UNIVERSITY OF BIRMINGHAM

Fusion Neutron **Acti**vation Spectra **U**nfolding by **N**eural **N**etworks
(FACTIUNN)

author:	Ocean Wong
	(Hoi Yeung Wong)
supervisor:	Ross Worrall
Submitted in fulfilment of the requirement for:	MSc. Physics and Technology of Nuclear Reactors
date:	June-September 2019
student ID:	1625143

I warrant that the content of this dissertation is the direct result of my own work and that any use made in it of published or unpublished materials is fully and correctly referenced.

Abstract

Include these things:

- Attempted this
- Got this result
- advice for the future

Keywords: activation, neutronics, fusion

Contents

List of Figures

1 Introduction

In a fusion reactor, the neutron fluence can go up to as high as 1.6×10^{21}[citation needed]. (For JET, [citation needed]; for ITER, [citation needed]). This leads to an unprecedented need of shielding against neutrons of up to 14.1 MeV or higher energies, which has not been experienced in fission reactors before [citation needed].

Neutrons are notoriously difficult to shield against due to their uncharged nature, and therefore low propensity to interact with matter[citation needed]. To develop effective shielding for various components of the reactor from these high energy neutrons, the energy spectrum of the neutrons created inside the nuclear reactor has to be well understood[citation needed]. It is also important to understand the neutron spectrum inside the reactor in order to develop Tritium breeding modules, which is essential for making fusion a sustainable source of clean energy. [citation needed] An accurate measurement of the neutron spectrum is required to properly model the energy distribution of neutrons to be used in neutron transport simulations for the above purposes.

Therefore, neutron energy measurement is a key focus [rewording needed] in the diagnostic systems in all fusion reactors.

Ironically, for the same reason that they are difficult to shield against, neutron energy is also difficult to measure. Neutrons, especially high energy neutrons such as the 14.1 MeV neutrons created in fusion reactors, do not easily deposit their full energy into a sufficiently small detection volume to allow direct measurement[citation needed]. Various neutron detectors has been developed to deal with this problem; [citation needed] however, most of them cannot stand this high neutron fluence that is found at the first wall of fusion reactors without additional shielding that changes the flux profile, defeating the objective of trying to measure neutrons energy distribution with minimal disturbance to the spectrum itself.[citation needed] The extreme temperature and magnetic fields inside the nuclear fusion reactor compounds the difficulty of employing other means of neutron measurement as most electronics will not be able to function in such environments effectively.

This is where the technique of neutron activation stands out:

By analyzing the level of activations in various elements induced by neutrons, relying on the fact that different reactions has different sizes of reaction cross-sections, each with varying sensitivities to neutrons of different energies, one can infer the neutron spectra that was previously present at the first wall.

This is a very robust method as it does not require any active components, thus can be employed for very high neutron fluxes and fluences [citation needed], as the total number of neutron activation reactions can be controlled by changing the thickness of the activation foils used [citation needed] according to the anticipated neutron fluence in the next campaign, so not to paralyze the γ radiation detector.

The disadvantage of this method is that it has to be time-integrated (over the whole campaign), i.e. no information about the temporal variation in neutron flux can be extracted. [citation needed]

Another disadvantage of using neutron activation as the means of measuring the neutron spectrum in a fusion reactor is that it is an indirect method of measurement, requiring the measured reaction rates to be 'unfolded' back into reaction rates. This is a 'mathematically incorrectly posed ' problem[2], as will be further explained in the next section (2.3), requiring an *a priori* spectrum to be provided before the unfolding procedure can take place. This is because the number of activities recorded (usually denoted as M) is fewer than the number of neutron groups (usually denoted as N) of whose activity we would like to know, i.e. M<N, thus the problem is underdetermined (the number of contraints is fewer than the number of variables). The *a priori* has to be used in order to introduce extra information into the problem. However, if this *a priori* spectrum deviates too much from the actual spectrum, then the result of the unfolding will be inaccurate.

To address this problem, an investigation into using neural networks for the purpose

of unfolding is presented in this thesis. Neural networks excels in incorperating previous spectra as *a priori* information, without requiring users to explicitly input an *a priori*. Two approaches are proposed. The first one is to use neural networks directly as an unfolding tool; and the second one is to use them as an *a priori* generator, which is then fed into an existing unfolding code, where the actual neutrons spectra is then calculated out of.

2 Theory

When a nuclide is placed in the activation module at the irradiation position inside a nuclear fusion reactor (or any other neutron sources), it is activated via one or more nuclear reactions with the incoming neutrons. The probability of interacting with the incoming neutron via reaction j is proportional to the microscopic cross-section $\sigma_j(E)$, where E is the neutron's energy, and reaction j is a neutron-induced reaction, i.e. (n,??) reaction.

By measuring the activity of reaction j's daughter nuclide in the activation foil (which has a known amount of the initial nuclide) after irradiation, and multiplying it by a correction factor of

$$\frac{1}{1 - exp(\lambda_j T)} \tag{1}$$

the reaction rate Z_{0j} can be obtained. This correction factor accounts for the decay of the daughter nuclide of reaction j which has a half-life of λ_j, over the period T which is the duration between irradiation and measurement. A more complicated correction factor is required if the irradiation period is comparable to the half-life λ_j, or if the population of the parent nuclides for reaction j changes over the course of the irradiation. This can be done using FISPACT-II, detailed in [citation needed].

The total reaction rate of the j^{th} reaction can then be expressed as a Fredholm integral as follows:

$$Z_{0j} = \int_0^\infty R_j(E)\phi_0(E)dE \tag{2}$$

where the reaction rate Z_{0j} has the unit of s^{-1}, ϕ_0 is the neutron flux (unit: $cm^{-2}s^{-1}eV^{-1}$), which is a function of energy E. The unfolding process aims to find a solution spectrum ϕ which approximates the actual spectrum ϕ_0 as closely as possible.

As for R in the equation above, (which has dimension of area)

$$R_j(E) = \sigma_j(E)\frac{N_A}{A}F_j\rho V \tag{3}$$

assuming that there is no self-shielding/down-scattering inside the foil. N_A is the Advogadro's constant (unit: mol^{-1}), A is the molar mass of the parent nuclide for reaction j (unit: g mol^{-1}), F_j is reaction j's parent isotope's mass fraction in the foil's constituent material (unit: dimensionless), ρ is the density of the alloy (unit: g $barn^{-1}$ cm^{-1}), V is the volume of the foil (unit: cm^3) Note that $\sigma(E)$ (unit: $barn$)is the only energy dependent component in R.

The neutron spectrum can be discretized into N energy bins:

$$Z_{0j} = \sum_{i=1}^{N} R_{ij}\phi_{0i} \tag{4}$$

where ϕ_{0i} is the scalar flux integrated over the energy bin's range

$$\phi_{0i} = \int_{E_{i-1}}^{E_i} \phi_0 d(E) \tag{5}$$

, thus having a unit of $cm^{-2}s^{-1}$.

By assuming that the scalar flux distribution inside each energy bin is relatively flat, equation 4 calculates Z_{0j} by replacing $(R_j(E), E_{i-1} \leq E \leq E_i)$ with

$$R_{ij} = R_j(E_{i-1}) \tag{6}$$

Let there be M neutron-induced reactions whose reaction rate was measured,

$$\begin{aligned} \forall j \in & \{1, ..., M\}, \\ \exists Z_{0j} \in & \mathbb{R}_{\geq 0} \end{aligned} \tag{7}$$

Collecting all reaction rates into a vector $\boldsymbol{Z_0}$ of M-dimensions, one can express eq. 4 as a matrix multiplication equation:

$$\boldsymbol{Z_0} = \underline{\underline{\mathbf{R}}} \boldsymbol{\phi_0} \tag{8}$$

where $\underline{\underline{\mathbf{R}}}$ is a $M \times N$ matrix, termed the *response matrix*. $\boldsymbol{\phi_0}$ is an N-dimensional vector containing the neutron flux in the each of the N bins. The subscripts 0's denotes that they are the measured/known quantity, as opposed to the conjectured solutions which will appear later in this text.

For nuclear fusion applications, the number of possible reaction investigated M is very limited, as the parent nuclide of each of these reactions must exist in solids which:

- does not melt in the reactor,
- can be machined into specified shape and thickness,
- are safe to be handled,
- has sifficiently stable parent and daughter isotopes for the activation and γ measurement to be carried out respectively.

in practice, fewer than 10 types of metals/alloys are used in these systems [citation needed]. For ACT, the system that analyses the largest number of activation reactions thus far [citation needed], M is still limited to 11.

Meanwhile, the number of bins, N, can be arbitrarily high; for some investigations, such at the one in [4] it goes up to 709 bins. This makes the unfolding problem a very underdetermined one.

In the mathematical sense of the problem, an inverse does not exist. This is because, theoretically, multiple neutron spectra, say $\boldsymbol{\phi_0}$, $\boldsymbol{\phi_1}$ and $\boldsymbol{\phi_2}$, can give the same set of reaction rates $\boldsymbol{Z_0}$, so there is no correct, unique choice of mapping of $\boldsymbol{Z_0}$ back to $\boldsymbol{\phi_0}$, $\boldsymbol{\phi_1}$ and $\boldsymbol{\phi_2}$. [2] A conceivable situation is detailed below as an example:

The three spectra $\boldsymbol{\phi_0}$, $\boldsymbol{\phi_1}$, $\boldsymbol{\phi_2}$ has identical flux values in all but the first two energy bins:

- $\boldsymbol{\phi_0}$ has a flux of $0cm^{-1}s^{-1}$ in the 1^{st} bin and a flux of $2 \times 10^{10}cm^{-2}s^{-1}$ in the 2^{nd} bin;
- $\boldsymbol{\phi_1}$ has a flux of $2 \times 10^{10}cm^{-2}s^{-1}$ in the 1^{st} bin and a flux of $1 \times 10^{10}cm^{-2}s^{-1}$ in the 2^{nd} bin;
- $\boldsymbol{\phi_2}$ has a flux of $4 \times 10^{10}cm^{-2}s^{-1}$ in the 1^{st} bin and a flux of $0cm^{-2}s^{-1}$ in the 2^{nd} bin;

And the reaction cross-sections in this energy range (very low neutron energy/thermal energy) for all but the 1^{st} reaction is vanishingly small, as all other reactions than the 1^{st} reaction are threshold reactions.

If the first two columns of the response matrix $\underline{\underline{\mathbf{R}}}$ are given as follows:

$$R_{1,j} = \delta_{1j}(5 \times 10^{-11})cm^2 \tag{9}$$

$$R_{2,j} = \delta_{1j}(1 \times 10^{-10})cm^2 \tag{10}$$

where the δ used is the Kronecker delta,

then one can see that $\boldsymbol{\phi_0}$, $\boldsymbol{\phi_1}$ and $\boldsymbol{\phi_2}$ will all lead to the same reaction rate Z_0. This is because, in each of these cases, the first two bins of each of the $\boldsymbol{\phi}$ contributes the same amount of reaction rate of 2 counts s^{-1} to the 1^{st} reaction rate (Z_{0j} where j=1), ultimately resulting in the same measured reaction rate of $\boldsymbol{Z_0}$.

Such a problem is termed 'mathematically incorrectly posed'. [2]

2.1 General unfolding methods

The most straight-forward way of getting back a solution ϕ is by using the Moore-Penrose inverse matrix. This matrix inversion operation generalizes the usual matrix inversion operation for square matrices, where the $M \times N$ response matrix $\underline{\underline{\mathbf{R}}}$ in equation 8 is inverted into an $N \times M$ matrix $\underline{\underline{\mathbf{R}}}^{-1}$, so that ϕ can be obtained by $\boldsymbol{\phi} = \underline{\underline{\mathbf{R}}}^{-1}\boldsymbol{Z_0}$. However, this method is the equivalent of rotating a 2-D photo of a 3-D object from a horizontal position to an upright/tilted position: the solution is still 'trapped' in a flat, M-dimensional surface within the N-dimensional solution space.

Therefore to start the unfolding process, extra information has to be given to the program. This is termed the *a priori* spectrum [citation needed].

The most general unfolding program can, ideally, find a solution $\boldsymbol{Z}$, $\underline{\underline{\mathbf{R}}}$ and $\boldsymbol{\phi}$ [3], such that their overall deviation from the measured reaction rates ($\boldsymbol{Z_0}$), expected response matrix ($\underline{\underline{\mathbf{R_0}}}$), and the initial guessed neutron spectrum ($\boldsymbol{\phi_0}$), is minimized.

However, this then requires a solution search in a very large number of dimensions, namely $(M \times N) + M + N$ dimensions. To make the problem more approachable, we can reduce the number of dimensions by $M \times N$ by assuming that the response matrix $\underline{\underline{\mathbf{R_0}}}$ is accurately and precisely defined, fixing the response matrix during the solution search (which is a χ^2 minimization process).

2.2 Current practice

Some programs, such as GRAVEL and SAND-II, simply start their iterative solution search from this *a priori* spectrum, with the aim of minimizing the χ^2 (which measures the deviation of $\boldsymbol{Z}$ from $\boldsymbol{Z_0}$); while others, such as MAXED, add the deviation of the solution spectrum from the *a priori* spectrum ($\boldsymbol{\phi}$ from $\boldsymbol{\phi_0}$) **on top of** the deviation of the solution reaction rates from the measured reaction rates ($\boldsymbol{Z}$ from $\boldsymbol{Z_0}$) when evaluating the χ^2.

1. introduction to neural networks: what it does and how they're used as a blackbox that guesses the function instead of the answer.
2. how using a neural network may solve it: because we're using more information, by learning the patterns among previous neutron spectra patterns.
3. briefly mention that training NN, however, does comes with its own set of problems: overfitting, etc.

- General theory includes minimizations in M+N+(MxN) dimensions: write down the minimization equation, stating:
- when we assume no covariance, then S_i quantity i is diagonal.

current practice ignores: ... and ...

- MAXED
- GRAVEL
- others ...

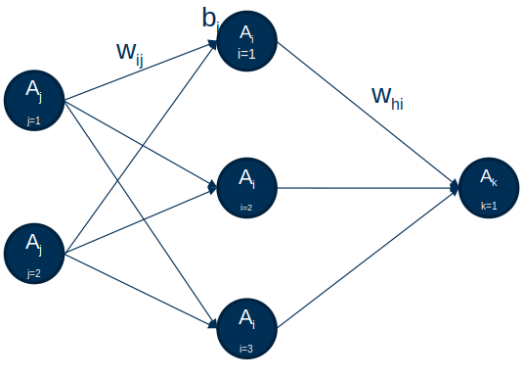


Figure 1: Illustration of the structure of a typical neural network

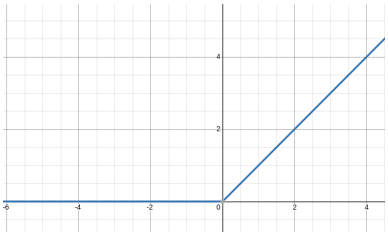


Figure 2: A ReLU function (a rectifying function)
Function input=x-axis; function output=y-axis.

A ReLU function (Figure 2) is typically used as the non-linear activation function σ, as it is one of the simplest and therefore fastest non-linear activation function to compute.

For N neutron groups folded by an MxN matrix into the reaction rates of M isotopes, all of the above algorithms are trying to find a solution (coordinate of a point) in M+N dimensional space, which is constrained on an (M+N)-M=N dimensional surface (as required by the M activities obtained through folding the spectrum through the response matrix), while deviating the least from both the N dimensional a priori (guess neutron spectrum provided by the user) and the M reaction rates (measured isotopic activities) simultaneously.

JET's current practice: Uses MCNP to create a response matrix for each of the 175 groups, ignoring the variance information given to create the matrix, and then...

2.3 Neural Networks

Neuralnetworks, on the other hand, minimizes the chi square during the training phase by changing the parameters in the function itself, and therefore uses MORE information available, to solve the problem of underdetermination.

Figure **??** Is a crude representation of how a 1-hidden-layer neural network can approximate a cubic function. A single hidden layer neural network with one scalar input and one scalar output is able to approximate any non-linear functions, provided that there are enough input and output neurons. (Explain adjusting bias leads to offset, and weights changes the slope, etc...) (Talk more about how this is a scalar func, not a vector field, etc. And interpolation is good, extrapolation is bad.)

2.3.1 Universal approximation theory

1. general theory of neural network, and it's problems:
   - explain in detail what is the structure of a Feed forward neural network, (act func, bias, weights)
   - explain the need for normalization of features

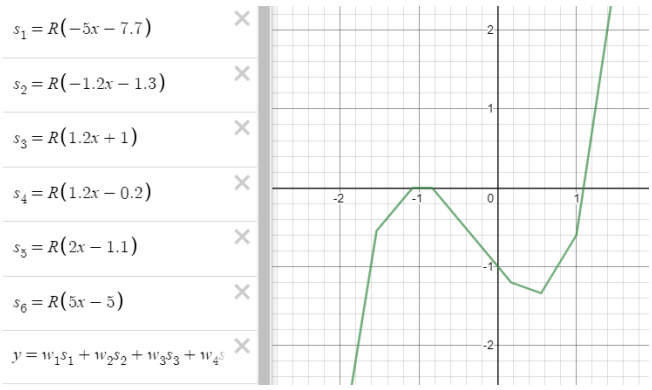


Figure 3: A cubic function approximated by a neural network
The neural network used has 1 hidden layer containing 6 neurons.
R is an alias for the ReLU function (see Figure 2), and y is the output layer neuron's activation value, obtained by summing over the product of the activation of the i^{th} neuron s_i with the weight of its connection to the final layer, w_i.

- explain what is underfitting
- explain what is overfitting (the problem with memorization) (i.e. too many free paramters, so it just sits in a local minima that is slightly off from where it should be.)
- explain why it's important to keep the number of parameters low in order to minimize the problem of getting stuck at plateaus and saddle points (because of the unlikihood of minima)
- and list the hyperparameters that are tweaked in practice in order to make it more efficient.
  (a) number of layers: generally a more complex problem requires more layers, e.g. more is required for facial recognitiion and differentiation than for a hand-written text classification engine.
  (b) activation function used
  (c) dropout (reduce connectedness of the NN)
  (d) weight regularlization
  (e) reduce learning rate on plateau
  (f) early stopping
  (g) etc... (whatever else I can think of)

2. this universal approximation theory can be applied in the M isotopes (M features) to N neutron groups (N labels), by making a k-layer neural networks, and list the hyperparameters that will/will not be tweaked:
   - give an intuitive understanding of what's going on: mapping onto an N dimensional surface in a M dimensional volume/space?
   - this works also because we expect it to find the (linear and non-linear components of) covariance between labels implicitly? Or is there more to it?
   - in other words, it'll find where the bumps and dimples are on this "surface", the orientation of this surface, etc. * (gotta think about this more)
   - ASSUMING that we have a sufficient number of nuclear reactions to account for each of these components, and that the features (reaction rates) are noiseless, then a NN will be able to perfectly replicate it.
   - otherwise, a neural network may only be able to partially replicate these spectra, with poor quality

2.3.2 The way that neural network will be applied in here:

What are labels, what are features.

By assuming that invert to equation 8 exist, i.e.

$$\boldsymbol{Z} = \underline{\underline{\mathbf{R}}}^{-1}\boldsymbol{\phi_0} \tag{11}$$

where the possible set of solution $\boldsymbol{Z}$ exist in an M-dimensional manifold in the N-dimensional space,

- approximate number of nodes per layer
- activation function to be used
- number of epochs to be used: 10000 or less, as it was observed
- number of layers:
- loss
- neural network can work WITH or WITHOUT relying on the physics...
- here's it without relying on the physics...
- here's it with relying on the physics... Usually NN has fewer labels than features. Therefore the function to turn labels back into features doesn't always exist.

  But in this case, there are more labels than features. Therefore the function (denoted as f as follows) that maps labels to feature exist (but not the other way around).

  Therefore, when using features to predict labels, we can generate, while trainig, the "would-have-been" features from the predicted labels using feature = f(predicted labels), giving us yet another metric to train the neural network on, from which we expect to yield an increased learning rate as a result.

Plus the list of hyperparameters that will be kept constant/tweaked:

- num epochs: we do EarlyStopping instead
- num nodes:
- activation: always ReLU, since it's the standard, and to keep the number of tweaked hyperparamters simple
- loss function: pairwise, tried using cosine distance but it didn't work because it simply gave NaN's.

3 Literature review

Previous attempts of machine learning techniques applied to unfolding only include:

- Bonner spheres
- square matrices
- genetic algorithms

Therefore the neural network approach to unfolding the few-channel case is entirely new.

4 Initial attempt on parameterised code

For the simple 5×5 case, did perfectly well.

But once we take log of both sides, it's become an impossible problem because we can't take log of a matrix that's singular. See Figure 4.

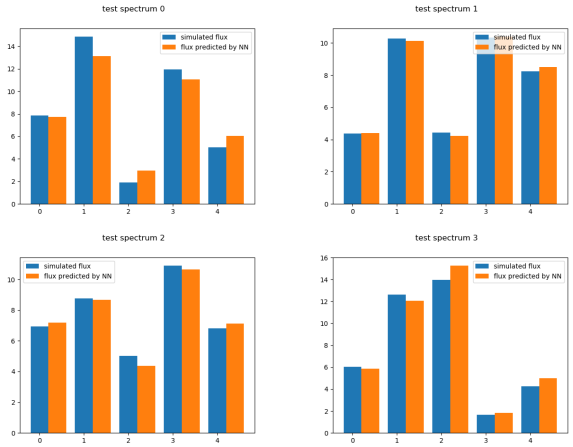
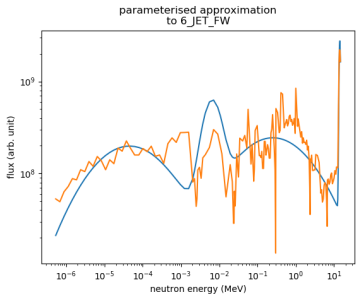
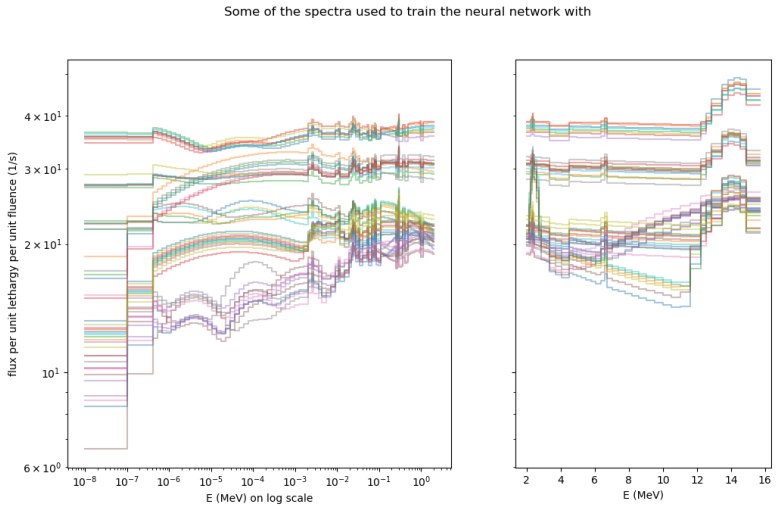


Figure 4: The spectra predicted by a 2-hidden-layers neural network (with 16 nodes per layer) on a fully determined system.



(a) An example of parametrisation performed on the JET spectra. These flux values are the total flux inside each energy bin, *not* divided by the lethargy span of each bin, so they are higher/lower in wider/narrower energy bins.



(b) 300 × perturbed spectra were generated for each of the 6 original fusion spectra and plotted here, in flux per unit lethargy.

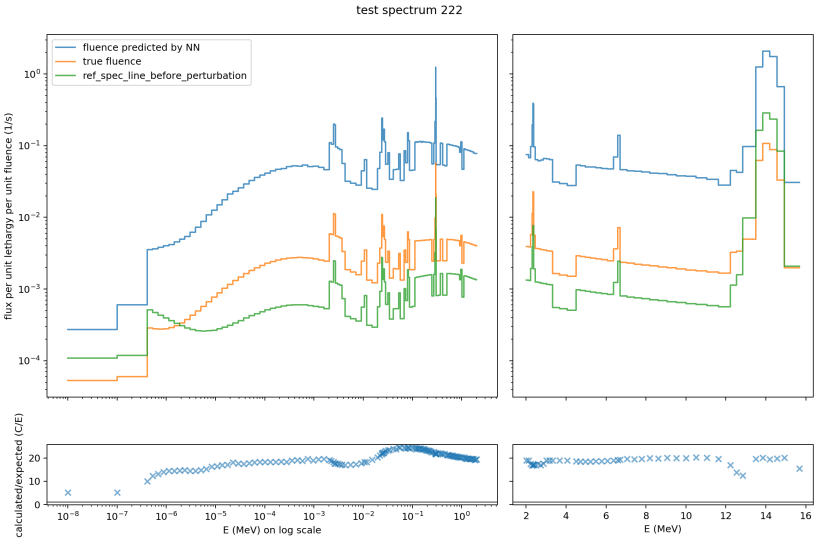
Figure 5: Data augmentation performed to create simulated spectra.

4.1 Creating simulated spectra by data augmentation

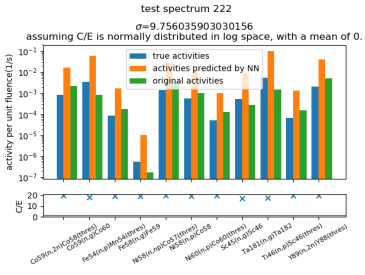
To demonstrate that the neural network is capable of doing the unfolding procedure in an underdetermined condition, 1800 spectra were made to train the neural network. Further increase in complexities (number of layers and numebr of nodes per layers) yield no significant improvement in the fit. This is because increasing the complexity further will simply lead to over-fitting, as the neural network 'memorizes' the relationship between features and labels. This was reflected by the continual decrease in the loss value over the training dataset, while the loss value over the validation dataset stops improving.

After being trained on variants of the same spectrum (only with slighly different peak heights), along with the perturbed spectra of 6 other fusion neutron sources.

It should be trying to approximate the underlying pattern, In deed it does quite well. But it's still not perfect. Though that doesn't matter too much for our purpose. We deem it as good enough; move on.



(a) The neural network's predicted fluence was not normalized unlike the other two, therefore it was scaled up by a constant factor relative to the true fluence.



(b) The activities (features) used. The neural network was given the true activities, and asked to predict the fluence (6a).
Note that the colour scheme was reversed, i.e. the blue bars denote the activities predicted by the neural network instead of the true activities, and vice versa.

Figure 6: The neural network's attempt at predicting a perturbed JET first wall spectrum.

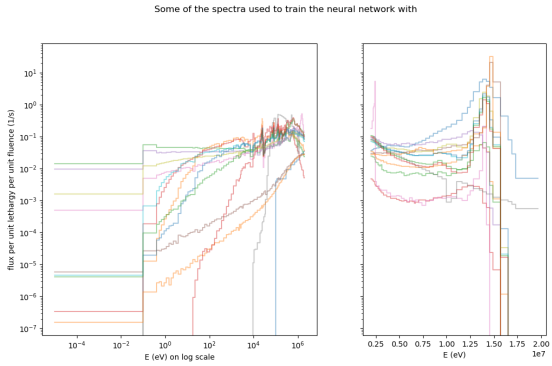


Figure 7: All fusion spectra used, obtained from [5]

5 Attempt to work on the real data

IF there's an underlying pattern, it will pick it out. It performs just as well.

5.1 Data attained

Acquired from the IAEA UKAEA compendium.

Rebinning

The neural networks were trained on 19 fusion neutron spectra, which were all rebinned into the Vitamin-J group structure. How did you sort it

By assuming that the flux per unit lethargy inside each bin are relatively flat (energy independent), the activities contributed by the neutron flux in each bin is then proportional to the product of neutron flux with the microscopic. Symbolically,

$$Z_j \propto \sum_i \sigma_{ij}\phi_i \tag{12}$$

Akin to the equation 4. Therefore these microscopic cross-section values were used in place of the response function for each of the reaction, assuming the constant of proportionality

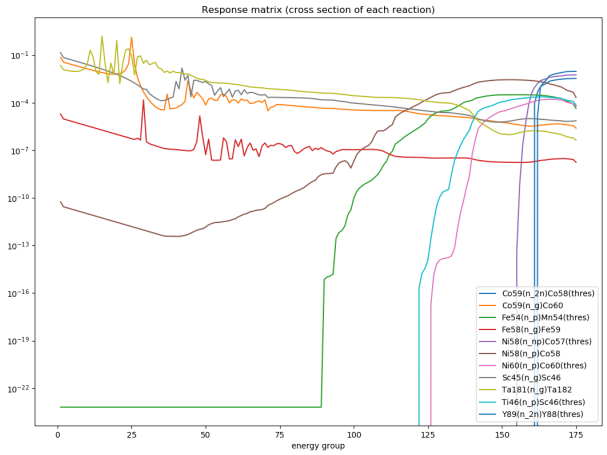


Figure 8: Microscopic cross-section of each reaction
These values are obtained from TENDL15 via FISPACT-II [1] at the left edge of each bin in the Vitamin-J group structure.

in equation 12 is unity.

The experiment was repeated using other metrics as the loss values. The complete list of tested loss values are listed below:

- mean-squared-error (rms) = $\sum_i$
- mean-pairwise-squared-error
- mean-squared-error-including-folded-reaction-rates
- mean-pairwise-squared-error-including-folded-reaction-rates

This shows that Figure 10 likely only achieves the above average performance serendipiteously by re-tracing the same average spectrum. This hypothesis is supported by it replicating a very similar spectrum when it attempts to deduce the spectra corresponding to the other two test data.

- What work has been done to find the optimal
- How to quantify optimal

5.2 potential future improvements

1. In the future we can calculate the spectral index, and plug THOSE values in, instead of plugging in the direct values, it might be better at picking out these differences, because it may be more obvious, and can pick it out even under so few data.

6 Benchmarking against existing codes

6.1 As an unfolding tool

If they would like to use it directly as an unfolding tool, then they can incorperate the whole folding process into the loss function; but this method requires:

- (optional) The response matrix to already been known → better results?

*Gotta make a fair comparison between a neural network unfolded against an a priori unfolded one.

The more exciting aspect arises from the fact that it can be used as an a priori generator code:

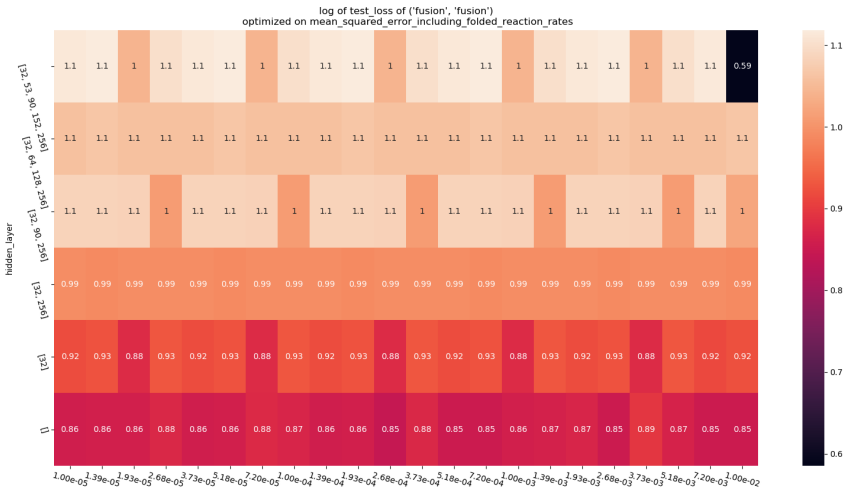


Figure 9: Heatmap visualizing the loss values of the nerual networks' prediction on the test dataset.

Each square represents a neural network with a particular set of hyperparameters, i.e. learning rate and number of layers. The learning rate increases logarithmically across the x-axis; while the number of layers increase linearly across the y-axis. The number of nodes per layer is increased logarithmically from 32 to 256 (if the number of layers ≥ 2).

Neural networks which performs better has lower loss values, and are represented with darker colours.

At the top right hand corner, i.e. the neural network with the hyperparameters of (learning rate=0.01, number of nodes in each layer=[32,53,90,152,256]), a particularly loss value is obtained. Therefore this neural network is regarded as the neural network with the optimal hyperparameter, and further investigation into the predictions of this neural network is conducted.

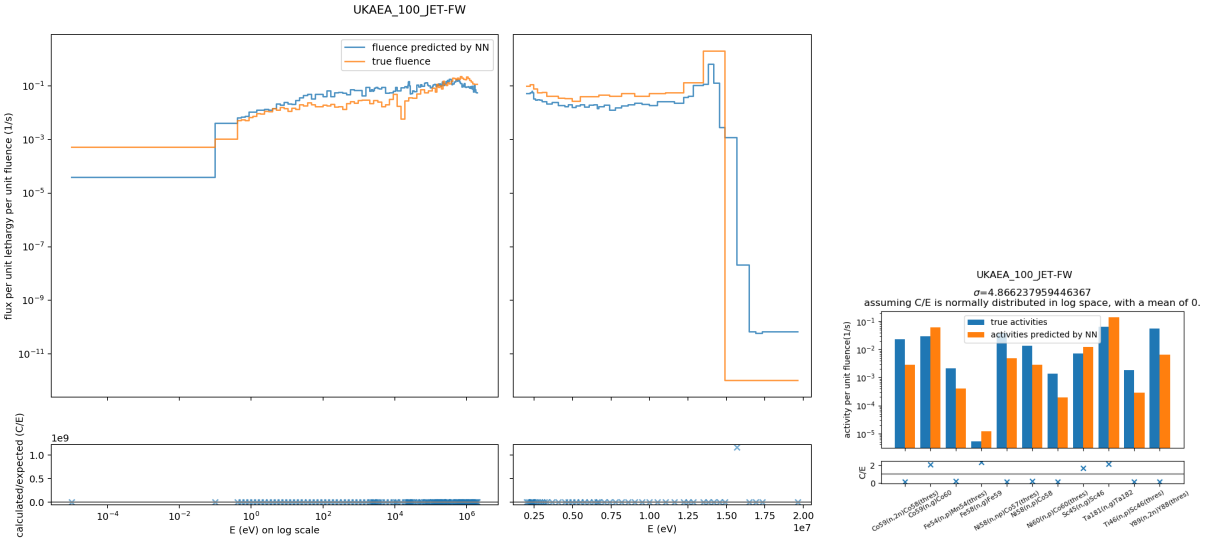


Figure 10: JET first wall spectrum as predicted by the optimally performing NN among all NN trained on fusion data.

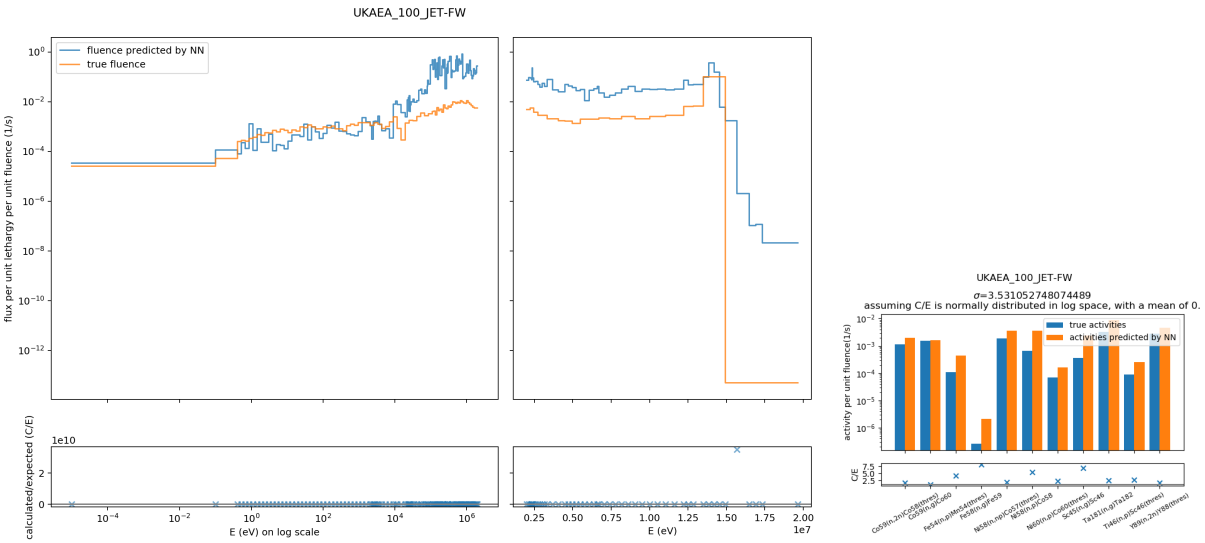


Figure 11: JET first wall spectrum as predicted by the second best performing NN among all NN trained on fusion data.

6.2 As an a priori generator

Can be used as the a priori generator?

But it also sucks as an a priori generator, giving , which is only a marginal improvement on the 10.188233 stated in Figure 13

Which is almost as bad as using a naive prior, i.e. using a flat a priori and thus giving no meaningful information to gravel before unfolding (Figure 13). In this case it achieves an average mean squared error of

- the user doesn't want to commit to hours of MCNP model generation (cite a paper where Lee Packer's group has used a whole MCNP model to get the response matrix and the reaction rates);
- and already has a few similar neutron spectra to pick from;
- want a higher reproducibility/credibility than hand-drawing an a priori with reference to the previous spectra/ averaging over the existing spectra.

EVEN if the response matrix is not known.

Allows for a probability distribution of weights? does that account for the variance and covariance between the labels and features? * But this is beyond the scope of this paper, which is to demonstrate that the idea of NN works.

7 An attempt at using fission data to predict fusion data

Explain why: we have so many fission spectra... but only very few fusion spectra

But even the best neural network trained on fission spectra and obtained the lowest loss value when tested on fusion spectra gave very poor results:

For the record, this optimal neural network has the hyperparameters of (learning rate = 0.01, number of nodes in each layer=[32,53,90,152,256]).

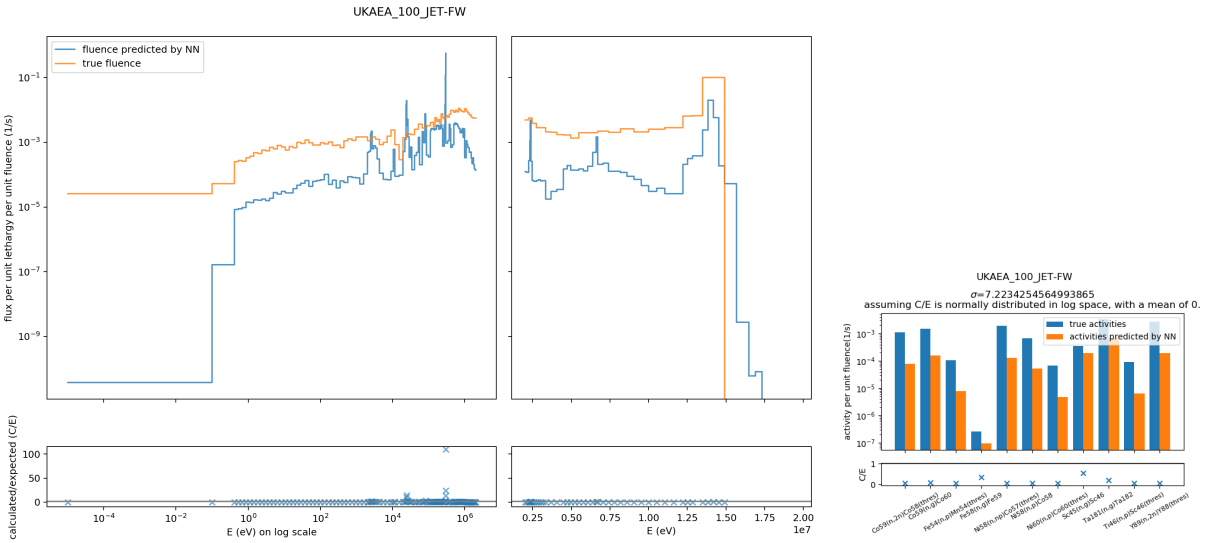


Figure 12: JET first wall spectrum as unfolded by GRAVEL upon using the NN's output as the *a priori* spectrum.

An average rms value of 9.89936 was achieved when using the neural network's prediction as the *a priori* for unfolding by gravel.

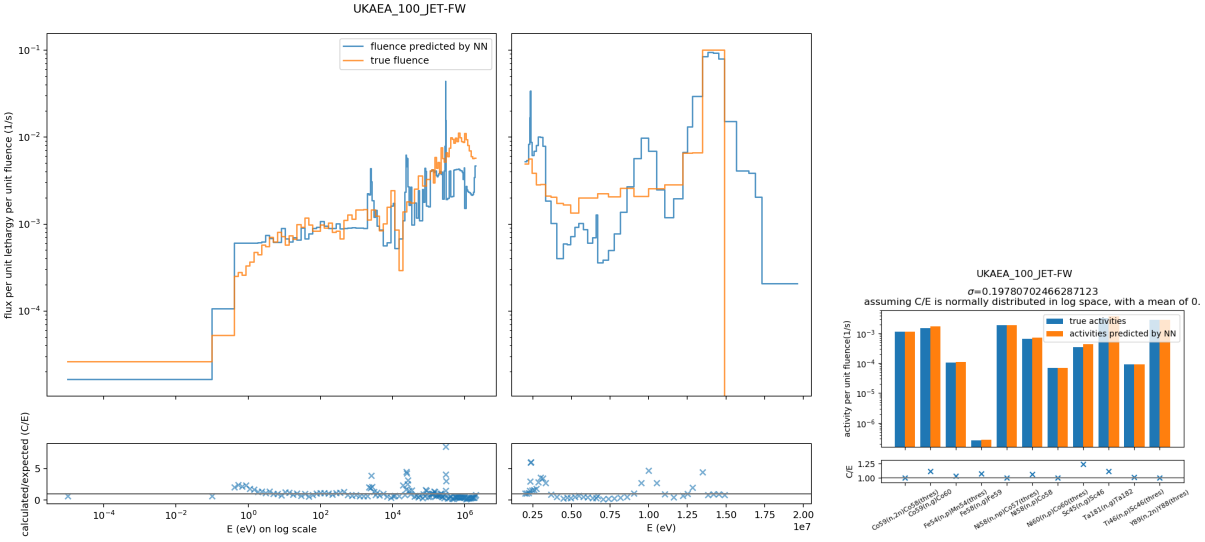


Figure 13: JET first wall spectrum as unfolded by GRAVEL upon using a flat a priori as the *a priori*

A rms value of 10.188233 when comparing the unfolded spectra against the true spectra in log space.

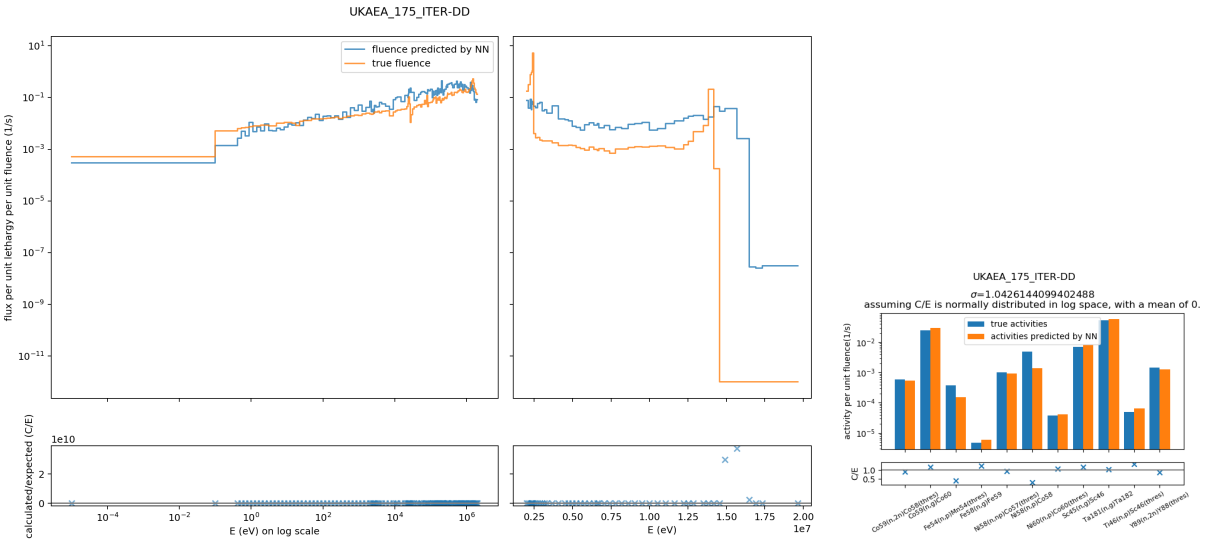


Figure 14: (calculated) ITER spectrum as predicted by the optimal NN among all NN trained on fission spectra

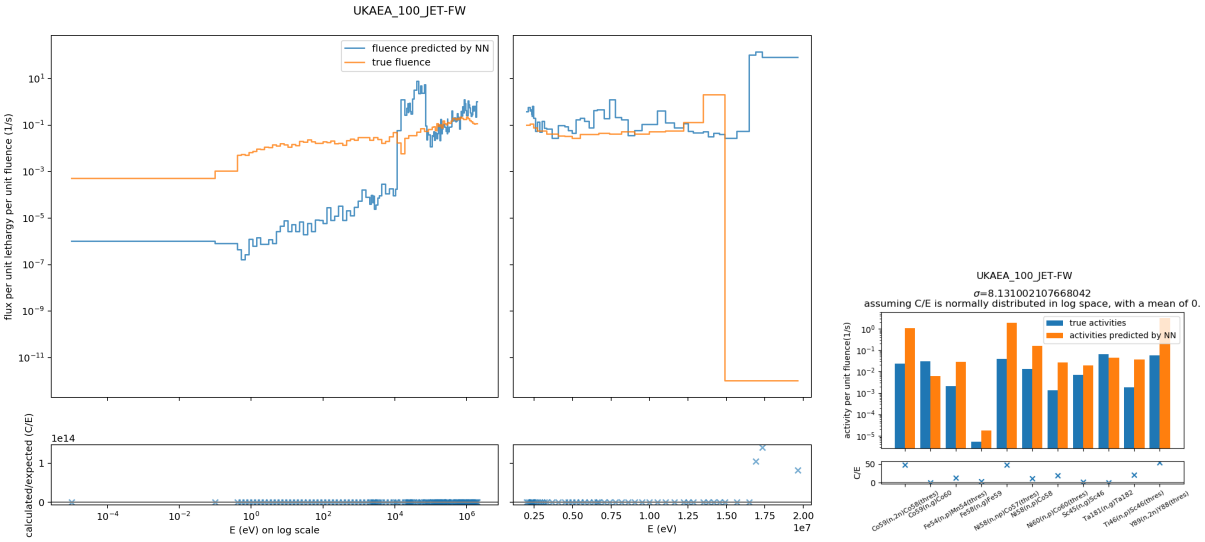


Figure 15: JET first wall spectrum as predicted by the optimal NN among all NN trained on fission spectra

8 Future improvements

- Transfer learning: start with fission data, fix the weights of the second half of the matrix as it gives the connection
- Use RBF NN or GRNN, which are known to perform better under low sample number conditions, though it is more complicated to implement.
- infer the uncertainty (σ) associated with the neural network's prediction using Monte Carlo method

9 Conclusion

- What's the loss values
- achieved using what structure of NN
- trained upon what data
- How does it compare to neutron spectrum unfolding using other methods
- What's the significance on the unfolding community: should they use it more? Should they improve upon it?
- what additional observation did you find regarding training on different dataset.

References

[1] Culham Centre for Fusion Energy. Ukaea: Fistact-ii, 2017.

[2] Slobodan Krevinac. *Neutron energy spectrum unfolding method.* PhD thesis, University of Birmingham, Birmingham, 1971.

[3] Manfred Matzke. Unfolding methods. *Physikalisch-Technische Bundesanstalt, Germany*, 2003.

[4] L. W. Packer, P. Batistoni, S. C. Bradnam, S. Conroy, Z. Ghani, M. R Gilbert, E. Laszyska, I. Lengar, C.R. Nobs, M. Pillon, S. Popovichev, P. Raj, I.E. Stamatelatos, T. Vasilopoulou, A. Wojcik-Gargula, R. Worrall, and JET contributors. Neutron spectrum and fluence determination at the iter material irradiation stations at jet. 5 2019.

[5] Lee W. Packer, Mark R. Gilbert, Jonathan Naish, and Steven Bradnam. Ukaea-r-17-nt1 unfolding code support: progress report. 3 2017.