as separate bands (see Section 2). The bands are arbitrarily placed above 100% in the figures and the band widths represent the 2σ physics-list uncertainties. Note that, as the Perfect tracking algorithm assigns a shower head to each neutron that produced at least one “Hit” (see Section 2.4), the black curves essentially describe the probability

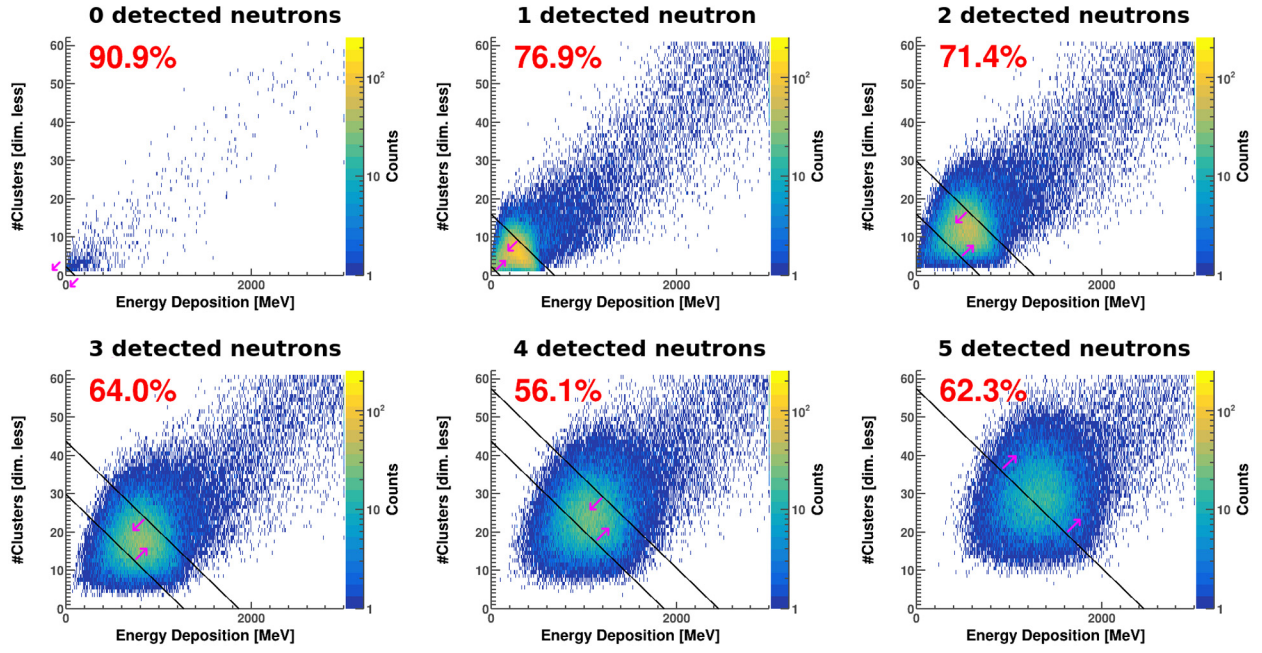


Fig. 5. Illustration of the TDR algorithm for multiplicity determination in the situation for a 30 dp NeuLAND and a 600 MeV neutron energy. The number of detected neutrons refers to the number of assigned shower heads by the “Perfect Tracking” algorithm. The red numbers in the upper left corners refer to the fraction of events in each histogram that falls between the respective cuts (black lines) in that histogram. See text for further details.

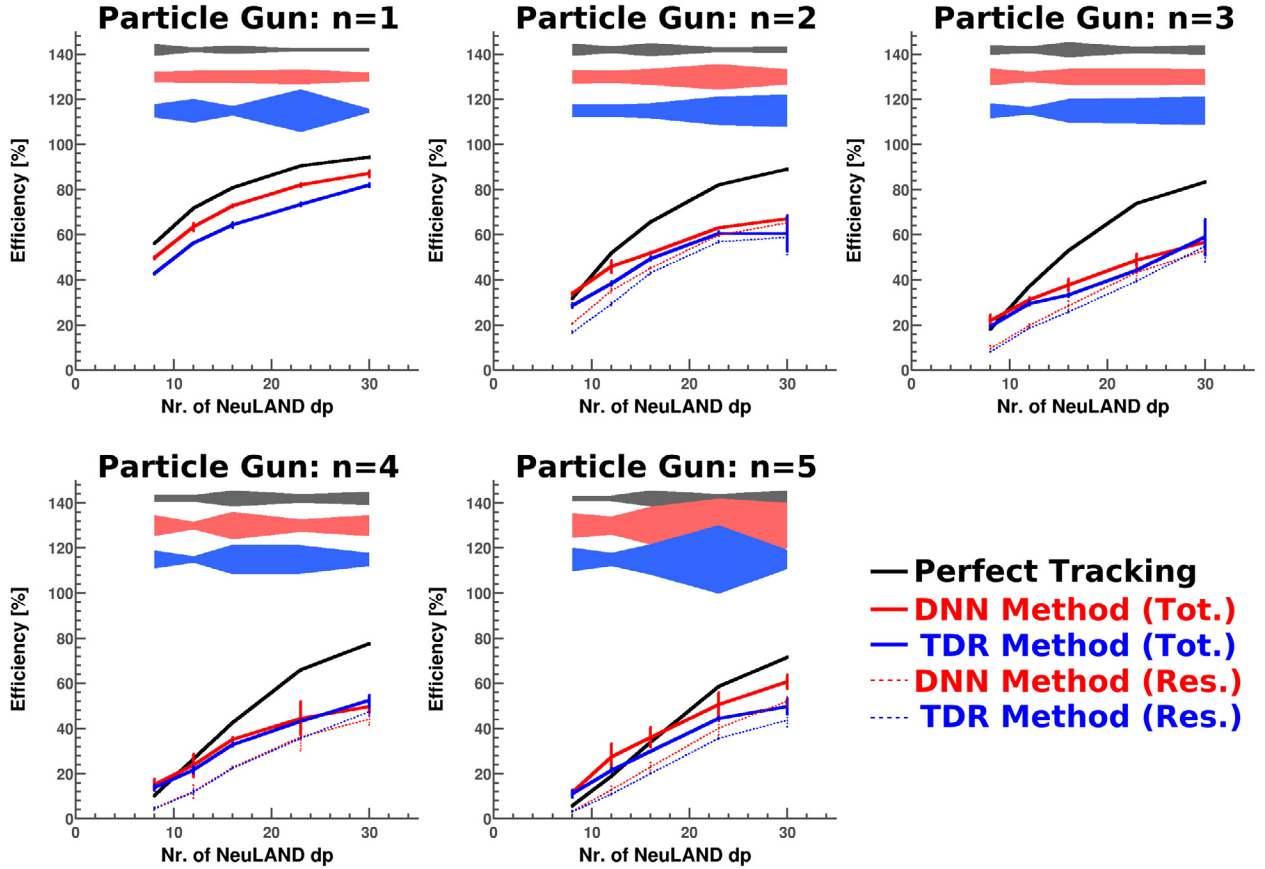


Fig. 6. Performance of the multiplicity determination in NeuLAND. For different neutron multiplicities fired by the particle gun, the percentage of events where that multiplicity was established, is plotted against the number of NeuLAND double-planes (dp). The full results are shown as solid lines for the TDR algorithm, the new DNN algorithm and for the “Perfect Tracking” algorithm, which gives the highest possible performance. The dashed lines correspond to the restricted scenario; the part of events where false-positives are excluded. The simulations are performed with a neutron energy of 200 MeV. 200,000 events were simulated per multiplicity-figure. See text for further details.

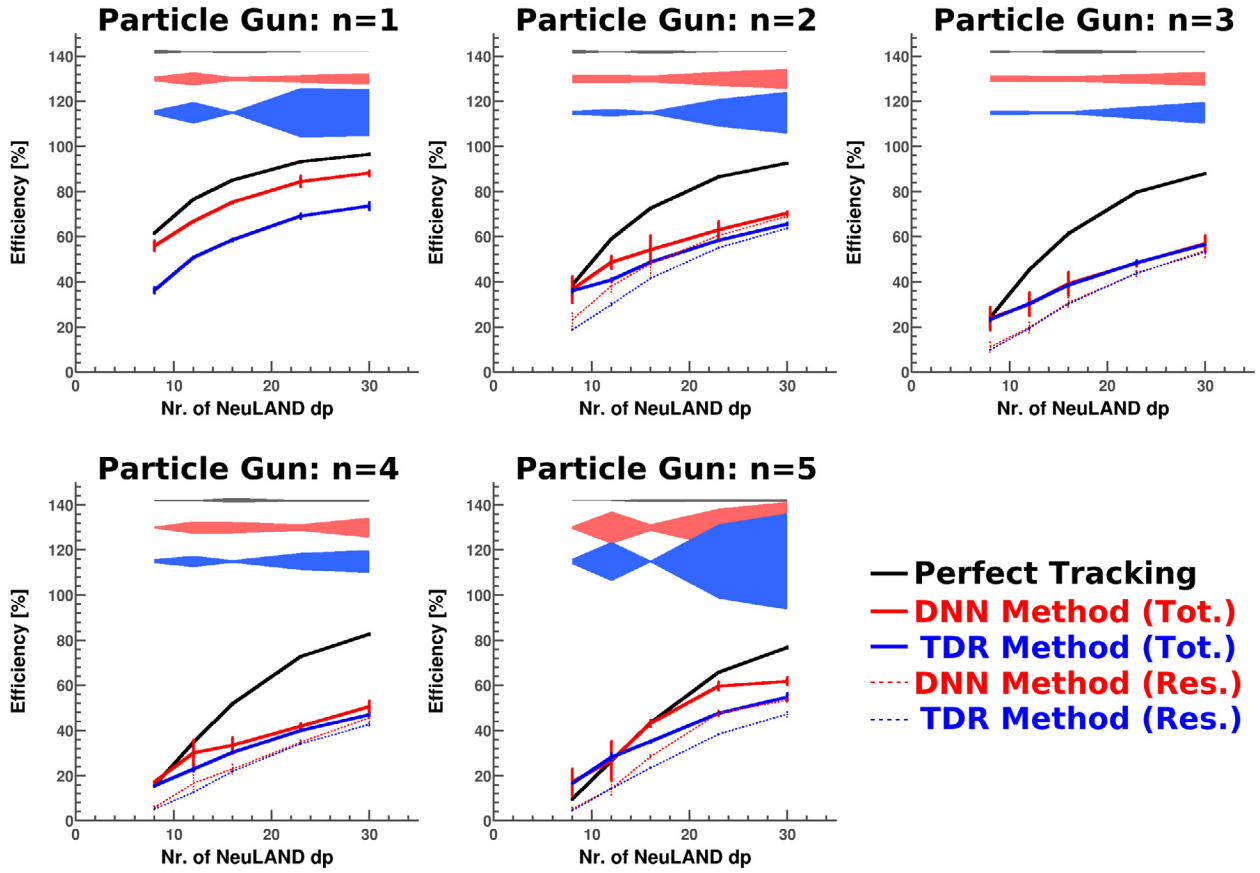


Fig. 7. Same as Fig. 6, but for a neutron energy of 600 MeV.

that all neutrons fired by the particle gun interacted with NeuLAND and were also detected by NeuLAND.

As can be seen in Figs. 6–8, the TDR and the DNN results (solid lines) are sometimes higher than the results obtained by the “Perfect Tracking” algorithm. Since the “Perfect Tracking” algorithm contains the best possible shower-head identification (see Section 2), this may seem like a contradiction. However, some events may be ‘accidentally’ assigned the correct multiplicity by either the TDR or the DNN algorithm. In order to understand how this works, consider an event where four neutrons come from the target. In this case, it is possible that three neutrons produce “Hits” in NeuLAND and that the fourth one does not. Subsequently, either the TDR or the DNN algorithm may wrongly classify the “Hits” produced by three neutrons as a four-neutron event (consider, for example, the part of the blob above the cuts in the 3n-figure of Fig. 5). Hence, it is possible for the TDR and DNN algorithms that the correct multiplicity of the event is found, while not all neutrons were detected by NeuLAND. For these events it is impossible to come up with a correct shower head for all neutrons in the “Hit” selection stage. This phenomenon is designated as ‘false-positive’ multiplicity assignments. It cannot occur for the “Perfect Tracking” algorithm, as it only assigns shower heads to neutrons that are detected (in the sense that they produced scintillator “Hits”, see Section 2). As a result, it is possible that the TDR and DNN algorithms have a higher multiplicity performance than the “Perfect Tracking” algorithm, as false-positive multiplicity assignments are included. The dashed lines in Figs. 6–8 give the TDR and DNN result when the false-positive multiplicity assignments are excluded (These graphs are designated as the ‘Restricted’ case). However, one should realize that the computation of the dashed lines in Figs. 6–8 requires the use of the “Perfect Tracking” algorithm (to determine how many neutrons were actually detected), which means that they can only be computed for

simulation data. Hence, during an experiment only the solid lines in Figs. 6–8 will be known.

From Figs. 6–8 it can be seen that the problem of false-positive multiplicity assignments gets relatively smaller as the number of double-planes increases. This is due to the fact that the probability of neutrons interacting with NeuLAND increases with the number of dp, which makes the multiplicity determination more accurate.

From Figs. 6–8, we conclude that the DNN performance is higher than the TDR performance for all studied neutron energies, multiplicities and NeuLAND configurations (except for the 30 dp point in the 3n and 4n cases of the 200 MeV neutron energies). However, in many situations (mostly the 3n and 4n cases) the difference in performance is quite small. However, even such a small difference in multiplicity performance has a significant influence on the “Hit” selection performance due to our choice for handling different multiplicities with different “Hit” selection DNNs (see Section 2).

Another advantage of the new algorithm is that in several cases the physics-list uncertainties in Figs. 6–8 are smaller for the DNN algorithm than for the TDR algorithm. The reason for this is that the optimization of the cuts in Fig. 5, which only relies on two inputs (number of clusters and total energy deposition), typically results in a very shallow minimum. On the other hand, the DNN algorithm uses thousands of inputs and, hence, has a less shallow minimum. A shallow minimum causes large fluctuations in the final results when some simulation parameters (like the physics list) are changed.

Figs. 6–8 also reveal some limitations in the use of both the conventional TDR algorithm and the newly developed DNN algorithm. The first limitation is introduced by the physics-list uncertainties, which can sometimes be quite significant (especially at a neutron energy of 200 MeV, see Fig. 6). Hence, accurate multiplicity determination

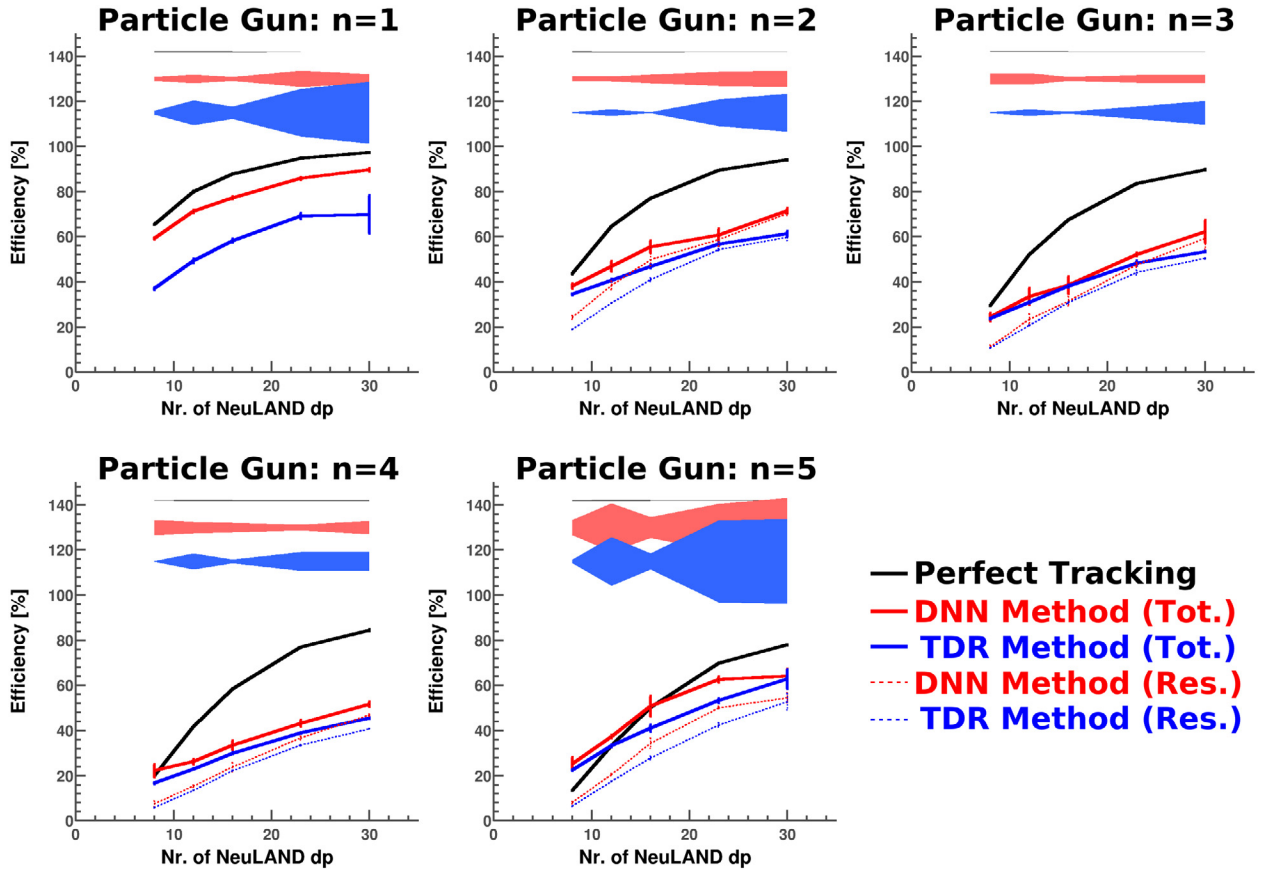


Fig. 8. Same as Fig. 6, but for a neutron energy of 1000 MeV.

will not be possible for experimental data unless these physics-list uncertainties are better understood. This requires both improvements in the models used in the physics lists as well as accurate benchmarks against experimental data. As discussed in Section 2.5, event mixing could also help for this.

As can be seen in Figs. 6–8, the magnitudes of the physics-list uncertainties are a complicated function of the number of dp. This is because any error estimate (like the physics-list uncertainty bands) is influenced by two opposite effects: overlapping particle showers and the neutron interaction probability. As the number of NeuLAND dp increases, so does the probability for a neutron to interact with NeuLAND, which allows for a more accurate multiplicity determination. On the other hand, having more dp also leads to the particles showers (produced by the neutrons) to become more complex, and, as a result, overlap more often. This effect leads to less accuracy in the determination of the multiplicity. Since both of these effects increase with the number of dp, one effect may be slightly more dominant for a certain configuration, while the other effect is slightly more dominant for another configuration. This results in fluctuations in the magnitude of the physics-list uncertainties as a function of the number of dp.

The physics-list uncertainties for both the TDR and DNN algorithms for multiplicity five are relatively large (at all neutron energies). This is because multiplicity five is the highest multiplicity considered in our simulations, meaning that any neutron multiplicity above four will be classified as five. For this reason, multiplicity five suffers from ‘endpoint fluctuations’, which have a large impact on our estimate of the physics-list uncertainties. The endpoint fluctuations can be nicely illustrated with Fig. 5 for the TDR algorithm. Variations in the parameters of the lower cut in any multiplicity window are partially compensated by those same variations in the upper cut of that same window (because all

cuts have the same slope). However, this does not happen for multiplicity five, because it does not have an upper cut. Hence, multiplicity five suffers from larger fluctuations: the endpoint fluctuations. Fortunately, endpoint fluctuations can be easily avoided by training the algorithms (both TDR and DNN) up to one multiplicity higher than what is actually expected in the experiment.

The second limitation of both the conventional TDR algorithm and the newly-developed DNN algorithm is the number of false-positive multiplicity assignments. Since not all neutrons from the target are actually detected for these events, a proper reconstruction of the neutron 4-momenta will not be possible, despite the fact that the correct multiplicity was found (see next section). The number of false-positive multiplicity assignments is particularly large for NeuLAND configurations with a smaller number of dp. The severity of this problem decreases as the number of double-planes increases. From the figures, it can be seen why a choice of 30 double-planes was made in the original design.

From the results presented in this section, we conclude that the DNN algorithm offers advantages over the traditionally-used TDR algorithm, both in terms of efficiency and in terms of physics-list uncertainties (although the advantages are sometimes small). However, the (sometimes large) physics-list uncertainties inhibit a good multiplicity determination, meaning that both model improvements and accurate benchmarks are needed for the physics list (for all neutron energies, but particularly for neutron energies around 200 MeV). As discussed in Section 2.5, event mixing could also help in this. Furthermore, the number of false-positive multiplicity assignments can, in turn, inhibit a good shower-head identification, which is why the number of NeuLAND dp should be as large as possible.

4. “Hit” selection performance

In order to assess the performance of the “Hit” selection step (see Fig. 4), it is necessary to consider the 4-momentum vectors of the neutrons involved [13]. Because of NeuLANDs recent use in a tetra-neutron experiment [29,30] and its design goal for multi-neutron detection capability, a 4-neutron invariant mass spectrum is a good method to assess the “Hit” selection performance. Figs. 9–11 show such spectra for the same neutron energies, NeuLAND configurations and shower-head identification algorithms as studied in Section 3. Four times the mass of the free neutron was subtracted from the total invariant mass, which is why the x-axis is labeled ‘Invariant Mass Difference’. Physics-list uncertainties are, again, shown as separate bands (as in the previous section).

From Figs. 9–11 it is clear that the new DNN algorithm significantly improves the “Hit” selection with respect to the traditionally-used TDR algorithm. In terms of efficiency, the DNN algorithm has about 50% higher performance than the TDR algorithm for 200 MeV neutrons. For higher neutron energies, this is about a factor 3. Since an increase in efficiency was the reason for developing the new DNN algorithm, we conclude that the new DNN algorithm achieves that goal. Moreover, since the difference in performance between the TDR and the DNN algorithms was quite small for the multiplicity determination (see previous section), we also conclude that almost all the improvement shown in Figs. 9–11 is due to the improved “Hit” selection procedure.

Figs. 9–11 also confirm our conclusions made in Section 3 about the limitations due to physics-list uncertainties and the number of NeuLAND dp. The physics-list uncertainties are significant for all neutron energies and are particularly large at 200 MeV (see Fig. 9). Hence, also for extracting more complex observables with NeuLAND, like the invariant mass, a reduction of the physics-list uncertainties by model improvements and more accurate benchmarks seems necessary (especially at lower energies like 200 MeV).

These figures also effectively demonstrate the effects of false-positive multiplicity assignments on measurable observables like the invariant mass. As an example, we take a closer look at the situation in Fig. 9a. In this situation, false-positive multiplicity assignments make up about 60% of the data in this situation (see the 8 dp point in the 4n picture of Fig. 6). This is why the DNN peak has a larger area than the “Perfect-Tracking” peak. Moreover, there is even a significant shift in the peak position between these two algorithms. Since not all four neutrons have been detected in the false-positive multiplicity assignments, the “Hit” selection algorithm is forced to come up with a false fourth shower head. The contributions of these false shower heads to the invariant mass are the cause of the shift. From Figs. 9–11, it can be concluded that such shifts occur for all neutron energies and all NeuLAND configurations. However, the magnitude of the shift decreases when the neutron energy increases (because the physics-list uncertainties decrease) and when the number of NeuLAND dp increases (because the ratio of true to false-positives becomes more favorable). Hence, in order to allow accurate determination of measurable observables with NeuLAND like the invariant mass, we conclude that the number of NeuLAND dp should be as large as possible.

Some NeuLAND configurations also suffer from large statistical uncertainties besides the physics-list uncertainties. These statistical uncertainties are a combination of the traditional Poisson uncertainty and of the simulation initialization. Decision parameters (like the DNN weights and the TDR cuts) are initialized as random numbers and then optimized using a numerical minimization algorithm. However, since the minimization algorithm uses a finite number of iterations and/or the minimization could be trapped in local minima (which happens frequently for DNNs [14]), the outcome of the minimization procedure depends (somewhat) on the initialization. As a result, the statistical uncertainties contain a contribution from this initialization. This contribution was estimated by repeating both the training and the validation twice with the exact same input parameters (but different

random seeds) and is shown in Figs. 9–11 as a band around the histogram-data. This band includes the Poisson uncertainty.

From the limited number of simulations that were performed to estimate the statistical uncertainties, the general trend seems to show that the statistical uncertainties are smaller for the larger number of dp. Looking at Figs. 9–11, it is advised that for measuring observables like a 4-neutron invariant mass with a reasonable accuracy, at least 16 dp are used. However, the number of required double-planes may be different for other neutron multiplicities.

Since false-positive multiplicity assignments and overlapping particle showers cannot occur for one-neutron events, basically any number of NeuLAND dp can be used for one-neutron experiments, provided that sufficient beam-time is available to compensate for the lower neutron detection probability (for 8 dp, this is 50%–60%, see the figures in the previous section). However, for two-neutron events and 8 dp, 40%–50% of the multiplicity assignments are false-positives. Given the results of Fig. 9a where the number of false-positive multiplicity assignments is roughly 60%, we conclude that 8 dp is not enough for two-neutron experiments. However, the number of 40%–50% for two-neutron events drops significantly (to roughly 25%) when going from 8 dp to 12 dp. From Figs. 9a (roughly 60% false-positive multiplicity assignments) and 9c (roughly 25% false-positive multiplicity assignments), we know that such a drop in false-positive multiplicity assignments significantly reduces the shift in observables. Hence, we conclude that 12 dp suffices for two-neutron experiments. Since the ratio of true to false-positive multiplicity assignments is roughly the same for three-neutron events, four-neutron events and five-neutron events, we conclude that for multiplicity above two, 16 dp are needed.

Hence, one can conclude that 8 dp would suffice for experiments where one is interested in only one-neutron detection, 12 dp would be enough in experiments where two neutrons are ejected and for higher neutron multiplicities, a minimum of 16 double planes would be required.

5. Extraction of the neutron scattering angle and its impact on observables

From the results presented in Sections 3 and 4, it can be concluded that the present DNN algorithm offers a significant improvement over the TDR algorithm, mainly at the “Hit” selection stage. However, there is still a significant difference in efficiency between the present DNN algorithm and the “Perfect Tracking” algorithm, which gives the maximally-achievable efficiency of any shower-head identification algorithm. This brings up the question whether the present DNN algorithm could be further improved.

In order to deal with this issue, it is important to realize that a (large) difference between the DNN and “Perfect Tracking” algorithms can have two different reasons: either that the present DNN algorithm has shortcomings which should be dealt with, or that the “Perfect Tracking” algorithm overestimates the maximally-achievable efficiency when other kinematical variables of the experiment are considered. In order to distinguish between these two possibilities, the accuracy of the neutron detection by NeuLAND was investigated. For this purpose, the neutron 4-momentum vectors obtained from the identified shower-heads (using the DNN, TDR and “Perfect Tracking” algorithms) were compared to their counterparts from the Monte Carlo data. Differences between these 4-momentum vectors are typically caused by a deflection of the incident neutron after its first interaction (at Monte-Carlo level) with NeuLAND, while this first interaction itself failed to be detected (due to energy thresholds).

The 4-momentum vector of a neutron is characterized by three parameters: the neutron scattering angles (polar and azimuthal angles at the R³B target) and the kinetic energy. The kinetic energy of the neutron is mainly determined by its TOF, which is extracted from the shower head. Since the TOF is dominated by the distance from the target to the detector, the exact position of the shower head will have

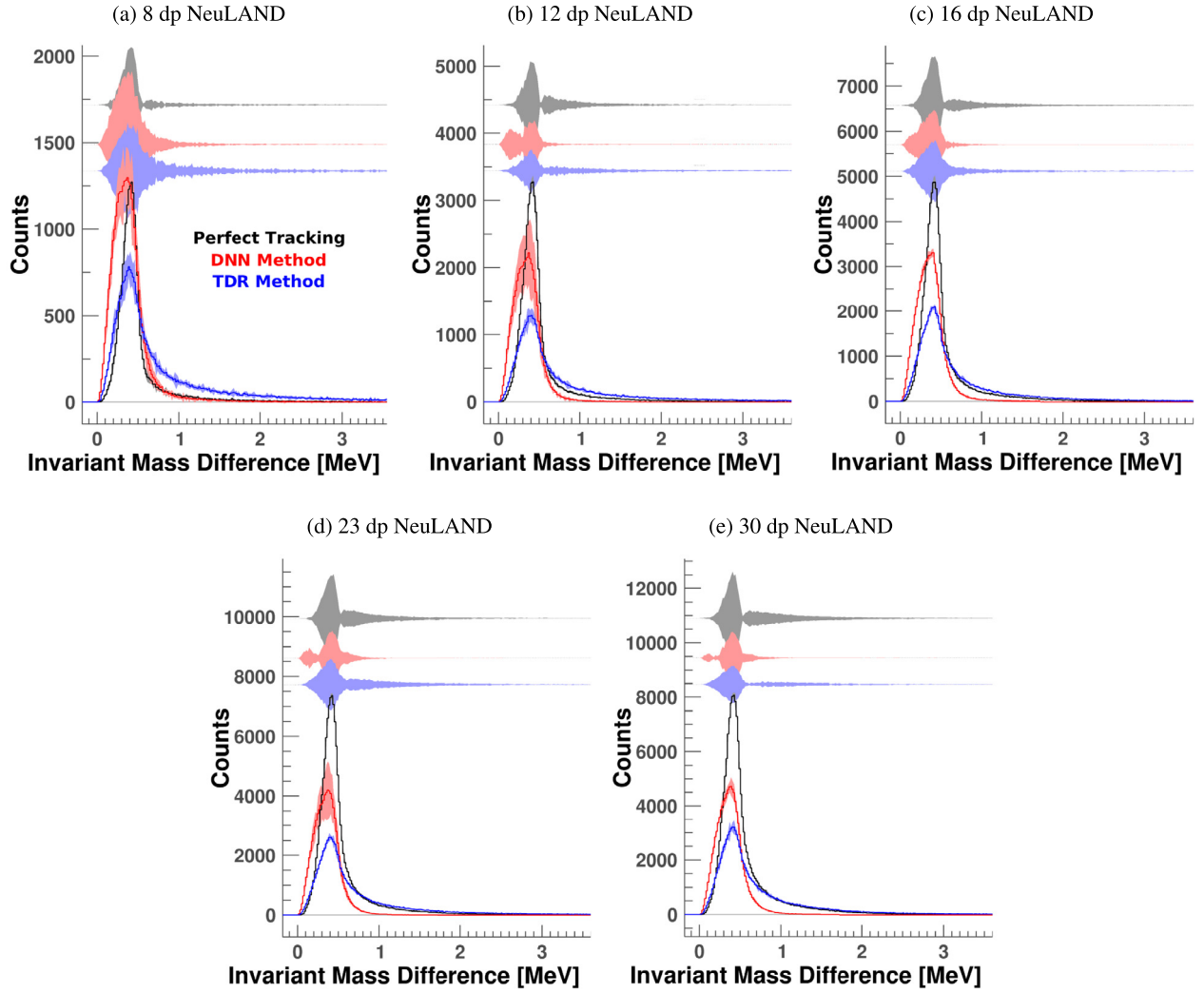


Fig. 9. Invariant Mass spectra for 4 neutrons, each carrying an energy of 200 MeV. The spectra (a, b, c, d and e) were reconstructed from different NeuLAND configurations (in terms of dp). Four times the mass of the free neutron was subtracted from the total invariant mass, which is why the figures are labeled ‘Invariant Mass Difference’. The results of the TDR, the DNN and the “Perfect Tracking” algorithms are plotted, together with their statistical uncertainties (around the data) and their systematic uncertainties from the physics list (separate bands). To obtain the results for each curve, 200,000 $4n$ events were simulated. See text for further details.

little effect on the extracted kinetic energy. For the determination of the scattering angles, however, the exact position of the shower head within the detection volume turns out to be important, as will be shown in this section.

In order to quantitatively study the uncertainties in the neutron scattering angles, a restriction was imposed on the “Perfect Tracking” algorithm. If the angle between the Monte Carlo neutron 4-momentum vector and the neutron 4-momentum vector obtained from the corresponding shower-head candidate was larger than a certain value (a designated accuracy, $\Delta\theta_{\max}$), the neutron was assigned no shower head in the “Perfect Tracking” analysis. If this angle was smaller than $\Delta\theta_{\max}$, a shower head was assigned (in the “Perfect Tracking” analysis) in the same way as described in Section 2.4. Up to this point in our analysis, no such restrictions were considered for the “Perfect Tracking” algorithm, meaning that all results presented so far correspond to an indefinite uncertainty in the scattering angles. The consequences of imposing this restriction on the “Perfect Tracking” algorithm were studied for an average neutron energy of 600 MeV and a NeuLAND configuration of 30 dp (the design goal). The distance between the target and the front-face of NeuLAND is 14 m and the physics list used for this study is QGSP_INCLXX_HP. The effects of imposing a restriction on the error in the neutron scattering angles for the “Perfect Tracking” algorithm are shown in Fig. 12. The conventions used in Figs. 6–8 also apply to Fig. 12.

From Fig. 12, we conclude that the efficiency of the “Perfect Tracking” algorithm drops with smaller $\Delta\theta_{\max}$. Hence, for experiments that require good precision (small $\Delta\theta_{\max}$), the “Perfect Tracking” results presented in Sections 3 and 4 (which correspond to $\Delta\theta_{\max} \rightarrow \infty$) are indeed overestimations of the efficiency that can maximally be achieved by any shower-head identification algorithm. The “Perfect Tracking” curves that were shown in the previous sections correspond to $\Delta\theta_{\max} \rightarrow \infty$ and represent the upper limit of this efficiency. This situation corresponds, for example, to the charge-exchange reactions discussed in Ref. [31], where the ejectile had to be detected with a precision of at least 0.5° . For experiments that can afford a larger $\Delta\theta_{\max}$, the “Perfect Tracking” results presented in Sections 3 and 4 are reasonably close to the efficiency that can maximally be achieved.

Although the restriction of $\Delta\theta_{\max}$ was only imposed on the “Perfect Tracking” algorithm, the DNN and TDR results in Fig. 12 are also affected by it, because these algorithms use the “Perfect Tracking” result for their training (see Section 2.4). During an experiment only the solid lines of the DNN and TDR algorithms in Fig. 12 can be observed (see Section 3). Therefore, we conclude that only for experiments where a large $\Delta\theta_{\max}$ is sufficient, the multiplicity classification can be done with reasonable efficiency. Experiments that require a small $\Delta\theta_{\max}$, will have a significant amount of false positives in their data (even for 30 dp), which cannot be removed. For example, Fig. 12 shows that about

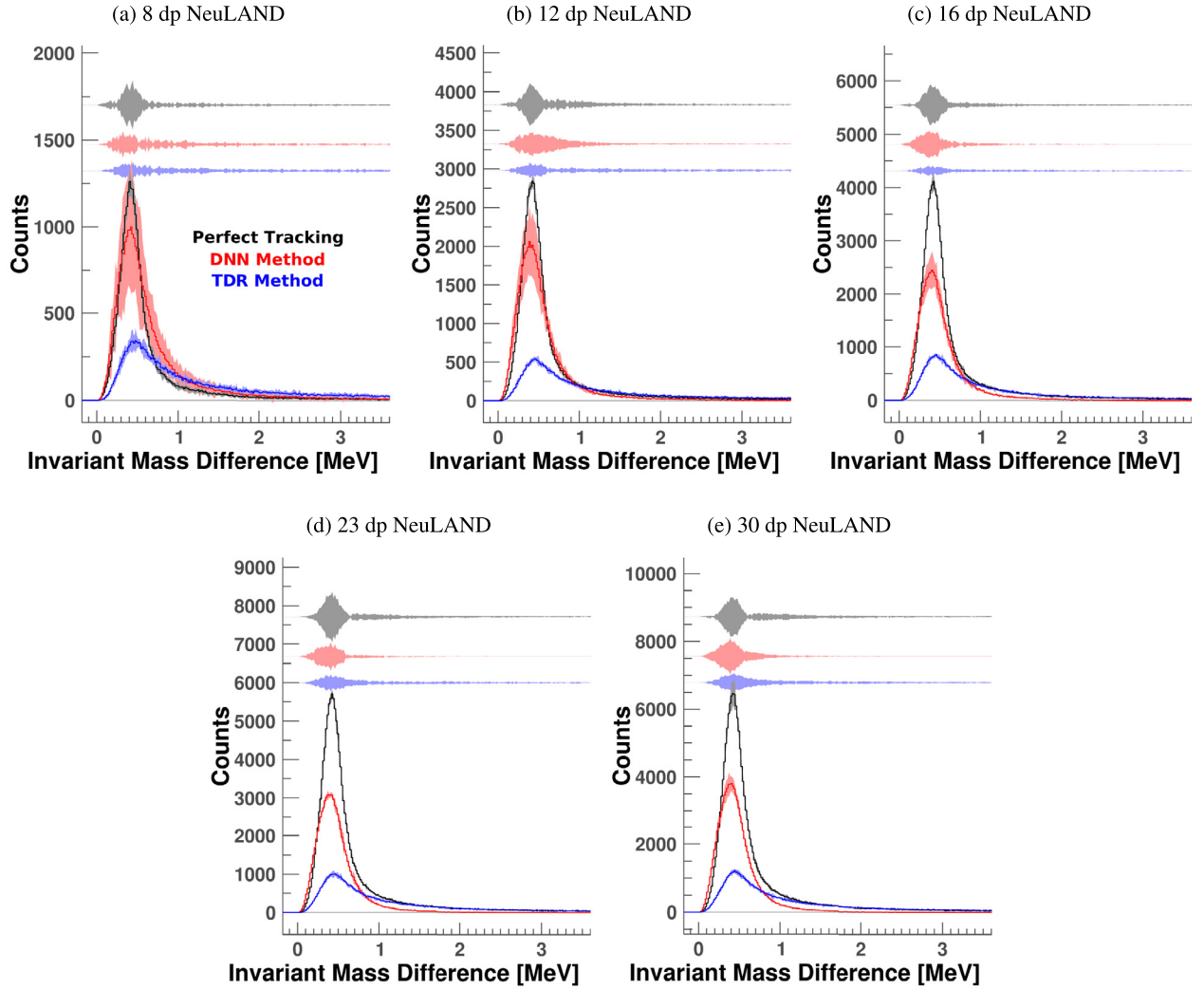


Fig. 10. Same as Fig. 9, but for a neutron energy of 600 MeV.

two-thirds of the correctly-classified events in a $4n$ experiment that requires $\Delta\theta_{\max} = 0.2^\circ$ are false positives, while this would only be one-tenth for $4n$ experiments that have no restrictions on $\Delta\theta_{\max}$ ($\Delta\theta_{\max} \rightarrow \infty$ in Fig. 12).

From Fig. 12, we also observe that the restricted DNN result drops with smaller $\Delta\theta_{\max}$ along with the “Perfect Tracking” result. However, since these results are also always (significantly) below the “Perfect Tracking” results, one could argue that there is still room for improving the DNN algorithm which would, in particular, be important for experiments that require a small value of $\Delta\theta_{\max}$. However, based on our experience in developing the present DNN algorithm, we do not expect much further improvement. Since the relative differences between the DNN results and the “Perfect Tracking” results are about the same at the multiplicity determination stage (Section 3) as they are at the “Hit” selection stage (Section 4), we conclude that improvement should start to take place at the multiplicity determination stage. However, for this network the best possible options were already selected (for network properties like the loss function, optimizer, learning rate, etc.). Moreover, all available NeuLAND data were already given as inputs to the network (see Section 2.2). Hence, the only further improvement that could be made is to increase the network complexity (to raise the number and/or the width of hidden layers). This has been attempted in this work and the network training failed to converge.

We believe that the main cause of the remaining differences between the “restricted DNN” and the “Perfect Tracking” algorithm is

that the particle showers in NeuLAND contain too many random components to allow for a significantly better multiplicity determination than what was achieved with the present DNN algorithm. This conclusion is further supported by the observation that the present DNN multiplicity classification is independent of $\Delta\theta_{\max}$. In Fig. 12, the “DNN Total” result is more-or-less constant with $\Delta\theta_{\max}$ and the ratio of the “DNN restricted” result to the “Perfect Tracking” result is also more-or-less constant. The randomness of “Hits” in the detector apparently deteriorates the information needed for a high-precision multiplicity classification.

A possible strategy for further improvement could be to also use information from other detectors in the setup than NeuLAND alone. Using the other detectors and the missing mass method, it is possible to obtain a 4-momentum vector containing the sum of all neutron tracks produced at the target. This 4-momentum vector carries information about the number of ejected neutrons (at the target). Therefore, depending on the capabilities (in terms of resolutions) of the other detectors in the setup, the information of this missing mass 4-momentum vector may have the potential to significantly improve the multiplicity classification. This option was not considered in the present study which focuses on the information solely obtained from NeuLAND and could be a valuable addition to the present DNN algorithm in future studies.

The present DNN algorithm has to deal with a significant amount of false positives for experiments that require good precision in the neutron scattering angle (small $\Delta\theta_{\max}$). Hence, we need to explore

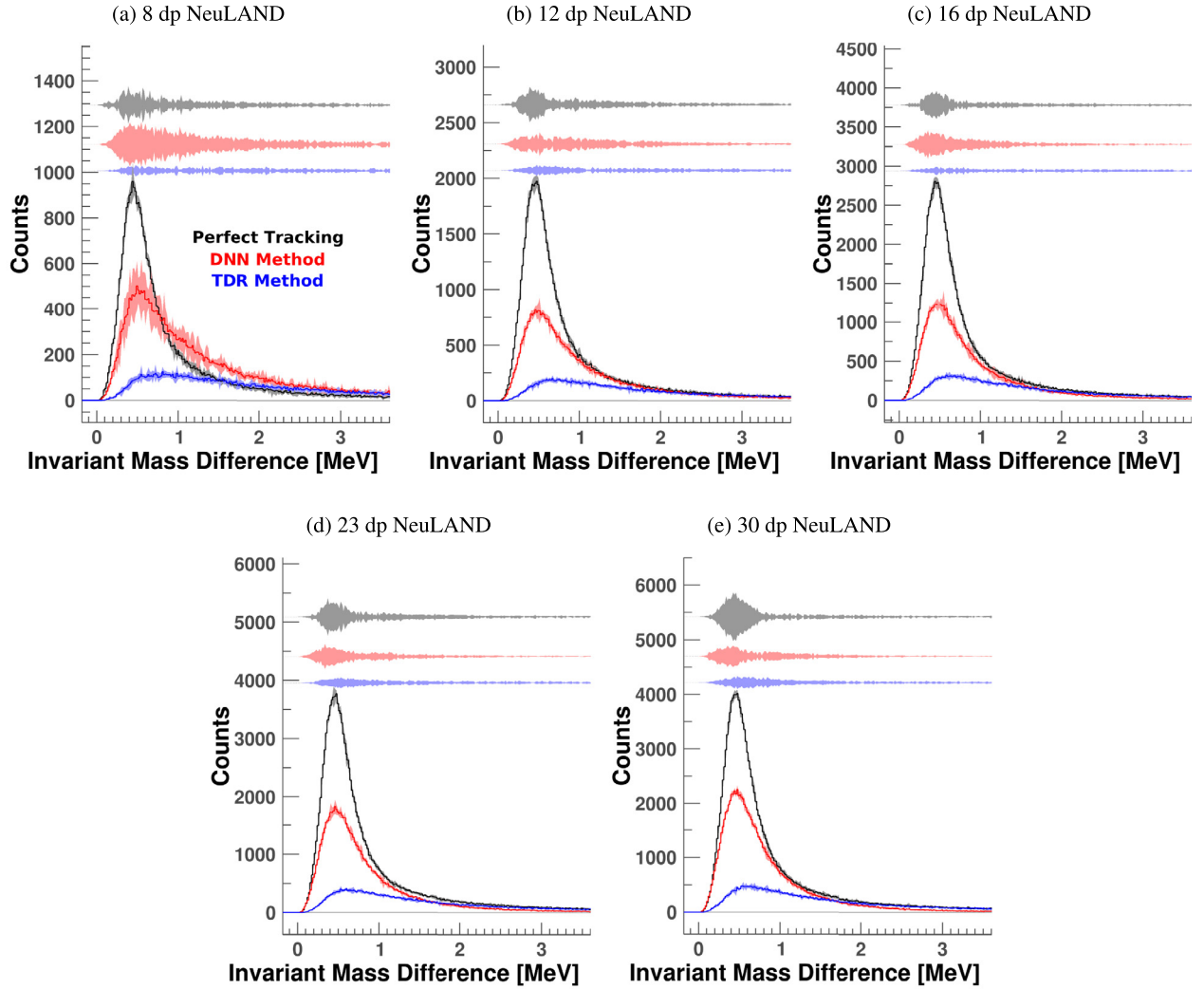


Fig. 11. Same as Fig. 9, but for a neutron energy of 1000 MeV.

the impact of these false positives on observables. Since this impact depends on the observable considered, we will limit our study to two of the most important observables in nuclear physics: invariant mass and cross section. To explore the impact of the false positives on these observables, the 4-neutron invariant mass spectra introduced in Section 4 were used. These spectra are shown in Fig. 13 for different values of $\Delta\theta_{\max}$.

It is important to realize that the number of “restricted” events shown in Fig. 12 go through another selection criterion (“Hit” selection stage) in order to obtain the results shown in Fig. 13. Hence, the number of events under the peak in Fig. 13 drops faster with $\Delta\theta_{\max}$ than the corresponding result in Fig. 12.

From Fig. 13 we conclude that the mean position and width of the peaks vary slightly with $\Delta\theta_{\max}$. Since these variations are much smaller than the experimental resolution, we conclude that the impact of the false positives on invariant mass values (such as resonance locations) is small. On the other hand, the peak area strongly depends on $\Delta\theta_{\max}$. Since during an experiment, one can only obtain the peak area corresponding to $\Delta\theta_{\max} \rightarrow \infty$, the false positives may introduce a substantial error in the peak area. Fortunately, this can be corrected for in the cross section computation. Cross section is proportional to the peak area divided by the overall detection efficiency [31]. Hence, if the false positives are included in both the peak area and the detection efficiency, the cross section can still be obtained with relatively good precision. The overall detection efficiency can be obtained from simulated data. First, one should divide the DNN peak area in the

spectrum to the “Perfect Tracking” peak area. Next, the result of this division should be multiplied with the “Perfect Tracking” multiplicity efficiency (as in Fig. 12) to obtain the overall detection efficiency. For the example shown in Figs. 12 and 13 and for multiplicity 4, this would mean an efficiency of 68% (for $\Delta\theta_{\max} \rightarrow \infty$) while this efficiency would go down to 7.8% percent for $\Delta\theta_{\max} = 0.2^\circ$. The corresponding numbers for the TDR algorithm would be 27% and 1.4%, respectively.

To summarize, it can be stated that the present DNN algorithm could be further improved. This improvement is most needed for experiments that require good precision in the neutron scattering angle (small $\Delta\theta_{\max}$) to reduce the large number of false positives. However, based on our experience, this improvement is unlikely to be realized using only NeuLAND data. On the other hand, the use of the missing mass method (from data obtained with the other detectors in the setup) may result in improvement of the present DNN multiplicity classification. The presently large number of false positives (for small $\Delta\theta_{\max}$) has little impact on invariant mass values. However, the effects of these false positives should be included in the detection efficiency for computing cross section values.

6. Conclusion

A new Machine Learning (DNN) algorithm was developed for the shower-head identification in the NeuLAND neutron detector. This DNN algorithm was compared to the TDR algorithm, a representative case

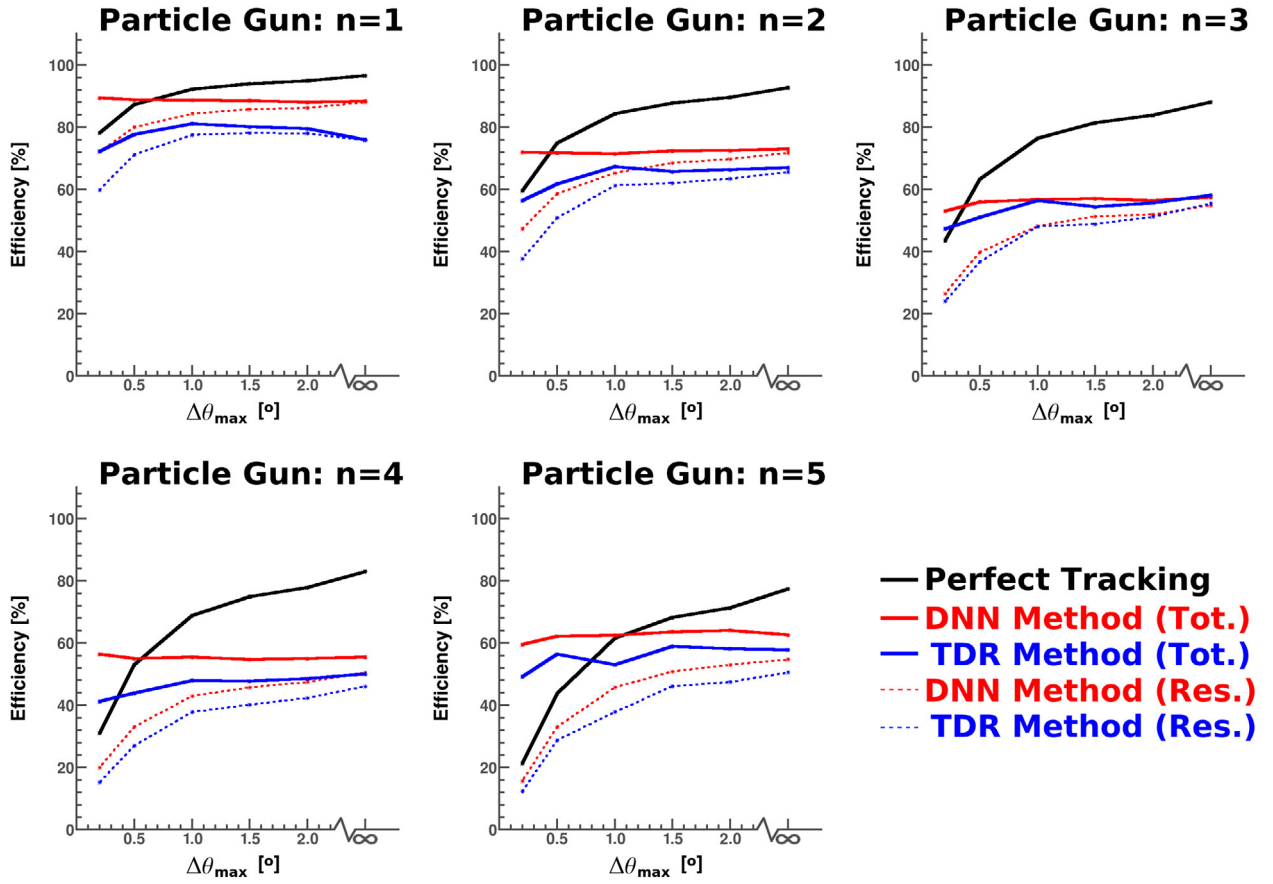


Fig. 12. Performance of the neutron multiplicity determination as a function of $\Delta\theta_{\max}$. The full results are shown as solid lines for the TDR, DNN and “Perfect Tracking” algorithms. The dashed lines correspond to the restricted scenario; the part of events where the false-positives are excluded. The simulations are performed for a neutron energy of 600 MeV, a NeuLAND configuration of 30 dp and a distance between NeuLAND and the target of 14 m. See text for further details.

(a) Comparison between the “Perfect Tracking” and DNN algorithms. (b) Comparison between the “Perfect Tracking” and TDR algorithms.

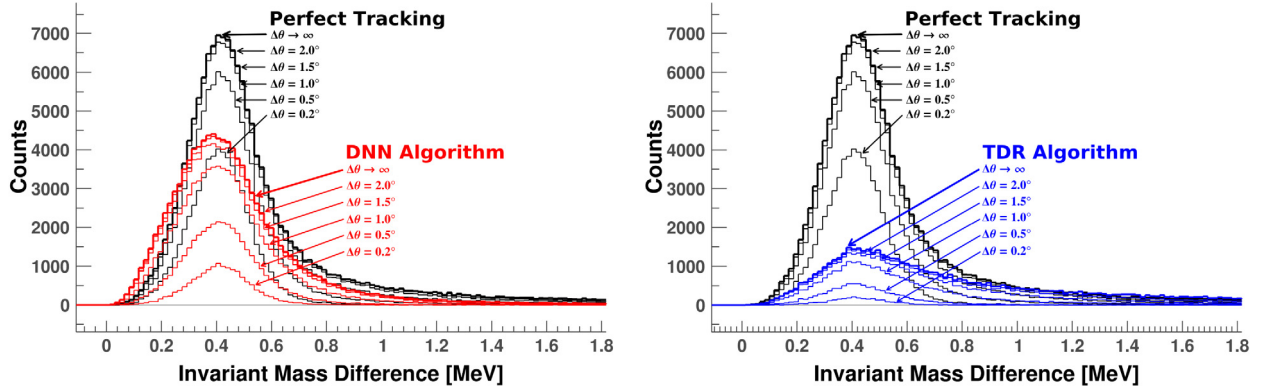


Fig. 13. Invariant Mass spectra for 4 neutrons, each carrying an energy of 600 MeV. A NeuLAND configuration of 30 dp at a distance of 14 m was used. Four times the mass of the free neutron was subtracted from the total invariant mass, which is why the figures are labeled ‘Invariant Mass Difference’. The spectra are plotted for different values of $\Delta\theta_{\max}$. To obtain the results for each curve, 200,000 $4n$ events were simulated. See text for further details.

for how shower-head identification can be done in NeuLAND without the use of Machine Learning. The main reason for developing the new DNN algorithm was to increase the efficiency of the shower-head identification so that the statistical accuracy of the measurements is increased.

In virtually all investigated scenarios (see Sections 3 and 4), the new DNN algorithm offers a better efficiency for the shower-head identification than the TDR algorithm. For this reason, we conclude

that the new DNN algorithm should be employed in the analysis of this type of detectors. The shower-head identification consists of two steps: the multiplicity determination and the “Hit” selection. For the multiplicity determination, the improvement in efficiency of the DNN algorithm with respect to the TDR algorithm is small. For the “Hit” selection, it can be very large (up to a factor 3), especially for higher neutron energies.

Physics-list uncertainties and a small number of double-planes (dp) used for NeuLAND can sometimes significantly inhibit a proper shower-head identification (see Sections 3 and 4). Therefore, we recommend to increase the number of NeuLAND dp as much as possible and to reduce the physics-list uncertainties. The latter requires both simulation-model improvements and accurate benchmarks against experimental data. As discussed in Section 2.5, event mixing could also help for this. For the required number of dp, we conclude that 8 dp suffices only for one-neutron experiments; for two-neutron experiments a minimum of 12 dp are needed, and for higher neutron multiplicities, at least 16 dp are required.

Our studies indicate that further improvements on top of the present DNN algorithm are also needed, especially for experiments that require good precision in the neutron scattering angle. However, it is unlikely that these improvements can be realized using only NeuLAND data. A possible improvement could come from using the other detectors in the setup to obtain a 4-momentum vector representing the sum of all neutron tracks utilizing the missing-mass method. This 4-momentum vector should then be fed into the multiplicity DNN to improve the classification. The degree by which this would improve the results should be further investigated.

CRedit authorship contribution statement

C.A. Douma: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing - original draft, Writing - review & editing, Visualization. **E. Hoemann:** Conceptualization, Methodology, Investigation, Writing - review & editing. **N. Kalantar-Nayestanaki:** Conceptualization, Validation, Resources, Data curation, Writing - review & editing, Visualization, Supervision, Project administration, Funding acquisition. **J. Mayer:** Conceptualization, Software, Validation, Writing - review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We would like to thank C. A. Marocico, J. Messchendorp and L. R. Zwerver for their helpful discussions about the use of Machine Learning Networks. We also would like to thank K. Boretsky, I. Gašparić, A. M. Heinz, H. T. Johansson and O. Tengblad for the helpful discussions about the results presented in this work and for their useful comments on the manuscript. J. Mayer is funded by the BMBF (05P19PKFNA) and the GSI (KZILGE1416). C. A. Douma is funded by the European Union's Horizon 2020 research and innovation programme collaboration, see Fig. 14.

Appendix. Network structure optimization

In this appendix, the optimization procedure for the DNN network structure is discussed. From general considerations (see Section 2.2), it was determined that the DNN should be a densely-connected network with ReLU activation functions on the hidden neurons and SoftMax activation functions on the output neurons. It was also determined that the DNN should have 9002 input neurons and (usually) 5 output neurons. This leaves the following parameters open: the number of hidden layers in the network, the number of neurons per hidden layer, the choice for the minimization function (the network loss function) and the choice for the minimization algorithm.

These parameters were found by considering different networks with different values for one of these parameters, thereby optimizing



Fig. 14. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 654002.

Table 2

Optimization of the DNN network structure.

Loss function	Layer 1	Layer 2	Optimizer	Accuracy
Cross entropy	9000	0	ADAM	75.5%
Hinge	9000	0	ADAM	59.9%
Mean squared error	9000	0	ADAM	73.7%
Cross entropy	3000	0	ADAM	74.6%
Cross entropy	9000	0	ADAM	75.5%
Cross entropy	9000	100	ADAM	74.1%
Cross entropy	9000	300	ADAM	74.7%
Cross entropy	9000	800	ADAM	74.2%
Cross entropy	9000	0	ADGRAD	74.3%
Cross entropy	9000	100	ADGRAD	73.3%
Cross entropy	9000	300	ADGRAD	77.5%
Cross entropy	9000	800	ADGRAD	79.7%
Cross entropy	3000	0	ADGRAD	75.4%
Cross entropy	9000	0	ADGRAD	80.0%
Cross entropy	9000	100	ADGRAD	79.5%
Cross entropy	9000	300	ADGRAD	79.1%
Cross entropy	9000	800	ADGRAD	79.7%
Cross entropy	9000	1200	ADGRAD	81.4%
Cross entropy	9000	3000	ADGRAD	81.1%
Cross entropy	9000	3500	ADGRAD	73.2%
Cross entropy	9000	6900	ADGRAD	20.0%
CNN network [21]	–	–	–	74.0%
TDR [13]	–	–	–	63.4%
Bayesian [13]	–	–	–	50.4%
Scoring [13]	–	–	–	40.8%

these parameters one-by-one. The outcome of this optimization procedure is given in Table 2. The Accuracy considered for the optimization procedure is the average efficiency for the multiplicities one to five. Hence, if one would take the average over the five subfigures in Figs. 6–8, one would obtain the accuracy in Table 2. The situation considered in Table 2, is the 30 dp design goal for NeuLAND, 600 MeV neutrons, multiplicities ranging from 1 till 5 and a distance of 14 m between NeuLAND and the target. The table explores different standard options [23] for the optimizer (minimization algorithm), loss function (minimization function) and number of neurons per hidden layer. The last section contains different algorithms from Refs. [13,21] which have been put into the table for comparison purposes.

As one can see from the first section of Table 2, the Categorical Cross entropy [27] was determined to be the optimal choice for the loss function. From the second and third section of Table 2, it is clear that ADGRAD is the better choice for the optimizer (other choices than ADAM or ADGRAD gave significantly worse results). It was also determined by optimization that for ADGRAD, learning rate 0.001, $\epsilon = 0$ and zero decay gave the best results. Finally, the fourth section of the table reveals that the optimal choice for the hidden layers is 9000 and 1200 neurons (the bold-types row). Similar Networks with more than two hidden layers failed to achieve convergence during the training.

A single DNN network training (like in Table 2) required several hours of computing time (we used an NVIDIA GeForce GTX 1050 Ti Mobile GPU for the training). On the other hand, it takes several weeks to train a single CNN network, and it could achieve an accuracy of 74% [21]. This is why we chose to use a DNN in our work rather than a CNN.

The exact values of the accuracy in Table 2 require some discussion. The values for the accuracies in Table 2 (including the CNN, TDR, Scoring and Bayesian algorithms) were obtained using the same “Perfect Tracking” algorithm as what was used in Refs. [8,13,21]. However, this algorithm turned out to incorrectly handle the Geant4 TrackIDs [32], causing the accuracies in Table 2 to come out too high. This is an overall effect in Table 2, which means that the conclusions about the optimal network parameters remain valid.

In this work, we corrected for this problem in the “Perfect Tracking” algorithm, causing the bold-typed accuracy from Table 2 to drop from 81.4% to 65.5% (which is consistent with the red curves in Fig. 7). Likewise, the TDR efficiency in Table 2 drops from 63.4% to 59.5% (which is consistent with the blue curves in Fig. 7). All results reported in this work (except in the Appendix) were based on the correct “Perfect Tracking” algorithm. However, for Table 2, we have deliberately chosen to display it using the previous “Perfect Tracking” algorithm to maintain backward compatibility with Refs. [8,13,21].

References

- [1] The R³B collaboration, Technical Proposal for the Design, Construction, Commissioning and Operation of R³B: A Universal Setup for Kinematical Complete Measurements of Reactions with Relativistic Radioactive Beams, Tech. rep., GSI Helmholtzzentrum für Schwerionenforschung, 2005.
- [2] NuSTAR Collaboration Super-FRS working group, H. Geissel, et al., Technical Design Report on the Super-FRS, Tech. rep., GSI and Collaborators, 2009, URL <https://www.yumpu.com/en/document/view/9225949/technical-design-report-super-fragment-separator>.
- [3] FAIR Baseline Technical Report, Vol. 2, Tech. rep., GSI, 2006, URL [https://www.nipne.ro/international/cooperations/fair/Accelerator_\(and\)_Scientific_Infrastructure.pdf](https://www.nipne.ro/international/cooperations/fair/Accelerator_(and)_Scientific_Infrastructure.pdf).
- [4] M. Borri, et al., Detector production for the R³B Si-tracker, Nucl. Instrum. Methods A 836 (2016) 105–112, URL <https://www.sciencedirect.com/science/article/pii/S0168900216308683>.
- [5] D. Cortina-Gil, et al., CALIFA, a dedicated calorimeter for the R³B/FAIR, Nucl. Data Sheets 120 (2014) 99–101, <http://dx.doi.org/10.1016/j.nds.2014.07.017>, URL <http://www.sciencedirect.com/science/article/pii/S0090375214004694>.
- [6] B. Gastineau, et al., Design status of the R³B-GLAD magnet: Large acceptance superconducting dipole with active shielding, graded coils, large forces and indirect cooling by thermosiphon, Trans. App. Supercond. 18 (2) (2008) 407–410, URL <https://ieeexplore.ieee.org/document/4495518>.
- [7] The R³B collaboration, Technical Report for the Design, Construction and Commissioning of the Tracking Detectors for R³B, Tech. rep., GSI Helmholtzzentrum für Schwerionenforschung, 2014, URL https://edms.cern.ch/ui/file/1865815/2/TDR_R3B_TrackingDetectors_public.pdf.
- [8] The R³B collaboration, Technical Report for the Design, Construction and Commissioning of NeuLAND: The High-Resolution Neutron Time-of-Flight Spectrometer for R³B, Tech. rep., GSI and Collaborators, 2011, URL https://edms.cern.ch/ui/file/1865739/2/TDR_R3B_NeuLAND_public.pdf.
- [9] C.A. Douma, et al., Design studies for the NeuLAND VETO detector, J. Phys. Conf. Series 1024 (1) (2018) 012027, URL <http://stacks.iop.org/1742-6596/1024/i=1/a=012027>.
- [10] C.A. Douma, et al., Feasibility study for the use of a VETO wall for the NeuLAND neutron detector, Nucl. Instr. Methods A 930 (2019) 203–209, URL <https://www.sciencedirect.com/science/article/pii/S016890021930395X>.
- [11] Rexon, Components & TLD Systems, Inc. RP-408 Scintillators. URL <http://www.rexon.com/RP408.htm>.
- [12] Saint-Gobain Crystals, BC408 Scintillators. URL <https://www.crystals.saint-gobain.com/products/bc-408-bc-412-bc-416>.
- [13] J. Mayer, Charting NeuLAND: Towards Multi-Neutron Reconstruction with the New Large Area Neutron Detector and the Virtual γ -ray Spectrometer (Ph.D. thesis), Universität zu Köln, 2018, URL <https://kups.ub.uni-koeln.de/24865/>.
- [14] Y. LeCun, et al., Deep learning, Nature 521 (2015) 436–444, <http://dx.doi.org/10.1038/nature14539>, URL <https://doi.org/10.1038/nature14539>.
- [15] C.A. Douma, Measurement of the Gamow-Teller States in ¹¹⁶Sb and ¹²²Sb (Ph.D. thesis), University of Groningen, The Netherlands, 2019, URL https://www.rug.nl/research/portal/files/74454235/Complete_thesis.pdf.
- [16] D. Bertini, R3BRoot, simulation and analysis framework for the R³B experiment at FAIR, J. Phys. Conf. Series 331 (3) (2011) 032036, URL <https://iopscience.iop.org/article/10.1088/1742-6596/331/3/032036/pdf>.
- [17] D. Kresan, R3Broot. URL <http://r3broot.gsi.de/>.
- [18] A. Revel, et al., Extending the southern shore of the Island of inversion to ²⁸F, Phys. Rev. Lett. 124 (2020) 152502, <http://dx.doi.org/10.1103/PhysRevLett.124.152502>, URL <https://link.aps.org/doi/10.1103/PhysRevLett.124.152502>.
- [19] The Geant4 collaboration, Geant4 Reference Physics List. URL http://geant4-userdoc.web.cern.ch/geant4-userdoc/UsersGuides/PhysicsListGuide/html/reference_PL/QGSP_BERT.html.
- [20] The ENSAR collaboration and its partners, INCL Geant4 physics list. URL <http://irfu.cea.fr/dphn/Spallation/physlist.html>.
- [21] M. Polleryd, Convolved Events: Neutron Reconstruction using Neural Networks (Ph.D. thesis), Chalmers University of Technology, 2017, URL <https://odr.chalmers.se/bitstream/20.500.12380/253132/1/253132.pdf>.
- [22] A. Martins, et al., From softmax to sparsemax: A sparse model of attention and multi-label classification, Proc. Mach. Learn. Res. 48 (2016) 1614–1623, URL <http://proceedings.mlr.press/v48/martins16.html>.
- [23] F. Chollet, et al., Keras, GitHub Repository. URL <https://keras.io/>.
- [24] M. Abadi, et al., TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems, ArXiv <http://arxiv.org/abs/1603.04467>.
- [25] J. Schmidhuber, Deep learning in neural networks: An overview, Neur. Netw. 61 (2015) 85–117, <http://dx.doi.org/10.1016/j.neunet.2014.09.003>, URL <http://www.sciencedirect.com/science/article/pii/S0893608014002135>.
- [26] J. Duchi, et al., Adaptive subgradient methods for online learning and stochastic optimization, J. Mach. Learn. Res. 12 (2011) 2121–2159, URL <http://www.jmlr.org/papers/volume12/duchi11a/duchi11a.pdf>.
- [27] Z. Szabó, et al., Cross-entropy optimization for independent process analysis, ICA 2006: Ind. Comp. A. and Blind Signal Sep. (2006) 909–916, http://dx.doi.org/10.1007/11679363_113.
- [28] C.A. Douma, DNN module for R³BRoot. URL https://github.com/R3BRootGroup/NeuLAND_DNN.git.
- [29] K. Boretzky, et al., NeuLAND - from Double-Planes to the Demonstrator, 2015-1 of GSI Report, GSI Helmholtzzentrum für Schwerionenforschung, Darmstadt, 2015, <http://dx.doi.org/10.15120/GR-2015-1-MU-NUSTAR-NR-12>, URL <https://repository.gsi.de/record/183944/files/MU-NUSTAR-NR-12.pdf>.
- [30] Z.H. Yang, et al., Study of Multi-neutron Systems with SAMURAI Spectrometer, Recent Progress in Few-Body Physics 238. URL.
- [31] C.A. Douma, et al., Gamow-Teller strength distributions of ¹¹⁶Sb and ¹²²Sb using the (³He, γ) charge-exchange reaction, Eur. Phys. J. A 56 (2) (2020) 51, <http://dx.doi.org/10.1140/epja/s10050-020-00044-9>, URL https://epja.epj.org/articles/epja/abs/2020/02/10050_2020_Article_44/10050_2020_Article_44.html.
- [32] The Geant4 collaboration, Geant4 User's Guide for Application Developers, Tech. rep., CERN, 2008, URL http://lmu.web.psi.ch/docu/manuals/software_manuals/Geant4/Geant4_BookForAppliDev.pdf.