Using machine learning for the detection of missing fuel pins in spent nuclear fuel assemblies based on measurements of the gradient of the neutron flux

Moad Aldbissi[1,2], Paolo Vinai[1], Riccardo Rossa[2], Alessandro Borella[2] and Imre Pázsit[1],

[1]Division of Subatomic, High Energy and Plasma Physics, Chalmers University of Technology, Göteborg, Sweden
[2]Belgian Nuclear Research Centre, SCK CEN, Mol, Belgium

Abstract

One of the main tasks in nuclear safeguards is the inspection of spent nuclear fuel (SNF) assemblies to detect possible diversions of special nuclear material such as ^{235}U and ^{239}Pu. In the inspection, measurements of relevant observable quantities are acquired from the assembly, e.g., neutrons emitted by the spent fuel, and used to verify whether they are consistent with the declared configuration of the assembly or not. The procedure requires a physical model that can estimate the response of the detectors for a given arrangement of fuel pins in the assembly, and an unfolding technique, based on the physical model, that can be applied to retrieve, from the detector responses, the parameters of the system configuration. In this work, the use of neutron flux gradient measurements for the identification and characterisation of diversions in a SNF assembly is investigated. The unfolding procedure relies on an artificial neural network (ANN), which has the advantage of generalizing in an efficient manner the mapping of the input (in this case, the measurements from the SNF assembly) to the output (i.e., the fuel pins that are intact or replaced with dummy pins in the assembly). The training and testing of the ANN makes use of a dataset generated using Monte Carlo simulations of a typical 17x17 PWR assembly with different patterns of missing fuel pins. The dataset is built of unique scenarios so that the ANN can be tested and assessed over scenarios that are not part of the learning phase. The study shows that information related to the neutron flux gradient can lead the ANN to be more accurate in identifying the replaced fuel pins. Although the developed ANN models cannot fully reconstruct any of the diversion patterns included in the dataset, they provide results close to the real assembly configurations in most cases.

KEYWORDS: spent nuclear fuel, neutron flux gradient, machine learning, Monte-Carlo, nuclear safeguards

1 Introduction

One of the most important tasks in nuclear safeguards is regular inspections to verify that no special nuclear material is missing from the Spent Nuclear Fuel (SNF) assemblies. In the safeguards community, such a task is known as detection of partial defects [1, 2]. Spent nuclear fuel is particularly sensitive from a safeguards perspective because of its residual fissile material such as ^{235}U and ^{239}Pu. In the recent years, about 80% of the material placed under safeguards was plutonium contained in SNF [3].

Several methods of Non-Destructive Assay (NDA) such as the Digital Cherenkov Viewing Device (DCVD) [4] and the Fork Detector (FD) [5] among others are used to detect possible diversions in SNF assemblies. These techniques are approved for inspection by the International Atomic Energy Agency (IAEA) and have been extensively applied for many years [6]. The processing and the interpretation of the measurements performed with these techniques mainly relies on the expert judgement of the inspectors. In addition, the investigations are focused on the coarse detection of possible illicit diversion of nuclear material.

In order to enhance safeguards inspection of spent fuel, research efforts have been focused on the development of both detection equipment and methods for the processing of the measurements. In the field of measurement devices for nuclear engineering, recent progress has been made in the construction of miniaturized neutron detectors that combine neutron scintillators and light guiding fibers [7, 8, 9]. These detectors are attractive to nuclear safeguards because they can be introduced inside a SNF assembly to obtain information about the neutron flux and its gradient at different locations without the need to move the assembly from its storage position. For the processing of the measurements to determine whether spent nuclear fuel has been replaced or not, machine learning has a great potential since it enables an efficient algorithmic approach that can generalize the relationship from sets of measurements to system configurations and extract a high level of detail from the data, reducing the possibility of undetected diversions [10, 11].

The current work investigates the use of measurements of neutron flux gradient (alone or together with measurements of the neutron flux) as input to a machine learning algorithm based on an artificial neural network (ANN) for the detection and characterisation of SNF assemblies with partial defects. The training and testing of the ANN relies on a dataset generated using Monte Carlo simulations of a typical 17x17 PWR assembly with different patterns of replaced fuel pins. The dataset is built of unique scenarios so that the ANN can be tested and assessed over scenarios that are not part of the learning phase of the algorithm.

The methodology is introduced in section 2. The performance of ANN models trained with data of neutron flux and/or its gradient and used to identify diversion patterns in SNF assemblies is discussed in section 3. Conclusions are drawn in section 4.

2 Methodology

The general strategy for the verification of the integrity of Spent Nuclear Fuel (SNF) assemblies is to acquire measurements of relevant observable quantities, such as neutrons emitted from the spent fuel, and determine whether the outcome of the measurements is consistent with the declared configuration of the assemblies or not. This procedure requires a physical model that can reproduce the response of the detectors for a given arrangement of fuel pins in the assembly. Then, an unfolding technique based on the physical model can be applied to retrieve, from the detector responses, the parameters of the system configuration. An underlying assumption is that there is a one-to-one correspondence between the spatial distribution of the observables and the actual composition of the fuel assembly, whether intact or not, which is the basis of the identification of the defects.

For the unfolding task, an artificial neural network is studied in this work to identify the configuration of a SNF assembly and characterise, if any, the diversion, from measurements of the neutron flux gradient alone or combined with measurements of neutron flux. The artificial neural network is trained to "learn" the relationship between the sets of measurements and the relative system configurations. The training and testing of the algorithm is performed with synthetic data generated via Monte Carlo simulations both for the intact system and for cases represented by specific patterns of replaced fuel pins.

2.1 Dataset

The dataset for the training and testing of the ANN algorithm relies on Monte Carlo simulations of a 17x17 PWR fuel assembly, see Figure 1 (a). The assembly consists of 264 fuel pins with Zircaloy cladding and an initial enrichment of 3.5 w%. The assembly also contains 25 empty guide tubes where detectors can be placed.

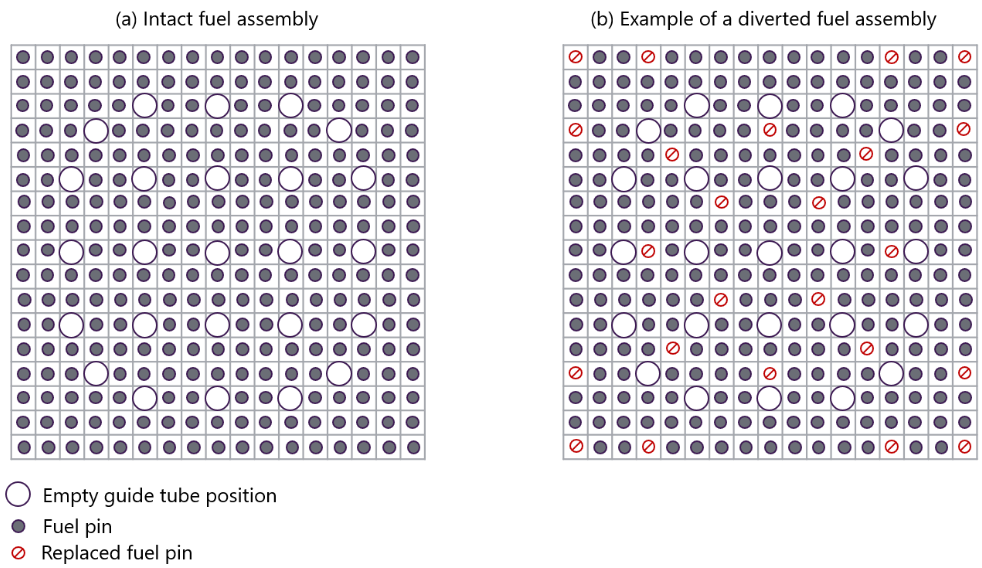

Figure 1: Intact 17x17 PWR spent nuclear fuel assembly (a) and an example of a diversion scenario (b).

In the current application, the detection of possible fuel pins missing from their positions within the assembly is a two-dimensional problem. Accordingly, a 2-dimensional model of the fuel assembly is developed for two-step simulations using the Monte Carlo code Serpent [12]. The first step is a burn-up simulation of the declared fresh fuel assembly, which consists of an irradiation cycle that continues until a final burn-up value of 40 MWd/kgU is achieved, followed by a decay cycle that replicates a cooling time of 5 years in the spent nuclear fuel pool. The second step is a fixed-source simulation that is performed with the neutron sources obtained from the burn-up simulation and distributed consistently with the diversion patterns of interest, in order to estimate the thermal neutron flux and its gradient in the guide tubes of the assembly (where neutron detectors may be inserted).

The dataset contains data of the fuel assembly without defects and 107 different diversion patterns, which can be symmetrical or asymmetrical, and have a minimum of 4 up to a maximum of 180 fuel pins replaced by dummy pins made of stainless steel, see example in Figure 1 (b).

Since the arrangement of a 17x17 PWR fuel assembly has 25 guide tubes the calculated system responses for each configuration are 75, i.e., 25 values of thermal neutron flux, 25 values of the magnitude of the gradient (absolute value), and 25 values of the angle of the gradient vector (direction). The components of the gradient vector in a guide tube are derived from the neutron flux estimated in 4 points inside the guide tube.

2.2 Artificial Neural Network

The ANN algorithm is built using the Tensorflow [13] and the Keras [14] open-source software libraries, and is based on a neural network with an input layer, a hidden layer and an output layer. The neurons that belong to the input and hidden layers are activated with the Rectified Linear Unit (ReLU) function, which allows for back-propagation with an efficient convergence rate. The weights and the learning rate of the network are optimized with the Adaptive Moment Estimation (ADAM).

The values given by the output nodes on each run lie between 0 and 1, which is due to using a Sigmoid activation function in the output layer. The Sigmoid function is a typical choice for outputs that are non-mutually exclusive, where each pin is treated independently and can be either present or replaced. The ANN is trained such that the output of a node corresponding to a missing pin should be 1, and that of a node corresponding to a pin which is present should be zero. In practice, even for a fully trained ANN, the output values will lie between 0 and 1 which are then interpreted as the probability that the pin is missing. If the probability of a fuel pin to be identified is between 0.5 and 1, the fuel pin is labelled as missing (1), If the probability is less than 0.5, the fuel pin is labelled as present (0). Then, the problem can be considered a multi-label binary classification, and the performance of the ANN model can be evaluated with the Binary Cross-Entropy loss function.

The number of neurons in the input layer of the models is fixed to the number of input features used for training, and the the number of neurons in the output layer is equal to the number of outputs, i.e., one for each of the 264 fuel pins in the assembly. A grid search optimization is performed to determine the number of neurons for the hidden layer and the number of epochs and the batch size in the training process.

3 Results

The ANN is trained using: the magnitude and direction of the thermal neutron flux gradient (G_m+G_d), only the magnitude of the gradient (G_m), only the direction of the gradient (G_d), only the thermal neutron flux (N), the combination between the neutron flux and two components of its gradient (N+G_m+G_d), the neutron flux and the magnitude of its gradient (N+G_m), and the neutron flux and the direction of its gradient (N+G_d).

The training and testing is performed via a 6-fold cross-validation process. Accordingly, the whole dataset is shuffled and divided into 6 random batches. Five of these batches are used to train the ANN, while the remaining one is used for the testing. The procedure is repeated 6 times so that each of the 6 batches serves as testing dataset one time, and then the results from the 6 tests are aggregated. The cross-validation approach is advantageous when handling small-sized datasets since all the fuel assemblies will have been used to test the model at the end of the training process. This results in a less biased assessment of the performance of the model which can occur if the model was tested only on one randomly selected test batch.

The dataset contains 108 fuel assemblies, each of them with a unique configuration. Therefore, the model is always tested over scenarios that are not seen in the training phase. This aspect allows assessing the ability of the model to generalize its predictions with respect to unknown data, which is important since it is not feasible to create a training dataset with all the diversion patterns (their number would be overwhelming for the limitation of the computational resources).

After the 6-fold cross-validation is completed, the algorithm is scored based on the number of fuel pins that it has identified correctly in all the fuel assemblies available from the dataset. As summarized in Figure 2, the fuel pins are scrutinized according to 4 categories (which define a so-called confusion matrix). The 'True Negatives' are all the correctly predicted intact fuel pins, and the 'True Positives' are all the correctly predicted missing fuel pins. On the other hand, the 'False Positives' are the intact fuel pins that are wrongly predicted as missing and the 'False negatives' are the missing fuel pins wrongly predicted as intact.

		Real label	
		Intact (0)	**Missing (1)**
Predicted label	**Intact (0)**	**True negative**	**False negative**
	Missing (1)	**False positive**	**True positive**

Figure 2: Definition of the categories for the scrutiny of the predicted fuel pins.

The performance of the ANN models is quantified with 4 metrics, i.e., the pin-accuracy, the precision, the recall and the F1 score. The pin-accuracy corresponds to the percentage of the correctly predicted fuel pins (the sum of the true positives and true negatives) out of the total number of fuel pins, considering all the fuel assemblies in the dataset. The precision is defined as the fraction of correctly predicted missing pins (the true positives) over all the pins predicted as missing (the sum of true and false positives). The recall is equal to the fraction of correctly predicted missing pins (true positives) over the total number of missing pins in the dataset (equivalent to the sum of true positives and false negatives). The F1 score is the harmonic mean of the precision and recall values.

Table 1 shows the comparison between the ANN models obtained from the training with different sets of simulated measurements. The model that uses the neutron flux and the magnitude of its gradient (N+G_m) has the best performance in all four metrics. The model that uses only the direction of the gradient vector (G_d) has the lowest performance in terms of pin-accuracy and precision. The model for only the thermal neutrons (N) has the lowest performance in terms of recall.

The value of the precision is directly connected to the number of false positives and hence it reflects the ability of the model to correctly predict intact fuel pins in the assembly. The use of both the thermal neutron flux and the magnitude of its gradient (either separate or combined) result in better precision values, i.e., better identification of the intact fuel pins. The direction of the gradient vector as an input feature always has a negative effect on the precision of the ANN model.

The recall value depends on the number of false negatives and thus is an indication of the ability of the model to correctly predict replaced fuel pins in the assembly. The models that use the gradient (either in magnitude, direction or both) have greater recall values and hence can better detect replaced fuel pins, while the model based on the thermal neutron flux has the lowest recall value.

Table 1: Performance metrics with respect to the input features used to train and test the ANN model.

Metric	Gradient Detector Responses						
	N + G_m	N + G_m + G_d	G_m + G_d	G_m	N	N + G_d	G_d
Pin-accuracy	0.82	0.81	0.80	0.80	0.79	0.77	0.76
Precision	0.66	0.63	0.62	0.64	0.64	0.56	0.55
Recall	0.60	0.59	0.59	0.52	0.43	0.44	0.44
F1	0.63	0.61	0.60	0.57	0.51	0.49	0.49

None of the models can fully reproduce any of the diversions. This is expected because the size of the dataset is relatively small and the training cases are different from the testing cases. However, the majority of the model predictions are close to the real diversion patterns. As an example, Figures 3 and 4 show two configurations with partial defects and their reconstruction via the models that use the neutron flux and the magnitude of its gradient (N+G_m), the neutron flux along with the magnitude and direction of its gradient (N+G_m+G_d), and the neutron flux (N), respectively.

Figure 3 shows an example where the results of the models reflect the global trends reported in Table 1. The N+G_m model provides the closest prediction to the real pattern as indicated by the values of the evaluation metrics for the specific case. The N model and the N+G_m+G_d model both have values of precision, recall and F1 score equal to zero because they do not identify any of the missing pins correctly (i.e., no true positives). In addition, the N model gives less false positives in comparison to the N+G_m+G_d model, which is consistent with the general finding that the N model tends to have slightly higher precision value than the N+G_m+G_d model.

A case that does not follow the global trend is also included, see Figure 4. The N model provides the best reconstruction of the diversion in terms of all the evaluation metrics. Such result might be related to different factors, e.g., the specific characteristics of the diversion pattern combined with the knowledge learned by the algorithm from similar scenarios in the training process. Further studies are needed to clarify these deviations in the behavior of the models.

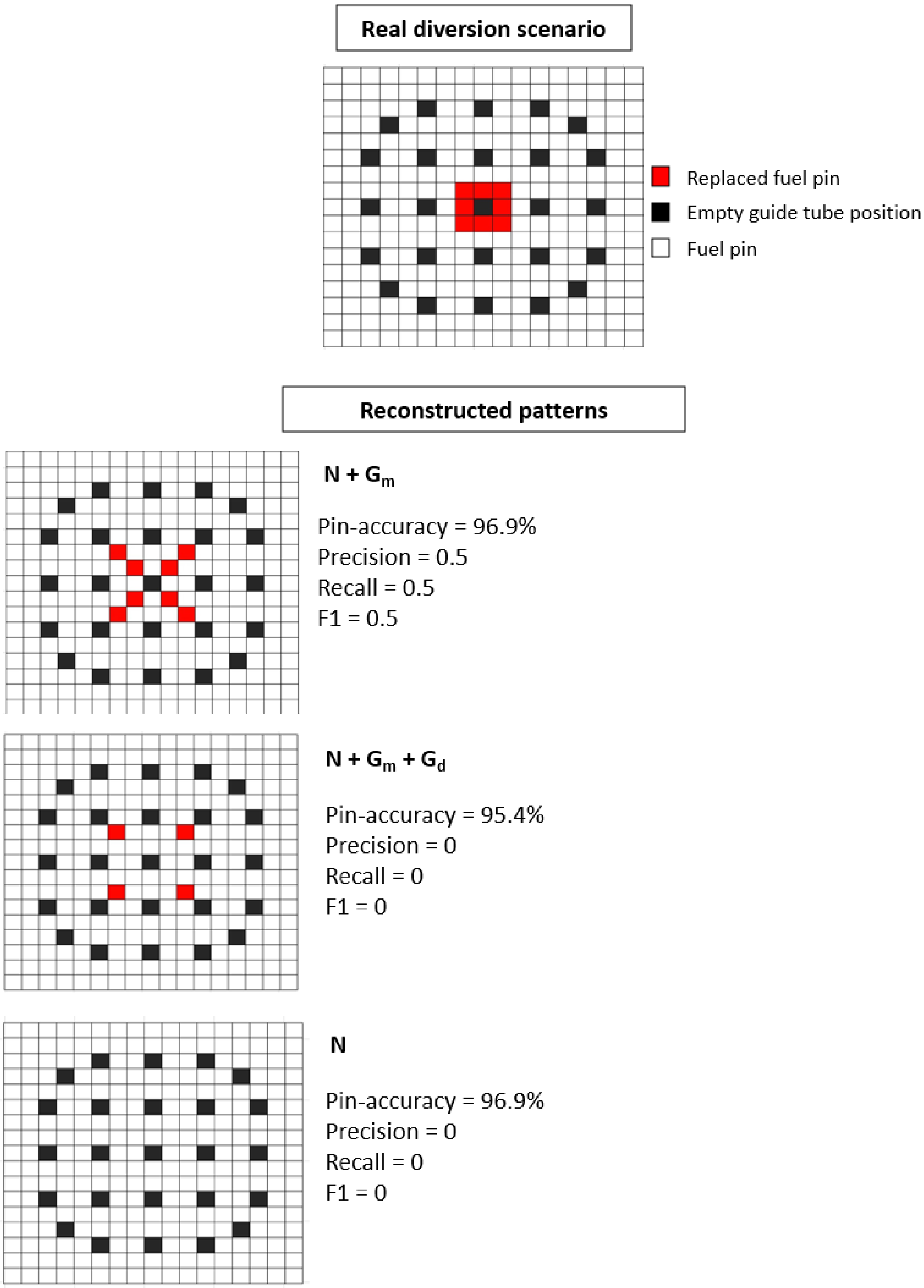

Figure 3: Example of a diversion scenario for which the results of the models are consistent with the global trends of Table 1.

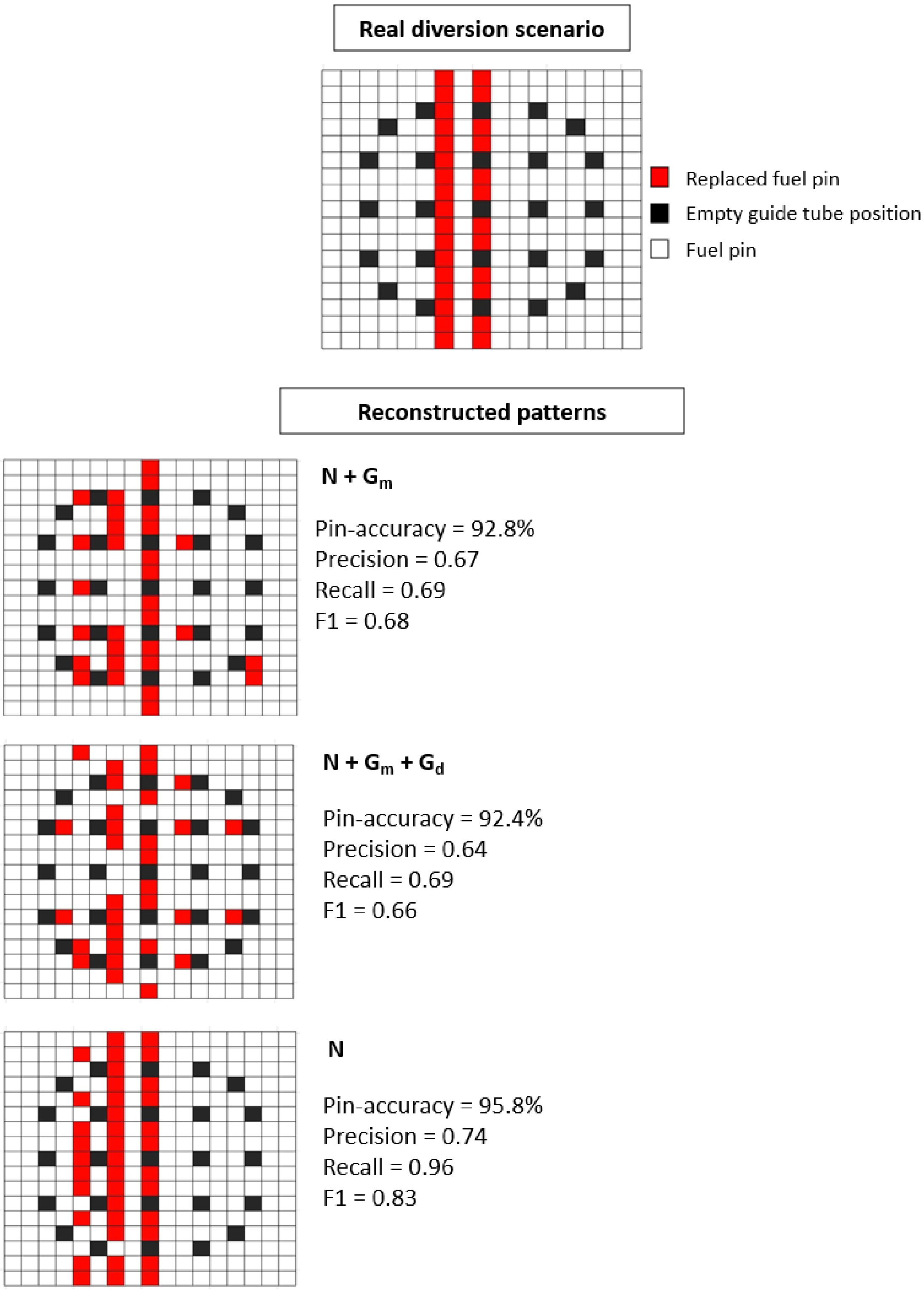

Figure 4: Example of a diversion scenario for which the results of the models deviate from the global trends of Table 1.

4 Conclusions

An artificial neural network has been investigated for the detection and characterisation of partial defects in a typical 17x17 PWR SNF assembly from measurements of neutron flux gradient alone or together with measurements of neutron flux. The ANN has been trained and tested on a dataset generated with Monte-Carlo simulations of the assembly with different diversion patterns. The simulated neutron flux and/or its gradient have been taken at the empty guide tube positions and served as input features to the ANN. As output, the probability of each fuel pin in the assembly to be replaced is estimated.

Different ANN models have been developed, depending on the input features, i.e., the neutron flux (N), the magnitude of the gradient (G_m), the direction of the gradient vector (G_d), and any possible combination of them. The performance of each model was quantified in terms of the pin-accuracy, the precision, the recall and the F1 score. The ANN model based on the neutron flux and the magnitude of its gradient ($N+G_m$) has been found to have the best performance in all four metrics, indicating that information from the neutron flux gradient within the fuel assembly can be beneficial to the ANN predictions.

An additional aspect of the study is that the ANN models have been tested and evaluated over "unknown" diversion scenarios, i.e., not included in the training phase. Although these ANN models cannot fully reconstruct any of the diversions available in the dataset, they do provide results close to the real assembly configurations in most cases. Then, the ANN models can generalize to some extent the mapping from measurements to patterns of replaced fuel pins, despite the limited size of the current dataset.

Further investigations are planned to gain insights into how the knowledge learned in the training affects the ANN predictions of "unknown" cases, which can be useful for the future expansion and optimization of the dataset. The analysis also shows the need to clarify the reasons for the negative effect of the direction of the gradient vector on the precision of an ANN model.

References

[1] IAEA, "Safeguards Glossary 2022 Edition," 2022.

[2] H. Lee and M. S. Yim, "Investigation of a fast partial defect detection method for safeguarding PWR spent fuel assemblies," *Annals of Nuclear Energy*, vol. 144, p. 107496, 2020.

[3] IAEA, "Annual Report 2013, GC(58)/3, Vienna, Austria," 2014.

[4] E. Branger, S. Grape, and P. Jansson, "Partial defect detection using the DCVD and a segmented region-of-interest," *Journal of Instrumentation*, july 2020.

[5] P. M. Rinard and G. E. Bosler, "Safeguarding LWR spent fuel with the FORK detector." Los Alamos National Laboratory report LA-11096-MS, 1988.

[6] IAEA, "Safeguards Techniques and Equipment, International Nuclear Verification Series, No. 1 (Rev. 2), Vienna, Austria," 2011.

[7] T. Yagi, H. Unesaki, T. Misawa, C. H. Pyeon, S. Shiroya, T. Matsumoto, and H. Harano, "Development of a small scintillation detector with an optical fiber for fast neutrons," *Applied Radiation and Isotopes*, vol. 69, no. 2, pp. 539–544, 2011.

[8] K. Watanabe, M. Yamanaka, T. Endo, and C. H. Pyeon, "Real-time subcriticality monitoring system based on a highly sensitive optical fiber detector in an accelerator-driven system at the Kyoto University Critical Assembly," *Journal of Nuclear Science and Technology*, vol. 57, no. 2, pp. 136–144, 2020.

[9] F. Vitullo, V. Lamirand, J.-B. Mosset, P. Frajtag, O. Pakari, G. Perret, and A. Pautz, "A mm^3 fiber-coupled scintillator for in-core thermal neutron detection in crocus," *IEEE Transactions on Nuclear Science*, vol. 67, no. 4, pp. 625–635, 2020.

[10] R. Rossa, A. Borella, and N. Giani, "Comparison of machine learning models for the detection of partial defects in spent nuclear fuel," *Annals of Nuclear Energy*, vol. 147, p. 107680, 2020.

[11] R. Rossa, A. Borella, and K. van der Meer, "Comparison of the SINRD and PDET detectors for the detection of fuel pins diversion in PWR fuel assemblies," in *INMM-59 Annual Meeting*, 1 October 2018.

[12] J. Leppänen, "The Serpent Monte Carlo code: Status, development and applications in 2013," *Annals of Nuclear Energy*, vol. 82, pp. 142–150, 12015.

[13] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015.

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