Pin to pin neutron flux reconstruction in a PWR reactor using support vector regression (SVR) technique

W.F.P. Neto¹, A.C.M. Alvim^{1,*}, F.C. Silva¹, and L.G.M. Alvim²

¹ Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia – COPPE/UFRJ, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil

² Universidade Federal Rural do Rio de Janeiro, Rio de Janeiro, Brazil

Received: 17 May 2018 / Received in final form: 18 October 2018 / Accepted: 11 December 2018

Abstract. Coarse mesh nodal methods are widely used in the analysis of nuclear reactors. However, these methods provide only average values of the neutron fluxes. From a safety point of view, it is important to have an accurate analysis of the pin to pin flux distribution that nodal methods are not able to provide. Many articles have been published that make use of mathematical techniques to determine flux distributions. Most of these techniques use expansion functions to estimate these distributions. The expansion coefficients of these works are determined by conditions that take into account the average values of certain fluxes supplied by the nodal methods. There are also methods that employ analytical solutions of the neutron diffusion equation. This article presents a different approach for calculating the pin to pin neutron flux distribution for a PWR reactor. The developed method uses support vector regression (SVR) technique to determine this pin to pin neutron flux. The SVR technique uses average data computed with the Nodal Expansion Method (NEM) for learning purposes. A total of 70% of the computed data were used for training and 30% for validation, using multifold-cross-validation. Two fuel elements were removed from the training and validation sets, to test the method. Less than 2% errors were found when compared to the values obtained by the nodal expansion method (NEM), using a fine-mesh spatial discretization. We concluded that use of SVR to reconstruct pin to pin fluxes is another option, which will be of great value in fuel reload calculations, since the same parameters will be applied to all cycles, thus expediting calculations when compared to standard procedure calculations.

1 Introduction

The nodal expansion method (NEM) [1] is widely used in reactor physics calculations. This method divides the reactor core into well-defined volumes, called nodes (n), with cross sections of the order of the fuel element cross-sectional dimensions. NEM is only able to provide nodal average results, as the nodal average neutron fluxes ($\bar{\phi}_g^n$), and if we are treating thermal reactors, usually two neutron groups are used. However, these values do not include detailed information of the neutron flux at each position of the fuel element, namely, the pin to pin neutron flux ($\phi_{g,\text{hom}}(x, y)$).

To overcome this problem, flux pin to pin reconstruction methods have been developed since the 1970s. The fundamental problem of reconstruction is precisely how to determine the pin to pin flux ($\phi_{g,\text{hom}}(x, y)$) from values of the nodal average fluxes determined by coarse mesh methods. Some of these methods use polynomial expansions,

others use analytic solutions to determine $\phi_{g,\text{hom}}(x, y)$. Therefore, flux reconstruction methods differ only on how to represent the function $\phi_{g,\text{hom}}(x, y)$.

The central idea of flux reconstruction is the determination of a function to represent the flux distribution within a fuel element. For methods using polynomial flux expansions, the expansion constants are determined by known values of nodal (average) fluxes, total and partial currents and fluxes on the faces of the nodes. In some methods, even the values of fluxes at the node corners are used. Since these values are not generated by NEM, specific procedures have to be developed.

Examining in a more detailed way the problem of reconstruction, it can be concluded that all of these procedures make use of correlations between the average nodal fluxes, determined by coarse mesh nodal methods, and pin to pin neutron flux values. These parameters have to be determined in order to obtain important safety related variables, like hot channel factors. The problem is that directly obtaining pin to pin fluxes is a time-consuming procedure, which justifies the use of coarse mesh nodal methods in conjunction with reconstruction techniques.

* e-mail: alvim@con.ufrj.br

This article presents a different approach for calculating neutron flux pin to pin distribution for a PWR reactor. The method developed here uses support vector regression (SVR) technique to determine pin to pin neutron fluxes. We can summarize this procedure in the following way: For a set of entry points (x_i, y_i) , with x_i representing the values of the nodal fluxes and y_i representing the values of the pin to pin flux, reconstruction can be accomplished by using machine learning technique. Specifically, the machine learning technique employed in this work uses support vector regression (SVR), with statistical learning technique developed by Vapnik [2].

The reactor studied has 121 fuel elements in its core. NEM has given 1936 samples with 19 representations, i.e. average nodal flux, six total currents, six partial currents on node faces and six average face fluxes. With the SVR technique, we were able to compute pin fluxes, so that the error for the fuel elements tested were lower than 2%. Pessoa and Martinez [3] have found errors as large as 14%. In this work, a new procedure is proposed to reconstruct the detailed pin to pin flux distribution in a PWR reactor, and results obtained demonstrate the feasibility of the procedure.

2 Literature review

The work by Koebke and Wagner [4], in 1977, represented $\phi_{g,\text{hom}}(x, y)$ by two-dimensional polynomials for two groups. This polynomial expansion is

$$\phi_{g,\text{hom}}^n(x, y) = \sum_{i,j=0}^4 c_{ij,g}^n x^i y^j; g = 1, 2. \quad (1)$$

The coefficients of this expansion are determined by using the values of the average nodal fluxes determined with the coarse mesh nodal method. In 1985, Koebke and Hetzel [5] used a polynomial expansion, according to equation (1), in order to characterize the fast flux ($g = 1$), and a hyperbolic function expansion for the thermal flux ($g = 2$).

$$\phi_{2,\text{hom}}^n(x, y) = \phi_{1,\text{hom}}^n(x, y) \sum_{i,j=0}^4 c_{ij,2}^n \eta_i(x) \eta_j(y),$$

with $c_{33,2}^n = c_{34,2}^n = c_{43,2}^n = c_{44,2}^n = 0$, $\eta_0(\xi) = 1$, $\eta_1(\xi) = \sin h(\alpha_n \xi)$, $\eta_2(\xi) = \cos h(\alpha_n \xi)$, $\eta_3(\xi) = \sin h(2\alpha_n \xi)$ and $\eta_4(\xi) = \cos h(2\alpha_n \xi)$, $\xi = x, y$, $\alpha_n = h\sqrt{\sum a_2^n / D_2^n}$, h being the width of the fuel element.

Rempe et al. [6] also used polynomial expansions to determine the homogeneous distribution of fast neutron flux equation (1) and hyperbolic functions to represent the homogeneous distribution of thermal neutron flux. However, with a different approach than Koebke and Hetzelt [5] for the first term expansion,

$$\phi_{2,\text{hom}}^n(x, y) = c_{00}\phi_{1,\text{hom}}^n(x, y) + \sum_{\substack{i,j=0 \\ i \neq j}}^4 c_{ij,2}^n \eta_i(x) \eta_j(y).$$

Recently, Pessoa and Martinez [3] represented the pin to pin flux by a fourth-order expansion with a two-

dimensional axial leakage treatment, so that diffusion equation can be solved analytically. This solution is a sum of a particular solution of the non-homogeneous equation and the general solution of the homogeneous differential equation. The boundary conditions are the four-total current on the surfaces of the node and four average fluxes in the node corners. According to Pessoa and Martinez [3], the larger errors occur in assemblies near baffle-reflector interfaces. One of the reasons pointed out there is that in the nuclear data generation for these assemblies resulted in important information being lost. Errors can be as large as 14.6% in those regions.

It is apparent from this discussion that flux reconstruction techniques are kind of an art, since there is no sound basis for choosing the flux form to be used. This has motivated investigating the use of techniques employing Artificial Intelligence (AI), which could be trained with real data, hoping to find smaller errors than in the literature. From the techniques available, we decided to exploit the SVR algorithm, of which a brief description follows.

The SV algorithm is a generalization of the algorithm developed by Vapnik and Lerner [7] in 1963 and Vapnik and Chervonenkis [8] in 1964. In the following years, Vapnik [2] continued to develop the technique that is characterized by statistical learning or VC theory. First, these algorithms were applied to classification problems. Currently, in addition to classification, the theory was extended to regression problems. The support vectors regression technique was applied to predict time series by Müller et al. [9] Drucker et al. [10] and Mattera and Haykin [11] in the late 1990s.

3 Support vectors regression (SVR)

Support vectors regression presents the principle of structural risk minimization, which considers the minimization of the upper limit of the generalization error [12]. The expected risk is defined as a function of the empirical risk, R_{emp} , which measures the error rate in a training set for a number of finite and fixed observations of the problem:

$$R_{\text{emp}} = \frac{1}{N} \sum_{i=1}^N L(y_i, f(x_i)). \quad (2)$$

In equation (2), N is the number of points in the training set and $L(x_i, f(x_i))$ is the loss function, assumed quadratic in this work:

$$L = (y_i - f(x_i))^2.$$

Thus, the L loss function corresponds, in our problem, to the difference between a target neutron flux and the estimated one. Moreover, the expected risk R_{emp} corresponds to the average error of the estimated neutron flux. Hence, higher values of R_{emp} means higher error and consequently higher risk of poor estimation of neutron fluxes.

Considering that we are looking for linear functions that relate the input data x_i to the desired values y_i through $f(x_i) = w \cdot x_i + b$, with $b \in R$ and $w \in X$, where $w \cdot x_i$ is the inner product in space X , the expected risk equation for

N training samples is

$$R_{\text{exp}} = R_{\text{emp}} + \frac{1}{2} \|w\|^2. \quad (3)$$

The structural risk minimization was proposed by Vapnik et al. [13] and it consists of reducing the value on the right of equation (3) in order to obtain the smallest expected risk. The SVR attempts to find the function f which relates the values x_i to the values y_i , with a maximum deviation ε for the training data. Therefore, a space is created where all points with acceptable errors are within the interval $[y_i - \varepsilon, y_i + \varepsilon]$. The SVR focuses on minimizing the second term of equation (6) such that the problem is to determine:

$$\text{Min}_{w,b} \frac{1}{2} \|w\|^2,$$

subject to

$$\begin{cases} y_i - w \cdot x_i - b \leq \varepsilon \\ -y_i + w \cdot x_i + b \leq \varepsilon. \end{cases}$$

Equation (3) does not take into account the results outside the range set by the choice of ε .

For the points outside this range to be considered, slack variables ξ and ξ^* are introduced into the optimization problem.

$$\text{Min}_{w,b} \frac{1}{2} \|w\|^2 + C \left(\sum_{i=1}^N (\xi_i + \xi_i^*) \right),$$

subject to

$$\begin{cases} y_i - w \cdot x_i - b \leq \varepsilon + \xi_i \\ -y_i + w \cdot x_i + b \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0, \end{cases}$$

where C is a constant to be chosen and determines the trade-off among f and points x_i with errors greater than ε . The choice of C implies in determining w , which minimizes the slack variables. A loss function is used in this work to characterize the points outside ε , the ε -insensitive loss function, defined by equation (4):

$$|\xi|_\varepsilon = \begin{cases} 0, & \text{if } |\xi| \leq \varepsilon \\ |\xi|, & \text{otherwise.} \end{cases} \quad (4)$$

This primal optimization problem is difficult due to the inequalities in the constraints equations. Therefore, a dual problem is developed as an easier alternative to solve the optimization problem. This transformation from primal to dual problem considers a Lagrangian function having Lagrange multipliers as dual variables for each constraint of the primal problem. The dual problem Lagrangian function is given by

$$\begin{aligned} L = & \frac{1}{2} \|w\|^2 + C \left(\sum_{i=1}^N (\xi_i + \xi_i^*) \right) - \sum_{i=1}^N (\eta_i^* \xi_i + \eta_i \xi_i^*) \\ & - \sum_{i=1}^N \alpha_i (\varepsilon + \xi_i - y_i + w \cdot x_i + b) \\ & - \sum_{i=1}^N \alpha_i^* (\varepsilon^* + \xi_i^* - y_i + w \cdot x_i + b), \end{aligned} \quad (5)$$

where $\alpha_i, \alpha_i^*, \eta_i^*, \eta_i$ are Lagrange multipliers restricted to be ≤ 0 . Then, the problem becomes a problem of maximizing dual variables. On the other hand, the Lagrange function L will be minimized by deriving it with respect to each primal variable and equating these derivatives to zero:

$$\frac{\partial L}{\partial b} = \sum_{i=1}^N (\alpha_i - \alpha_i^*) = 0 \quad (6)$$

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^N (\alpha_i - \alpha_i^*) x_i = 0 \quad (7)$$

$$\frac{\partial L}{\partial \xi_i} = C - \alpha_i - \eta_i = 0 \quad (8)$$

$$\frac{\partial L}{\partial \alpha_i^*} = C - \alpha_i^* - \eta_i^* = 0. \quad (9)$$

Substituting equations (6)–(9) in equation (5), we have the dual problem as

$$\begin{aligned} \text{Max} \left\{ -\frac{1}{2} \sum_{\substack{i=1,N \\ j=1,N}} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)(x_i \cdot x_j) \right. \\ \left. - \varepsilon + \sum_{i=1}^N y_i (\alpha_i - \alpha_i^*) \right\}, \end{aligned} \quad (10)$$

subject to $\sum_{i=1}^N (\alpha_i - \alpha_i^*) = 0, \quad \forall \alpha_i, \alpha_i^* \in [0, C]$.

From equation (7), it can be seen that

$$w = \sum_{i=1}^N (\alpha_i - \alpha_i^*) x_i,$$

which determines $f(x)$ as

$$f(x) = \sum_{i=1}^N (\alpha_i - \alpha_i^*) x_i + b.$$

Therefore, to determine f , it is not necessary to determine w , which is a primal variable. Lagrange multipliers are obtained from the dual problem. The Karush–Kuhn–Tucker (KKT) conditions are used to interact between primal and dual problems as follows:

$$\alpha_i (\varepsilon + \xi_i - y_i + w \cdot x_i + b) = 0 \quad (11)$$

$$\alpha_i^* (\varepsilon^* + \xi_i^* - y_i + w \cdot x_i + b) = 0. \quad (12)$$

Equations (11) and (12) establish the first complementary KKT condition, where the dual variables α_i and α_i^* are different from zero when

$$|(f(x_i - y_i) \geq \varepsilon|. \quad (13)$$

From equation (13), it can be concluded that it is not necessary to use all points x_i to determine w but only those that make α_i and α_i^* greater than zero. These points are the so-called support vectors [14].

To calculate b , the complementary KKT conditions are used with

$$(C - \alpha_i)\xi_i = 0, \quad (14)$$

$$(C - \alpha_i^*)\xi_i^* = 0. \quad (15)$$

Since α_i and α_i^* in equations (14) and (15) must be different from zero, it results that the slack variables must be equal to zero. Thus, b can be determined by taking the points where Lagrangian multipliers vary on the interval $(0, C)$. Now using equations (11) and (12), we get

$$b = \frac{1}{N} \sum_{i=1}^N (\alpha_i - \alpha_i^*)(\varepsilon - \xi_i) - w \cdot x_i.$$

Now, define a kernel function

$$K(x_i, y_i) = \varphi(x_i) \cdot \varphi(y_i), \quad \forall x_i, y_i \in X.$$

Substituting this kernel function into the dual problem equations 15, we have

$$\text{Max} \left\{ -\frac{1}{2} \sum_{i=1, N} \sum_{j=1, N} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) K(x_i, x_j) - \varepsilon + \sum_{i=1}^N y_i (\alpha_i - \alpha_i^*) \right\},$$

subject to: $\sum_{i=1}^N (\alpha_i - \alpha_i^*) = 0, \quad \forall \alpha_i, \alpha_i^* \in [0, C]$.

Function f will now be determined by making

$$w = \sum_{i=1}^N (\alpha_i - \alpha_i^*) \varphi(x_i).$$

And then

$$f(x) = \sum_{i=1}^N (\alpha_i - \alpha_i^*) K(x_i, y_i) + b.$$

Finally, the kernel function used in this work is the radial basis function (RBF) presented below:

$$K(x_i, y_i) = \exp(-\gamma \|x_i - y_i\|^2).$$

Fig. 1. Core configuration.

4 Application of SVR methodology to pin to pin flux reconstruction for a PWR nuclear reactor

The reactor studied in this work is a PWR reactor similar to Angra-1. This reactor has 121 (Fig. 1) 20×20 cm fuel elements of 3.60 m height, set in a 16×16 array (Fig. 2). Each element contains 256 pins distributed among nuclear fuel, burnable poison, water holes, etc. The fuel element nuclear parameters were homogenized, and the reactor was discretized using $20 \times 20 \times 20$ cm nodes, totaling 1936 nodes: 121 nodes in x - and y -directions with 16 layers in the z -direction. These data are important for the nodal calculation with NEM. The nodes corresponding to lower and upper reflectors were discarded because they do not contribute to power generation.

4.1 Fundamental problem

NEM is applied to this discretized space in order to determine average values for fluxes and currents. In this process, as said earlier, the heterogeneity of information is lost, since NEM makes use of node homogenized cross sections. However, it is important for fuel cycle design and for safety reasons that pin to pin fluxes in every fuel element be determined for the whole core. Then, considering the average nodal flux values, total currents and partial fluxes at each face of the node, determined by NEM, which are input data for the SVR technique, one can obtain pin to pin neutron fluxes (representing the target data) in each of the 256 pins constituents of the fuel assembly. This study has focused on obtaining the thermal flux pin values, since the same procedure can be performed for the fast group. Therefore, from this point on, the $\phi_{g, \text{hom}}(x, y)$ shall be referred to as $\phi_{\text{hom}}(x, y)$.

To determine the optimal values for the parameter C , the penalty factor, and γ , a parameter of the RBF kernel function, the learning process was attained through an automatic search process [12]. The training and validation sets were divided into a proportion of 70–30% for training and validation purposes, using the multifold-

241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256
225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240
209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224
193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208
177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	192
161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176
145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160
129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144
113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128
97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112
81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96
65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Fig. 2. Fuel assembly pin identification.

cross-validation technique. After the learning process was done, the algorithm could use the pair (C, γ) for reconstruction purposes.

The data were processed according to the normalization:

$$x' = \frac{x - \min(X)}{\max(X) - \min(X)} (\text{New}_{\max} - \text{New}_{\min}) + \text{New}_{\min},$$

where x is a vector space X and $\text{New}_{\max}$ and $\text{New}_{\min}$ are chosen according to the purpose. In this work, the new range will be between $[-1, 1]$. The reference values for the pin fluxes were obtained from NEM, with a fine spatial discretization mesh, with dimensions on the order of the fuel rod size.

4.2 Methodology

First, the reconstruction of the pin to pin flux was done for two fuel rods, pin 01 and pin 170, in fuel assembly 01, as shown in Figures 1 and 2. Then, for a first test, one fuel element was removed for testing. Thus, a total of 120 sample fuel assemblies were available for training and validation purposes, with each instance containing a vector entry of order $[16 \times 19]$, because each instance has the values of the 16 fuel assembly layer divisions. Each example was correlated with 16 values of the pin to pin flux, one for each layer. So, for the test, a set of $[16 \times 19]$ was presented to the algorithm and the output was a vector of $[16 \times 1]$, regarding the neutron flux values at pin 01 of the element 01. It will be shown in Section 4.1 that determining a value for C and γ parameters to reconstruct the flux pin to pin for a complete fuel rod, at a given time, is not efficient. To solve this problem, we presented 16 pairs (C, γ) for the algorithm, one pair for each layer.

To examine the new methodology of training/validation and testing, all data from two selected fuel elements were removed, the first near the side reflector,

Fig. 3. Pin to pin flux distribution for assembly 53 on the first layer.

fuel element 53, and the second near the center of the reactor configuration, 87. The reconstruction was carried out for fuel element 53, but with a pair (C, γ) for each pin layer. This was taken into consideration because it was found that for a given fuel element, the values of the pin to pin fluxes for each layer do not present large variations (Fig. 3).

Finally, the reconstruction for fuel element 87 was performed using the same parameters used to reconstruct the fluxes for fuel element 53. Equation (8) shows the calculation of the error (e), where $(\phi_{\text{hom}}^*(x, y))$ is the pin flux determined by the algorithm and $(\phi_{\text{hom},R}(x, y))$ is the neutron flux in the reference pin, that is, the one obtained with fine mesh NEM discretization. For the training error, we have chosen the maximum value of 0.001% and for validation, a maximum error of 0.01%.

Fig. 4. Comparison of the pin to pin reference fluxes and those estimated by the algorithm for pin 01 of assembly 01.

Fig. 5. Comparison of the pin to pin of reference fluxes and those estimated by the algorithm for pin 170 of assembly 01.

0.188	0.192	0.194	0.197	0.200	0.202	0.202	0.201	0.201	0.203	0.203	0.204	0.199	0.197	0.193	0.188	Reference Estimated Error
0.189	0.192	0.196	0.199	0.202	0.204	0.204	0.203	0.203	0.204	0.205	0.203	0.200	0.196	0.193	0.189	
0.58%	0.23%	1.03%	1.14%	1.21%	0.84%	0.86%	0.95%	1.23%	0.58%	0.82%	0.22%	0.62%	0.15%	0.06%	0.64%	
0.189	0.195	0.197	0.202	0.206	0.212	0.208	0.206	0.207	0.209	0.213	0.208	0.202	0.198	0.193	0.189	
0.188	0.193	0.198	0.204	0.209	0.214	0.210	0.208	0.209	0.212	0.215	0.210	0.205	0.199	0.193	0.188	
0.54%	1.22%	0.70%	1.18%	1.19%	0.79%	1.06%	0.84%	0.89%	1.15%	0.98%	1.16%	1.19%	0.37%	0.43%	0.38%	
0.191	0.197	0.204	0.213	0.222	0.234	0.216	0.213	0.218	0.221	0.238	0.224	0.214	0.203	0.198	0.188	
0.188	0.196	0.205	0.215	0.221	0.237	0.219	0.216	0.220	0.223	0.240	0.221	0.216	0.205	0.196	0.189	
1.24%	0.55%	0.38%	1.05%	0.77%	1.23%	1.21%	1.06%	0.97%	1.13%	0.83%	0.99%	0.93%	1.23%	0.97%	0.34%	
0.192	0.200	0.213	0.235	0.224	0.222	0.218	0.220	0.240	0.226	0.225	0.223	0.235	0.212	0.200	0.188	
0.190	0.200	0.214	0.238	0.226	0.225	0.220	0.222	0.241	0.228	0.227	0.226	0.238	0.213	0.200	0.190	
1.04%	0.10%	0.64%	1.16%	0.69%	1.19%	0.92%	1.18%	0.56%	0.82%	0.68%	1.18%	1.05%	0.57%	0.09%	1.16%	
0.191	0.202	0.216	0.223	0.222	0.225	0.218	0.218	0.222	0.221	0.226	0.222	0.222	0.215	0.202	0.190	
0.193	0.204	0.219	0.225	0.225	0.226	0.221	0.220	0.224	0.224	0.227	0.224	0.224	0.217	0.204	0.193	
1.01%	1.20%	1.13%	1.10%	0.96%	0.62%	1.19%	0.96%	1.00%	1.19%	0.63%	0.96%	0.97%	1.26%	1.12%	1.06%	
0.194	0.208	0.235	0.223	0.225	0.242	0.222	0.216	0.216	0.222	0.241	0.222	0.220	0.232	0.207	0.194	
0.194	0.209	0.236	0.226	0.227	0.243	0.224	0.219	0.219	0.224	0.242	0.225	0.222	0.233	0.208	0.194	
0.29%	0.72%	0.69%	1.06%	1.21%	0.48%	0.99%	1.18%	1.18%	0.87%	0.32%	1.18%	0.94%	0.42%	0.37%	0.04%	
0.195	0.204	0.218	0.223	0.222	0.223	0.221	0.221	0.217	0.217	0.220	0.216	0.215	0.214	0.202	0.195	
0.194	0.206	0.220	0.225	0.224	0.225	0.222	0.222	0.218	0.218	0.221	0.218	0.217	0.215	0.204	0.193	
0.78%	1.06%	0.61%	1.18%	0.89%	0.88%	0.42%	0.82%	0.81%	0.42%	0.75%	0.83%	0.86%	0.54%	0.80%	1.17%	
0.191	0.203	0.215	0.237	0.221	0.217	0.221	0.000	0.219	0.213	0.213	0.214	0.217	0.210	0.202	0.191	
0.193	0.203	0.217	0.239	0.223	0.219	0.222	0.000	0.220	0.214	0.215	0.217	0.219	0.212	0.202	0.192	
0.82%	0.12%	0.83%	0.96%	1.22%	1.02%	0.78%	0.00%	0.52%	0.73%	0.73%	1.12%	1.15%	0.92%	0.31%	0.36%	
0.190	0.202	0.210	0.218	0.215	0.215	0.216	0.219	0.237	0.212	0.213	0.219	0.236	0.215	0.203	0.192	
0.192	0.202	0.212	0.220	0.218	0.217	0.218	0.220	0.239	0.213	0.215	0.221	0.238	0.216	0.203	0.193	
0.78%	0.15%	1.04%	1.00%	1.20%	1.00%	0.73%	0.36%	0.54%	0.66%	0.78%	1.03%	0.74%	0.51%	0.31%	0.54%	
0.195	0.202	0.214	0.214	0.217	0.220	0.217	0.213	0.212	0.213	0.220	0.221	0.222	0.218	0.204	0.196	
0.193	0.203	0.215	0.217	0.218	0.222	0.217	0.214	0.213	0.215	0.221	0.222	0.224	0.219	0.205	0.194	
1.24%	0.53%	0.34%	1.20%	0.68%	0.62%	0.24%	0.67%	0.56%	1.20%	0.54%	0.56%	0.79%	0.36%	0.51%	1.01%	
0.193	0.206	0.232	0.219	0.222	0.241	0.221	0.213	0.212	0.219	0.240	0.223	0.222	0.234	0.206	0.195	
0.193	0.207	0.233	0.222	0.224	0.241	0.222	0.215	0.214	0.220	0.240	0.225	0.224	0.235	0.208	0.194	
0.09%	0.40%	0.24%	1.11%	0.95%	0.07%	0.61%	0.86%	0.81%	0.41%	0.04%	0.97%	1.03%	0.18%	1.17%	0.30%	
0.191	0.200	0.214	0.220	0.220	0.223	0.220	0.219	0.215	0.216	0.221	0.220	0.222	0.215	0.202	0.192	
0.192	0.203	0.216	0.222	0.222	0.225	0.221	0.221	0.216	0.217	0.223	0.222	0.223	0.217	0.203	0.192	
0.66%	1.21%	1.12%	0.79%	1.00%	0.75%	0.43%	0.79%	0.80%	0.48%	0.75%	1.05%	0.81%	0.82%	0.76%	0.37%	
0.188	0.200	0.209	0.232	0.221	0.222	0.221	0.235	0.216	0.213	0.219	0.221	0.232	0.209	0.200	0.189	
0.189	0.198	0.211	0.234	0.223	0.223	0.224	0.237	0.218	0.216	0.221	0.222	0.235	0.211	0.198	0.190	
0.84%	0.89%	1.11%	0.86%	0.64%	0.74%	1.23%	1.03%	1.20%	1.25%	0.83%	0.64%	0.98%	1.11%	0.81%	0.48%	
0.187	0.196	0.202	0.209	0.215	0.231	0.218	0.214	0.209	0.214	0.228	0.214	0.209	0.202	0.191	0.186	
0.187	0.193	0.201	0.211	0.216	0.234	0.218	0.214	0.210	0.213	0.231	0.215	0.211	0.201	0.193	0.187	
0.01%	1.19%	0.44%	0.79%	0.76%	1.24%	0.18%	0.08%	0.27%	0.20%	1.27%	0.79%	0.86%	0.41%	0.90%	0.59%	
0.183	0.188	0.191	0.198	0.201	0.204	0.204	0.198	0.201	0.202	0.203	0.200	0.195	0.191	0.188	0.184	
0.185	0.188	0.193	0.197	0.202	0.206	0.203	0.200	0.199	0.201	0.205	0.201	0.197	0.192	0.188	0.185	
1.19%	0.06%	0.98%	0.14%	0.42%	1.25%	0.45%	1.24%	1.03%	0.11%	1.25%	0.52%	0.69%	0.40%	0.07%	0.50%	
0.183	0.183	0.185	0.188	0.191	0.189	0.188	0.191	0.191	0.187	0.188	0.190	0.188	0.185	0.184	0.182	
0.184	0.184	0.186	0.188	0.190	0.191	0.191	0.189	0.189	0.190	0.190	0.189	0.187	0.185	0.184	0.184	
0.65%	0.33%	0.13%	0.31%	0.67%	1.19%	1.29%	0.93%	0.88%	1.25%	1.19%	0.57%	0.40%	0.09%	0.05%	0.99%	

Fig. 6. Comparing the reference to estimated fluxes by the algorithm, for each pin of the first layer of assembly 53.

1.719	1.728	1.724	1.722	1.714	1.709	1.717	1.718	1.718	1.720	1.706	1.718	1.714	1.733	1.712	1.719	
1.716	1.714	1.716	1.719	1.716	1.720	1.724	1.716	1.714	1.716	1.724	1.716	1.722	1.721	1.721	1.721	
0.13%	0.80%	0.46%	0.20%	0.13%	0.63%	0.44%	0.14%	0.21%	0.23%	1.01%	0.12%	0.44%	0.66%	0.58%	0.15%	
1.724	1.692	1.722	1.714	1.722	1.736	1.715	1.714	1.715	1.724	1.710	1.721	1.721	1.726	1.711	1.708	
1.719	1.721	1.716	1.722	1.714	1.719	1.723	1.715	1.720	1.720	1.716	1.714	1.716	1.721	1.717	1.716	
0.32%	1.71%	0.33%	0.47%	0.50%	1.03%	0.44%	0.07%	0.27%	0.24%	0.36%	0.41%	0.28%	0.28%	0.36%	0.49%	
1.716	1.722	1.716	1.710	1.728	1.710	1.720	1.724	1.720	1.714	1.706	1.708	1.706	1.725	1.720	1.694	
1.716	1.721	1.721	1.720	1.716	1.723	1.718	1.720	1.718	1.713	1.717	1.721	1.718	1.716	1.718	1.711	
0.01%	0.09%	0.31%	0.58%	0.67%	0.77%	0.15%	0.19%	0.13%	0.03%	0.66%	0.79%	0.67%	0.52%	0.06%	1.01%	
1.710	1.721	1.719	1.713	1.713	1.737	1.714	1.737	1.719	1.719	1.740	1.712	1.718	1.715	1.712	1.725	
1.715	1.711	1.715	1.716	1.719	1.716	1.721	1.721	1.722	1.716	1.718	1.718	1.717	1.718	1.715	1.715	
0.28%	0.61%	0.26%	0.15%	0.33%	1.18%	0.41%	0.92%	0.17%	0.18%	1.24%	0.36%	0.04%	0.21%	0.20%	0.58%	
1.723	1.719	1.718	1.727	1.716	1.714	1.719	1.715	1.712	1.717	1.713	1.720	1.724	1.717	1.727	1.722	
1.718	1.712	1.720	1.718	1.718	1.718	1.712	1.721	1.720	1.720	1.718	1.723	1.718	1.721	1.718	1.721	
0.30%	0.44%	0.10%	0.51%	0.11%	0.25%	0.42%	0.34%	0.46%	0.19%	0.30%	0.18%	0.32%	0.23%	0.49%	0.06%	
1.719	1.724	1.719	1.725	1.723	1.714	1.716	1.716	1.716	1.719	1.716	1.717	1.714	1.709	1.725	1.722	Reference
1.719	1.721	1.716	1.717	1.721	1.719	1.718	1.720	1.720	1.723	1.722	1.719	1.720	1.717	1.722	1.715	Estimated
0.02%	0.17%	0.17%	0.43%	0.12%	0.29%	0.14%	0.23%	0.21%	0.22%	0.34%	0.09%	0.32%	0.48%	0.17%	0.40%	Error
1.728	1.731	1.720	1.724	1.722	1.720	1.720	1.718	1.720	1.719	1.717	1.713	1.714	1.714	1.728	1.722	
1.717	1.717	1.719	1.720	1.718	1.718	1.719	1.714	1.715	1.718	1.714	1.719	1.721	1.722	1.712	1.713	
0.60%	0.82%	0.04%	0.27%	0.26%	0.12%	0.07%	0.23%	0.30%	0.05%	0.17%	0.38%	0.41%	0.49%	0.89%	0.54%	
1.725	1.733	1.720	1.718	1.717	1.724	1.730	0.000	1.716	1.727	1.727	1.725	1.722	1.716	1.703	1.733	
1.717	1.721	1.722	1.716	1.716	1.723	1.719	0.000	1.713	1.715	1.714	1.721	1.718	1.719	1.719	1.721	
0.48%	0.70%	0.10%	0.14%	0.06%	0.10%	0.67%	0.00%	0.18%	0.71%	0.71%	0.25%	0.24%	0.18%	0.94%	0.69%	
1.724	1.705	1.725	1.720	1.719	1.716	1.719	1.723	1.728	1.712	1.718	1.717	1.717	1.721	1.707	1.731	
1.712	1.722	1.719	1.717	1.717	1.721	1.715	1.721	1.722	1.717	1.713	1.721	1.715	1.719	1.713	1.719	
0.67%	1.00%	0.34%	0.13%	0.12%	0.32%	0.26%	0.10%	0.34%	0.33%	0.31%	0.21%	0.09%	0.13%	0.35%	0.70%	
1.729	1.738	1.713	1.721	1.716	1.721	1.717	1.717	1.717	1.720	1.733	1.719	1.723	1.717	1.693	1.742	
1.721	1.723	1.721	1.720	1.718	1.721	1.721	1.723	1.724	1.716	1.721	1.719	1.716	1.722	1.717	1.711	
0.50%	0.87%	0.44%	0.05%	0.10%	0.05%	0.21%	0.34%	0.38%	0.23%	0.65%	0.01%	0.38%	0.29%	1.40%	1.76%	
1.735	1.704	1.712	1.713	1.714	1.719	1.713	1.712	1.721	1.718	1.713	1.714	1.713	1.714	1.699	1.738	
1.719	1.719	1.717	1.719	1.720	1.717	1.719	1.719	1.717	1.715	1.721	1.722	1.720	1.720	1.719	1.722	
0.94%	0.86%	0.28%	0.33%	0.36%	0.14%	0.39%	0.41%	0.21%	0.14%	0.46%	0.43%	0.44%	0.32%	1.19%	0.87%	
1.730	1.707	1.712	1.725	1.716	1.720	1.718	1.717	1.720	1.724	1.721	1.710	1.719	1.713	1.706	1.716	
1.720	1.723	1.720	1.719	1.718	1.716	1.721	1.713	1.715	1.719	1.719	1.718	1.719	1.720	1.717	1.718	
0.57%	0.89%	0.46%	0.35%	0.10%	0.21%	0.16%	0.25%	0.25%	0.34%	0.09%	0.52%	0.02%	0.41%	0.66%	0.08%	
1.732	1.721	1.695	1.713	1.708	1.725	1.720	1.711	1.715	1.720	1.715	1.714	1.712	1.705	1.724	1.716	
1.720	1.718	1.720	1.718	1.718	1.721	1.718	1.722	1.718	1.718	1.720	1.723	1.717	1.719	1.718	1.718	
0.72%	0.17%	1.42%	0.33%	0.60%	0.25%	0.13%	0.62%	0.17%	0.10%	0.31%	0.52%	0.24%	0.82%	0.34%	0.16%	
1.709	1.710	1.713	1.709	1.709	1.718	1.714	1.714	1.718	1.717	1.727	1.709	1.713	1.719	1.714	1.702	
1.718	1.716	1.714	1.722	1.721	1.719	1.720	1.712	1.718	1.723	1.714	1.721	1.717	1.717	1.720	1.722	
0.54%	0.32%	0.08%	0.77%	0.70%	0.05%	0.32%	0.10%	0.02%	0.34%	0.76%	0.69%	0.24%	0.16%	0.36%	1.14%	
1.714	1.701	1.712	1.712	1.715	1.719	1.719	1.719	1.714	1.721	1.718	1.712	1.716	1.707	1.710	1.718	
1.716	1.717	1.713	1.721	1.714	1.714	1.715	1.715	1.712	1.718	1.722	1.723	1.713	1.720	1.722	1.718	
0.13%	0.94%	0.06%	0.51%	0.04%	0.31%	0.20%	0.21%	0.12%	0.16%	0.24%	0.69%	0.19%	0.75%	0.72%	0.03%	
1.722	1.722	1.720	1.728	1.716	1.714	1.714	1.725	1.718	1.722	1.723	1.722	1.719	1.726	1.724	1.723	
1.717	1.717	1.715	1.720	1.714	1.715	1.724	1.720	1.724	1.716	1.713	1.720	1.714	1.720	1.720	1.720	
0.31%	0.30%	0.28%	0.49%	0.11%	0.03%	0.61%	0.31%	0.35%	0.32%	0.57%	0.11%	0.29%	0.36%	0.26%	0.18%	

Fig. 7. Ratios between maximum and average values of the pin to pin fluxes for assembly 53.

5 Analysis of results

5.1 Reconstruction for two fuel rods using the same parameters of learning

Figure 4 shows the results obtained for pin 01 of the fuel assembly 01. The training and validation algorithms were performed taking as the target the value of pin number 01 for all fuel assemblies. After the training and validation, the algorithm found the optimum pair (C, γ) for reconstruction of pin 01 flux of fuel assembly 01. Figure 4 shows that the algorithm obtained a 1.17% error in layer 8 for the pin flux. The same learning parameters for the reconstruction of the flux pin to pin 170 were used.

The differences between the reference fluxes and the predicted fluxes are larger for pin 170, as shown in Figure 5,

with a maximum error of 1.3% in layer 3. This is because the same parameters used to estimate the flux of pin 01 were used for pin 170. This result makes clear that each pin has its specificity, and the optimal parameters for a pin do not necessarily apply to others.

5.2 Neutron flux pin to pin reconstruction for two elements

This time, two fuel elements were removed from the training and validation sets. The elements removed for the test were elements 53 and 87, according to Figure 1. Analyzing the first layer element 01 (Fig. 3), it can be seen that the pin to pin neutron fluxes do not have wide variations. The average is given by $0.198584 \text{ cm}^{-2} \text{ s}^{-1}$ and variance and standard deviation are, respectively, 0.000706

Table 1. Maximum errors obtained for all layers of assembly 53.

Layer	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Error (%)	1.25	1.24	1.21	1.21	1.22	1.19	1.22	1.22	1.27	1.26	1.28	1.32	1.30	1.30	1.33	1.30

1.710	1.731	1.725	1.722	1.714	1.709	1.717	1.718	1.716	1.720	1.706	1.718	1.714	1.736	1.712	1.706	
1.716	1.714	1.716	1.719	1.716	1.720	1.724	1.716	1.714	1.716	1.724	1.716	1.722	1.721	1.721	1.721	
0.37%	0.95%	0.51%	0.20%	0.13%	0.64%	0.45%	0.14%	0.06%	0.22%	1.02%	0.14%	0.44%	0.81%	0.56%	0.89%	
1.725	1.700	1.721	1.715	1.722	1.728	1.713	1.712	1.715	1.724	1.709	1.721	1.722	1.726	1.715	1.719	
1.719	1.721	1.716	1.722	1.714	1.719	1.723	1.715	1.720	1.716	1.714	1.716	1.721	1.717	1.717	1.716	
0.39%	1.27%	0.28%	0.38%	0.50%	0.53%	0.58%	0.17%	0.29%	0.24%	0.43%	0.41%	0.34%	0.28%	0.16%	0.14%	
1.728	1.716	1.720	1.712	1.724	1.732	1.720	1.711	1.719	1.717	1.699	1.719	1.706	1.716	1.712	1.728	
1.716	1.721	1.721	1.720	1.716	1.723	1.718	1.720	1.718	1.713	1.717	1.721	1.718	1.716	1.718	1.711	
0.67%	0.28%	0.07%	0.46%	0.47%	0.49%	0.15%	0.52%	0.07%	0.20%	1.09%	0.16%	0.70%	0.01%	0.36%	0.95%	
1.710	1.723	1.719	1.714	1.718	1.710	1.732	1.734	1.718	1.731	1.710	1.717	1.726	1.720	1.724	1.710	
1.715	1.711	1.715	1.716	1.719	1.716	1.721	1.721	1.722	1.716	1.718	1.718	1.717	1.718	1.715	1.715	
0.28%	0.72%	0.21%	0.11%	0.04%	0.37%	0.64%	0.73%	0.24%	0.87%	0.49%	0.08%	0.49%	0.11%	0.48%	0.30%	
1.716	1.720	1.716	1.716	1.716	1.713	1.718	1.717	1.717	1.719	1.713	1.719	1.726	1.716	1.724	1.721	
1.718	1.712	1.720	1.718	1.718	1.718	1.712	1.721	1.720	1.720	1.718	1.723	1.718	1.721	1.718	1.721	
0.10%	0.48%	0.19%	0.11%	0.13%	0.32%	0.34%	0.24%	0.18%	0.07%	0.30%	0.20%	0.44%	0.26%	0.32%	0.01%	
1.718	1.713	1.714	1.723	1.715	1.717	1.716	1.716	1.720	1.719	1.726	1.714	1.716	1.716	1.723	1.715	Reference Estimated Error
1.719	1.721	1.716	1.717	1.721	1.719	1.718	1.720	1.723	1.722	1.719	1.720	1.717	1.722	1.715	1.715	
0.05%	0.44%	0.12%	0.31%	0.35%	0.10%	0.14%	0.22%	0.20%	0.17%	0.15%	0.46%	0.36%	0.09%	0.06%	0.02%	
1.714	1.722	1.718	1.718	1.714	1.721	1.728	1.723	1.724	1.727	1.709	1.715	1.715	1.716	1.721	1.721	
1.717	1.717	1.719	1.720	1.718	1.718	1.719	1.714	1.715	1.718	1.714	1.719	1.721	1.722	1.712	1.713	
0.17%	0.28%	0.05%	0.11%	0.20%	0.20%	0.49%	0.53%	0.53%	0.55%	0.30%	0.27%	0.32%	0.36%	0.50%	0.50%	
1.725	1.721	1.717	1.719	1.712	1.711	1.715	0.000	1.729	1.722	1.712	1.711	1.718	1.713	1.726	1.719	
1.717	1.721	1.722	1.716	1.716	1.723	1.719	0.000	1.713	1.715	1.714	1.721	1.718	1.719	1.719	1.721	
0.48%	0.03%	0.31%	0.19%	0.23%	0.69%	0.23%	0.00%	0.93%	0.45%	0.16%	0.57%	0.03%	0.33%	0.40%	0.11%	
1.721	1.727	1.712	1.718	1.725	1.710	1.719	1.706	1.720	1.711	1.707	1.716	1.721	1.714	1.729	1.724	
1.712	1.722	1.719	1.717	1.717	1.721	1.715	1.721	1.722	1.717	1.713	1.721	1.715	1.719	1.713	1.719	
0.51%	0.28%	0.41%	0.05%	0.47%	0.69%	0.21%	0.89%	0.11%	0.37%	0.36%	0.27%	0.34%	0.32%	0.94%	0.29%	
1.736	1.723	1.723	1.720	1.727	1.703	1.721	1.715	1.716	1.704	1.703	1.706	1.727	1.722	1.718	1.728	
1.721	1.723	1.721	1.720	1.718	1.721	1.721	1.723	1.724	1.716	1.721	1.719	1.716	1.722	1.717	1.711	
0.86%	0.02%	0.16%	0.03%	0.52%	1.01%	0.03%	0.46%	0.46%	0.67%	1.05%	0.79%	0.60%	0.01%	0.07%	0.98%	
1.735	1.720	1.723	1.727	1.724	1.711	1.711	1.731	1.729	1.703	1.712	1.725	1.718	1.719	1.715	1.738	
1.719	1.719	1.717	1.719	1.720	1.717	1.719	1.717	1.715	1.721	1.722	1.720	1.720	1.720	1.719	1.722	
0.95%	0.06%	0.37%	0.44%	0.22%	0.37%	0.50%	0.65%	0.69%	0.74%	0.51%	0.19%	0.14%	0.03%	0.24%	0.90%	
1.713	1.736	1.717	1.715	1.717	1.721	1.704	1.716	1.719	1.714	1.723	1.718	1.719	1.717	1.722	1.720	
1.720	1.723	1.720	1.719	1.718	1.716	1.721	1.713	1.715	1.719	1.719	1.718	1.719	1.720	1.717	1.718	
0.41%	0.74%	0.21%	0.21%	0.03%	0.30%	0.98%	0.20%	0.18%	0.28%	0.23%	0.02%	0.01%	0.16%	0.29%	0.13%	
1.720	1.721	1.724	1.721	1.724	1.714	1.711	1.712	1.722	1.710	1.716	1.719	1.723	1.720	1.720	1.707	
1.720	1.718	1.720	1.718	1.718	1.721	1.718	1.722	1.718	1.718	1.720	1.723	1.717	1.719	1.718	1.718	
0.00%	0.19%	0.26%	0.17%	0.33%	0.38%	0.36%	0.58%	0.23%	0.49%	0.25%	0.28%	0.39%	0.07%	0.16%	0.65%	
1.717	1.723	1.714	1.739	1.714	1.718	1.724	1.729	1.731	1.720	1.714	1.707	1.732	1.721	1.712	1.725	
1.718	1.716	1.714	1.722	1.721	1.719	1.720	1.712	1.718	1.723	1.714	1.721	1.717	1.717	1.720	1.722	
0.04%	0.40%	0.03%	0.98%	0.43%	0.07%	0.27%	0.94%	0.73%	0.15%	0.01%	0.79%	0.91%	0.26%	0.44%	0.18%	
1.728	1.719	1.724	1.718	1.715	1.731	1.718	1.716	1.727	1.722	1.729	1.722	1.712	1.716	1.721	1.716	
1.716	1.717	1.713	1.721	1.714	1.714	1.715	1.712	1.718	1.722	1.723	1.713	1.720	1.722	1.722	1.718	
0.68%	0.14%	0.65%	0.15%	0.05%	0.98%	0.17%	0.08%	0.85%	0.23%	0.39%	0.10%	0.07%	0.25%	0.07%	0.12%	
1.727	1.717	1.725	1.720	1.720	1.724	1.715	1.724	1.716	1.713	1.718	1.727	1.725	1.731	1.719	1.718	
1.717	1.717	1.715	1.720	1.714	1.715	1.724	1.720	1.724	1.716	1.713	1.720	1.714	1.720	1.720	1.720	
0.57%	0.03%	0.56%	0.01%	0.36%	0.51%	0.54%	0.20%	0.45%	0.20%	0.28%	0.39%	0.60%	0.62%	0.02%	0.11%	

Fig. 8. Ratios between maximum and average values of the pin to pin fluxes for assembly 87.

and 0.026584. With these observations, a new approach to input data for the training and validation methods was performed. Input data were divided into 16 layers. So, at this point, an array with the pairs (C, γ) was generated for each node for each layer.

Figure 6 shows the results for predicted values for the neutron fluxes in the first layer, assembly 53, which has been submitted to the training and validation algorithm. As can be seen, the maximum error obtained is 1.29%, according to equation (8). Recall that the fuel assemblies

are surrounded by the side reflector of the reactor, which according to Pessoa and Martinez [3] is the most critical part of the reconstruction.

Figure 7 shows the reconstruction of the pin to pin fluxes for assembly 53. It also shows the ratio between the maximum and the average pin to pin fluxes, with a maximum of 1.76% error.

Table 1 shows the maximum values of errors obtained, according to equation (8), the distribution of neutrons by layers of all the pins 256 of assembly 53.

Figure 8 shows the comparison between the values obtained from the ratio between maximum and average flux value of the same pin. The comparison shows a maximum error of 1.27% for the 256 pins in assembly 87, using the same learning parameters to determine the pin to pin fluxes in assembly 53.

The computational time for the search of the optimal parameters of learning, with respect to training and validation procedures is very large, about 72 h. However, with the optimal parameters defined, the algorithm provides the values of the pin to pin fluxes for each test fuel assembly in negligible time.

6 Conclusions

These results show that the machine learning technique using SVR is able to reconstruct the pin to pin fluxes. From the data of the first nuclear reactor cycle, the learning parameters can be determined. The search for optimal learning parameters takes a very large computational time, about 72 h. However, the prediction, once these parameters are found, is very quick. Pessoa and Martinez [3] found error exceeding 14% in pin to pin flux reconstruction for the assembly surrounding the baffle/reflector. For assembly 53, which surrounds the baffle/reflector, the maximum error obtained by the technique presented in this work was 1.76 %. A direct application of this work is that the machine learning using SVR can be incorporated into reactor physics codes. The SVR could work in parallel with reactor physics codes for the data in real time and determine pin to pin neutron fluxes for subsequent cycles in a nuclear reactor, that is, in fuel reload calculations. After this, the system is able to determine the safety factors, which depend on these fluxes, in a very small computational time for future cycles of a nuclear reactor power.

Author contribution statement

This paper was developed as a doctoral thesis theme, which I (A. Alvim) offered to Wanderson, to be developed under my supervision and also of Professor Fernando Carvalho da Silva, in the Reactor Physics area of the Nuclear Engineering Programme of COPPE/UFRJ. Professor Fernando and I have worked previously in flux reconstruction problems with the methods described in the review section of the paper. Professor Leandro Guimarães Marques Alvim, of UFRJ, has provided assistance to Wanderson with respect to the SVM method in relation to the flux reconstruction problem, which is one of his areas of expertise.

All the authors have contributed with analyses and critical judgement of results obtained in the course of the work. All the authors have reviewed technically the paper and I was in charge, as corresponding author, of

incorporating the referees' requirements and of the writing of the entire paragraph detailing the SVM method.

References

1. H. Finneemann, F. Bennewitz, M.R. Wagner, Interface current techniques for multidimensional reactor calculations, *Atomkernenergie* **300**, 123 (1977)
2. V.N. Vapnik, *The Nature of Statistical Learning Theory*, 2nd edn. (Springer, New York, 1999)
3. P.O. Pessoa, A.S. Martinez, Methods for reconstruction of the density distribution of nuclear power, *Ann. Nucl. Energy* **83**, 76 (2015)
4. K. Koebke, M.R. Wagner, The determination of the pin power distribution in a reactor core on the basis of coarse mesh methods, *Atomkernenergie* **30**, 136 (1977)
5. K. Koebke, L. Helzelt, On the reconstruction of local homogeneous neutron flux and current distributions of light water reactor nodal schemes, *Nucl. Sci. Eng.* **91**, 123 (1985)
6. K.R. Rempe, K.S. Smith, A.F. Henry, SIMULATE-3 pin power reconstruction: methodology and benchmarking, *Nucl. Sci. Eng.* **103**, 334 (1989)
7. V. Vapnik, A. Lerner, Pattern recognition using generalized portrait method, *Autom. Remote Control* **24**, 774 (1963)
8. V. Vapnik, A. Chervonenkis, A note on one class of perceptrons, *Autom. Remote Control* **25**, 103 (1964)
9. K.R. Müller, A. Smola, G. Rätsch, B. Schölkopf, J. Kohlmorgen, V. Vapnik, Predicting time series with support vector machine, in *Artificial Neural Networks – ICANN'97*, Springer Lecture Notes in Computer Science, edited by W. Gerstner, A. Germond, M. Hasler, J.-D. Nicoud (Springer, Berlin, 1997), Vol. 1327, pp. 999–1004
10. H. Drucker, C.J.C. Burges, L. Kaufman, A. Smola, V. Vapnik, Support vector regression machines, in *Advances in Neural Information Processing Systems 9*, edited by M. Mozer, M. Jordan, T. Petsche (MIT Press, Cambridge, MA), pp. 155–161
11. D. Matterna, S. Haykin, Support vector machine for dynamic reconstruction of a chaotic system, in *Advances in Kernel Methods – Support Vector Learning*, edited by B. Schölkopf, C.J.C. Burgues, A.J. Smola (MIT Press, Cambridge, MA, 1999), pp. 211–242
12. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: machine learning in Python, *J. Mach. Learn. Res.* **12**, 2825 (2011)
13. V. Vapnik, S. Golowich, A. Smola, Support vector method for function approximation, regression estimation, and signal processing, in *Neural Information Processing Systems*, edited by M. Mozer, M. Jordan, T. Petsche (MIT Press, Cambridge, MA, 1997), Vol. 9
14. A.J. Smola, B. Schölkopf, A Tutorial on Support Vector Regression, Royal Holloway College, NeuroColt Technical Report (NC-TR-98-030), University of London, UK, 1998