INSTITUT DE FRANCE
Académie des sciences

Comptes Rendus

Mécanique

Imane Hajjout, Manal Haddouch and El Mostapha Boudi

Optimal influence cover for an element free Galerkin MFree method based on artificial neural network

Volume 348, issue 1 (2020), p. 63-76

Published online: 30 March 2020

https://doi.org/10.5802/crmeca.5

Les Comptes Rendus. Mécanique *sont membres du Centre Mersenne pour l'édition scientifique ouverte*
www.centre-mersenne.org
e-ISSN : 1873-7234

Comptes Rendus
Mécanique
2020, 348, nº 1, p. 63-76
https://doi.org/10.5802/crmeca.5

Computational solid mechanics / *Mécanique des solides numériques*

Optimal influence cover for an element free Galerkin MFree method based on artificial neural network

Imane Hajjout *, [a], **Manal Haddouch** [a] **and El Mostapha Boudi** [a]

[a] Mohammadia School of Engineers, Turbomachinery Lab, Mohammed V-University in Rabat, Rabat BP 765, Morocco.

E-mails: hajjout.ima@gmail.com (I. Hajjout), manal.haddouch@gmail.com (M. Haddouch), elmostapha.boudi7@gmail.com (EM. Boudi).

Abstract. The present investigation presents an efficient meshless method based on the weak form of an element-free-Galerkin method. The formulation of the numerical solution was conducted using an artificial neural network (ANN) approach to compute the optimal number of nodes in the influence domain for each point of interest. The numerical results using the ANN model were tested and compared with different approaches in the literature. Results show a reduction in the computational cost and an enhancement in an error criterion of up to 11%.

Keywords. Meshless methods, Element free Galerkin, Influence domain, Artificial neural network.

Manuscript received 4th July 2019, revised 10th September 2019 and 28th November 2019, accepted 29th November 2019.

1. Introduction

Meshless or MFree methods (MMs) are considered numerical alternatives to solve the limitations of the finite element method (FEM) and the boundary element method (BEM). Meshless methods have attracted considerable attention over the past decade in solving practical engineering problems in heat transfer [1], fluid mechanics, and applied mechanics. The use of MMs as element-free Galerkin (EFG) method becomes essential in the case of mesh distortion. Due to changes in the shape of the structure, ensuring the accuracy of continually evolving deformed mesh model is difficult. Consequently, accuracy commonly degrades during the evaluation of the structural response [2]. MFree Methods generate smooth strain and stress fields. These methods can treat important deformation fracture [3] and interface problems with more precision than the FEM [4].

The first MM is the smoothed particle hydrodynamics (SPH) method with application in astrophysical phenomena [5]. After the diffuse element method (DEM) of Nayroles [6], several

* Corresponding author.

other categories of MMs have appeared, including the EFG method [4], the meshfree reproducing kernel particle method (RKPM) [7], the meshless local petrov Galerkin (MLPG) [8], the local radial point interpolation method (LRPIM) [9], and a point interpolation method (PIM) produced for stress analysis for two-dimensional solids [10].

The EFG method developed by Belystchko [11] is the standard numerical method for solving several types of partial differential equations (PDEs). It has been used for multiple practical engineering problems in heat transfer [1], crack propagation [12], elastoplastic contact problems [13], boundary problems [14], fluid mechanics, and applied mechanics. The EFG method uses local weak forms over a local sub-domain and the shape functions are constructed via the moving least-squares (MLS) approximation [15]. Moreover, in this approximation, each node in the domain has compact support called the influence domain. A weighting function is employed to determine the intensity of a node's effect at different points in its cover. The shape of the local supports can be arbitrary, such as circle, square, or rectangle (conventional domains) in 2D geometries [16] and circular, square, or rectangular parallelepipeds in 3D geometries.

The size of the influence domain is crucial for the precision, stability, and computation cost of numerical solution in EFG method, wich defines the degree of continuity between nodes and the size of the matrix system. In the literature, different methods are available to calculate the size of the influence domain (for the 2D problem) on the basis of tests and experiences. However, evaluating the results is necessary to conclude the appropriate size. Belytschko defines the size of the influence domain as the number of neighboring nodes at the node of interest, allowing a regular moment matrix [11]. Furthermore, Dolbow and Belytschko propose scale parameter c to compute the size of the influence domain for static analyses. This constant c is assigned between 2 and 4 and multiplied by the distance necessary for a conventional system [17]. Zhuang et al. present the parameters that influence the numerical result, especially the discretization error, and highlight the nodes contained in the support domain without giving a precise solution [18]. Liu and Tu assign a density or scale factor to characterize each point and calculate the radius of the influence domain by multiplying this factor by constant c [19]. Zhang et al. use the arbitrary convex polygon shape of the influence domain with their simplified approach on the imposition and implementation of boundary conditions [20]. Therefore, the boundaries are more precise than the standard form of the cover, the computation time decreases, and the EFG numerical result is better with the usual shape of the influence domain. Conversely, Cai et al. [21] employ an independent cover meshless method (ICMM) with a polynomial approximation for 2D elastic solids and crack propagation. The authors used the Delaunay triangulation to construct a unique and independent influence cover. However, Sheng et al. [22] recommend to fix the number of nodes in the cover in all the structures and then calculate the radius of the influence domain for each point of interest. Furthermore, the strategy of the authors minimizes the computation time and improves the result unlike other computation techniques.

A growing concern for this factor emerges (size of the influence domain) in the refinement step. Naturally, finding the optimal number of nodes in the cover is crucial to minimize the computation time for the added nodes, which requires several tests and experiments, as discussed by Rossi [23]. A crucial issue is that a large influence cover demands for a high computation time and the MLS method loses its local character. However, a small number of nodes in the cover causes a singular moment matrix.

In the present investigation, the EFG meshless method is used to investigate the number of nodes in the influence domain. Due to the noncompliance of the Kronecker delta property by the MLS shape functions [24], the Lagrange multipliers are used to enforce the essential boundary conditions. An ANN model is proposed and gives the appropriate number of nodes in the influence domain for each interest point with a minimal global error. This approach considers the node discretization, the order of basis function, and the number of Gauss quadrature.

Table 1. Approaches computing the size of the influence domain (dm)

Authors	Years	Doi	Designation
[11]	1998	dm	The size of the influence domain must be sufficient for a regular system.
[17]	1998	$dm = d_{\max} C_i$	$d_{\max}$: the scaling parameters; C_i: the nodal spacing between the node of interest ($\tilde{x}$) and the nearest node (x_i).
[19]	2002	$r_I = cS_I$	Each node has nodal density $S_i = \sqrt{2/m \sum_{i=1\ldots m} a_i}$; C is a constant.
[26]	2005	$d_s = \alpha_s d_c$	d_c distance between two neighbor nodes for a uniform distribution; $d_c = (\sqrt{A_s}/\sqrt{n_{A_s}}) - 1$ for a non-uniformly distributed node; A_s: area of the estimated support domain; n_{A_s}: the number of nodes covered by the estimated domain with the area of A_s.
[22]	2015	dm	The number of nodes contained in all supports is constant.

The rest of this paper is organized as follows: Section 2 describes the MLS approximation; Section 3 discusses the proposed method with an ANN; Section 4 presents the application of the ANN model; Section 5 presents the obtained results; and Section 6 concludes the paper.

2. Moving least square approximation

For the computation of the shape function and the approximation of displacement $u^h(x)$, the MLS method is used [15, 25] and written as follows:

$$u^h_{\tilde{x}} = p^T(x)a(x) = \sum_{i=1}^{N} \phi_i(x) u_i, \tag{1}$$

$$\phi_i(x) = c^T(x) p(x_i) w_i(x), \tag{2}$$

$$c(x) = A^{-1}(x) p(x), \tag{3}$$

$$A(x) = \sum_{i=1}^{\hat{n}} w_i(x) p(x_i) p^T(x_i), \tag{4}$$

where $p(x)$ is the basis function and $w_i(x)$ is the weight function centered at x and a is a known coefficient. Here, $\phi_i(x)$ is the shape function defined at particle x, $A(x)$ is the moment matrix, and N is the number of nodes contained in the influence domain.

In this article, the 3th-order spline weight function is used for all the examples and is written as follows:

$$f_3(r) = \begin{cases} \dfrac{2}{3} - 4r^2 + 4r^3 & \text{if } |r| \le \dfrac{1}{2}, \\ \dfrac{4}{3} - 4r + 4r^2 - \dfrac{4}{3} r^3 & \text{if } \dfrac{1}{2} < |r| \le 1, \\ 0 & \text{if } |r| > 1, \end{cases}$$

where $r = ||\tilde{x} - x||/dm$ and dm is the size of the influence domain surrounding the node i. Table 1 describes some approaches to calculate the size of the influence domain.

In the present study, the error between the EFG numerical solution method and the exact solution is written as follows:

$$\sigma^e(x) = \sigma(x) - \sigma^{\text{num}}(x). \tag{5}$$

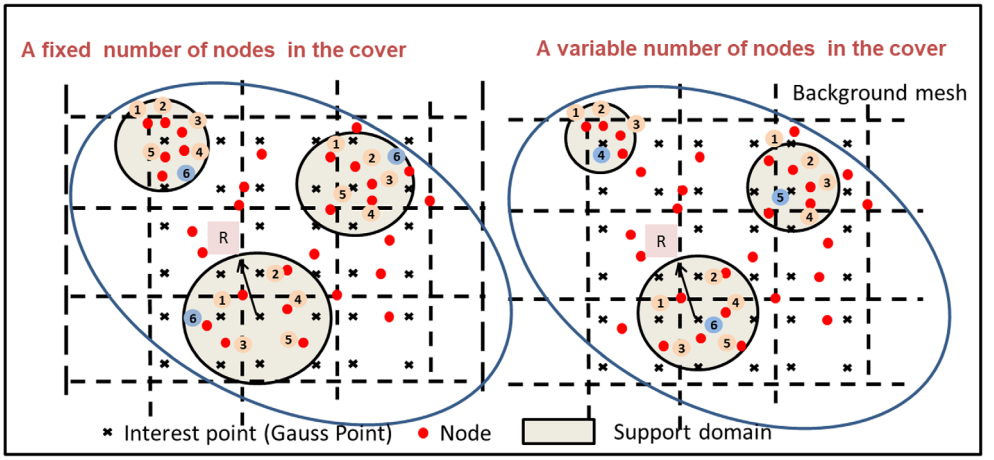

Figure 1. Approaches to compute the size of the influence domain.

Here, $\sigma(x)$ is the exact stress, and $\sigma^{\text{num}}(x)$ is the EFG numerical stress. The exact local error is calculated per cell or Gauss point [27]. The energy norm for the EFG method $\|e_e\|_E$:

$$\|e_\sigma\|_E = \left(\int_{\Omega_e} \sigma^e(x)^T D^{-1} \sigma^e(x) d\Omega\right)^{1/2}, \tag{6}$$

for the local error at a Gauss point:

$$\left\|e_\sigma(x_g)\right\| = \left(\frac{1}{2}\sigma(x_g)^T D^{-1} \sigma(x_g)\right)^{1/2}. \tag{7}$$

3. Optimization of the number of nodes in the cover

3.1. *Problem description*

As outlined in the introduction section, two approaches exist in the literature to compute the size of the influence domain. In the first one, a variable number of nodes is in the influence domain [11, 17, 26], and in the second one is a fixed number of nodes for each point of interest in the entire domain [22]. Figure 1 illustrates that the two approaches have a variable size of the influence domain. Each strategy has its own advantages and disadvantages. The objective of this work is to determine the technique for calculating the influence domain that allows a minimum global error and an optimal CPU. For this purpose, a study is carried out on a linear elastic problem with a circular hole plate (see paragraph 3.1.1). The number of nodes contained in all covers is initially fixed in the given meshing, and then the global and local error energy norm is calculated. This process is made for multiple fixed nodes in the range of 5–16 (see Section 3.2). Furthermore, for the interpretation of the obtained results, we calculate the minimum global error $\left\|e_\sigma(x_g)\right\|$ obtained this time from the local minimum errors of each Gauss point for the 12 scenarios (fixed number of nodes in the influence domain 5–16). Table 2 presents an example of the results obtained for 203 nodes, the minimum global energy norm $\left\|e_\sigma^{\text{Min}}\right\|_E = 0{,}00753011$ enhances the result obtained with 11 nodes fixed in the cover by $\|e_\sigma\|_E = 0{,}00819991$ which is 8.2%. Thus, a variable number of nodes in the cover improves the numerical result comparing to a fixed number of nodes. Moreover, Table 3 exhibits that the minimum global error $\left\|e_\sigma^{\text{Min}}\right\|_E$ constantly enhances the result for several mesh steps.

Table 2. Computation of the minimum global error for a variable size of the influence cover

N° of nodes in the cover	Local error for the integration Gauss point for the whole problem									Global error in energy norm
	$\|e_\sigma(X_{g1})\|$	$\|e_\sigma(X_{g2})\|$	$\|e_\sigma(X_{g3})\|$	$\|e_\sigma(X_{g4})\|$	$\|e_\sigma(X_{g5})\|$	$\|e_\sigma(X_{g6})\|$	$\|e_\sigma(X_{g7})\|$	…	$\|e_\sigma(X_{gn})\|$	$\|e_\sigma\|_E$
5	1,21E-07	1,17E-07	1,29E-07	1,23E-07	9,58E-08	1,08E-07	2,71E-08	…	1,64E-08	0,01064767
6	1,11E-07	1,39E-07	1,34E-07	9,83E-08	9,01E-08	1,08E-07	2,87E-08	…	1,58E-08	0,00941966
7	1,11E-07	1,50E-07	1,54E-07	7,98E-08	8,67E-08	1,13E-07	2,78E-08	…	1,50E-08	0,00890441
8	1,06E-07	1,55E-07	1,59E-07	7,48E-08	8,30E-08	1,14E-07	2,58E-08	…	1,56E-08	0,00849299
9	1,10E-07	1,66E-07	1,59E-07	7,09E-08	8,40E-08	1,17E-07	2,23E-08	…	1,79E-08	0,00825805
10	1,12E-07	1,71E-07	1,59E-07	7,04E-08	8,47E-08	1,17E-07	2,08E-08	…	1,85E-08	0,00820654
11	1,12E-07	1,76E-07	1,59E-07	6,99E-08	8,55E-08	1,16E-07	1,66E-08	…	1,91E-08	0,00819991
12	1,14E-07	1,79E-07	1,57E-07	7,01E-08	8,71E-08	1,15E-07	1,57E-08	…	1,91E-08	0,00825908
13	1,16E-07	1,79E-07	1,56E-07	7,03E-08	8,76E-08	1,13E-07	1,31E-08	…	1,90E-08	0,00828553
14	1,19E-07	1,80E-07	1,53E-07	7,15E-08	8,78E-08	1,13E-07	1,21E-08	…	1,84E-08	0,00833634
15	1,19E-07	1,81E-07	1,43E-07	7,35E-08	8,79E-08	1,10E-07	9,84E-09	…	1,84E-08	0,0084239
16	1,20E-07	1,81E-07	1,40E-07	7,80E-08	8,87E-08	1,10E-07	8,39E-09	…	1,83E-08	0,00850297
Min $\|e_\sigma(x_g)\|$ = 0,00753011	1,06E-07	1,17E-07	1,29E-07	6,99E-08	8,30E-08	1,08E-07	8,39E-09	…	1,50E-08	$=\sum_{i=1:n}\sqrt{\text{Min}\|e_\sigma(x_g)\|}$
Goal	8	5	5	11	8	5	16	…	7	

Table 3. Observations

Meshing step	007	008	009	010	011	012	013	014	015	018	020
$\|e_\sigma\|_E$	0,008255	0,008239	0,008239	0,008222	0,008146	0,008156	0,008025	0,008232	0,00830	0,008199	0,008842
$\|e_\sigma^{\min}\|_E$	0,007838	0,007794	0,007737	0,007706	0,007613	0,007592	0,007435	0,007575	0,007584	0,007530	0,007937
the gain in %	5,05%	5,41%	6,10%	6,28%	6,55%	6,92%	7,35%	7,98%	8,72%	8,17%	10,23%

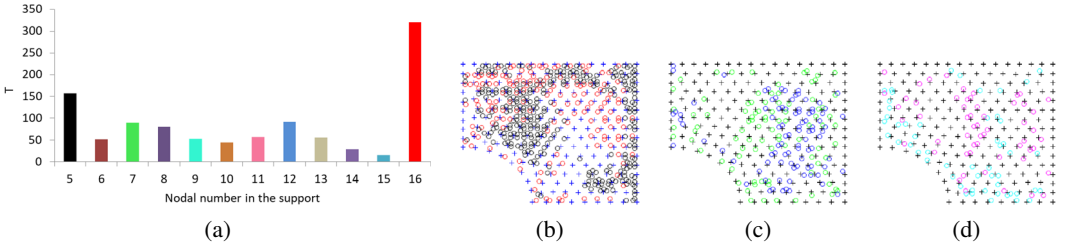

Figure 2. Position of the optimal number of nodes in the cover at the Gauss point, (a) [5;16], (b) [7;12], (c) [9;11].

Figure 2 displays the distribution of the optimal number of nodes in the cover for each Gauss point. Unfortunately, the arrangement of the nodes in the cover is arbitrary and non-linear, which means that some regions need more nodes than others for approximation. Figure 2 also depicts that the choice of the cover size for each area is not evident; specifically, the number of nodes in the cover varies between 5 and 16 (T is the sum of the Gauss point number for r_i = 5:16 where the local error is minimal). Consequently, an ANN model is used to define the number of nodes in the cover, allowing a minimal local error (see Table 2). The ANN model is used for classification. The classes are the outputs of the ANN model, which represent the number of nodes in the influence domain, and the point of interest against to each class.

3.1.1. *Plate with a circular hole*

A plate with a circular hole subjected at its extremities to a simple state of tension (σ = 1 N/m) is studied. Figure 3 illustrates the model and its quarter (symmetry problems), where L = 5 m, a = 1 m, $E = 10^3$ MPa, and ν = 0.3. The shape of the influence domain is circular in this example,

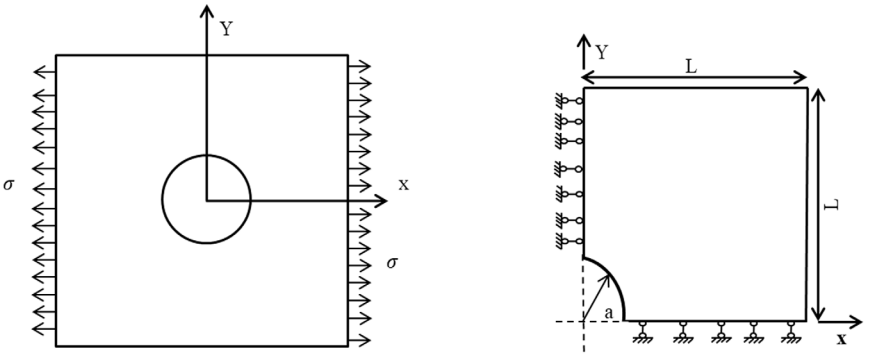

Figure 3. Plate with a circular hole and the quadrant studied.

and the Delaunay algorithm is used to generate the coordinate of the Gauss point and the node field. The problem is solved in 2D with the plane stress hypothesis. Timoshenko and Goodier give the analytical solutions [28].

3.2. *Determination of the number of nodes in the cover with an ANN*

3.2.1. *Artificial neural network*

ANN is a computation process, developed by neurologist Frank Rosenblatt in 1950, and became a dominant algorithm in 2012. The artificial neural network also refers to the biological neurons and is inspired by the human brain [29]. This system is based on learning by training and then generating results and is used in various fields like facial recognition domains and image processing. The ANN process has several models: multi-layer perceptron (MLP), adaptive neuro-fuzzy inference system (ANFIS) [30], Support Vector Machine (SVM) [31]. Moreover, ANN is generally composed of three types of layers: input, hidden, and output. The principle of this structure is that each layer is composed of several neurons (N_i), and their input comes from the previous layer. An activation function is required for the hidden and output layers. Each neuron has a weight value, which is estimated in the training step and constitutes the memory of the network. In this study, the MLP with feed-forward neural network training by the back-propagation (BP) algorithm is used. In the MLP model, a hidden layer neuron is connected at the input to each neuron of the next layer and the output to each neuron of the previous layer. The principle of the BP model consists of transforming input data forward by a non-linear activation function toward the output layer and then back-propagating the prediction error to adjust the weighting used [32]. The MLP realizes a transformation forward from the i^{th} layer to the j^{th} layer given by the following:

$$y_j^m(n) = f\left(\left(\sum_{k=1}^{p} w_{ij}^k(n)\, y_i^k(n)\right) + b^k\right), \tag{8}$$

where $y_j^m(n)$ is the output value of the m^{th} neuron of the j^{th} layer, $f(x)$ is the activation function, p is the total number of nodes in the cover of the i^{th} layer and $w_{ij}^k(n)$ and b^k are the weight and the bias from the k^{th} node of the i^{th} layer at the n^{th} iteration respectively. Figure 4 illustrates the architecture of the four-layer BP model.

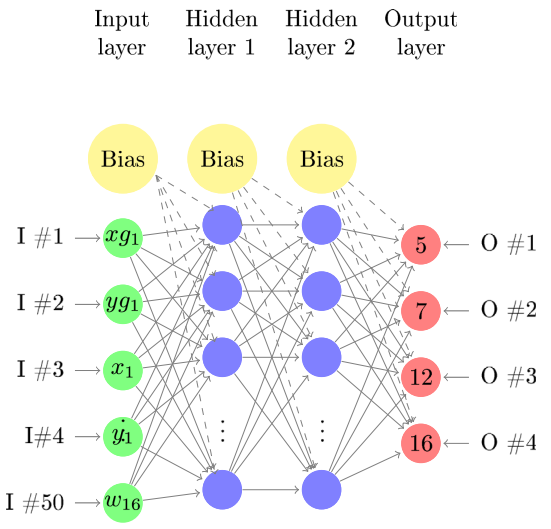

Figure 4. MLP architecture.

3.2.2. *Data preparation*

The literature has confirmed that scale parameters are required to limit the number of nodes in the support. Therefore, an interval must be fixed beforehand to learn the ANN model. The lower limit is the number of nodes that allows a regular moment matrix. However, a study is performed to define the higher limit. The present study confirms that even a sizeable number of nodes in the support in some regions gives a minimum error. However, the generated computational time is more critical. Thus, a parameter (the profit) is used (see (9)) to determine the maximum number of nodes in the support. The technique employed is to fix the number of nodes $(r_i, i = 5, 6 \ldots, 20)$ in the support for all Gauss points and is performed 15 times for the number of nodes from 5 to 20. In each step, the profit is calculated which is the percentage between the global minimum error energy norm in the range 5–20 and the global error energy norm calculated by the number of nodes r_i in all the supports of the domain, which is presented as follows:

$$Profit(\%) = \frac{\left\| e_\sigma^{\text{Min}} \right\|_{[5:20]} - \| e_\sigma \|_{[r_i]}}{\left\| e_\sigma^{\text{Min}} \right\|_{[5:20]}}. \tag{9}$$

The objective for fixing the upper limit of the interval is to maximize the profit. Figure 5 shows the average of the profit variation between r_i and r_{i+1} at different mesh sizes. The curve decreases slightly between 16 and 20 points, and their profits vary between 9% and 9.6% respectively. we conclude from that 16 is the upper limit of the nodes number in the influence domain. The difference in profit at this upper limit is small, but the computational cost is more optimal compared with 20 points. This number (16) respects the size range of the influence domain which is described by Doblow and Belystchko [17] and Sheng [22].

The input data of the MLP model (see paragraph 1.) are: X_g the coordinates of Gauss points (support center), X_i the coordinates of the 16 points surrounded X_g, and the weights assigned to each point $W_i(X_g)$. Therefore, the output data for the optimization problem are the number of nodes in the influence domain $\{5, 7, 12, 16\}$ (see paragraph 2.). After the selection and definition of the inputs and outputs of the MLP model, files are generated (1 file per mesh step), and the input vector with the optimal class is attributed in the following form: $[xg_s; yg_s; x_1; y_1; x_2; y_2; y_2; \ldots; x_{16}; w_1; w_2; \ldots w_{16}, C_s]$, Table 6 summarizes the content of these files.

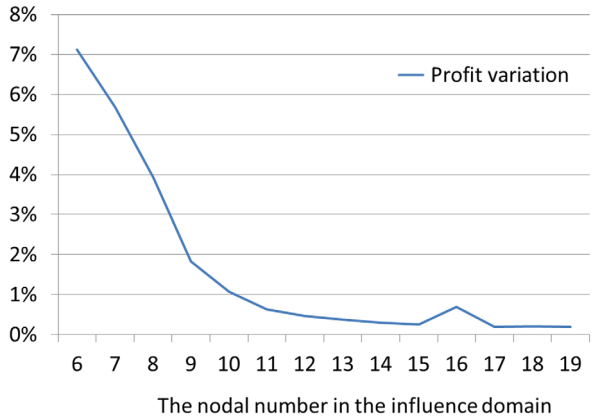

Figure 5. Profit variation versus number of nodes in the support.

1. Input data selection:

   The input vector plays a crucial role in the learning process of a classification problem in the optimization algorithm. The model converges with a high accuracy rate with high vectors of information. In the training process, several input vectors must be used for each class. Therefore, the mesh step of the Delaunay algorithm has been varied to have several input vectors $\{X_g; X_i; W_i(X_g)\}$.
2. Output data selection:

   The class number plays a critical role in the accuracy and complexity of the classification problem; several classes make a complex architecture and increase the converging process. For that purpose, a study is done to minimize this number without impacting the result.

   The output choice is defined according to several tests. Table 5 reveals the selection process for the output data. As mentioned earlier, the maximum number of nodes in the influence cover is 16. Initially, the output size of 12 classes 5–16 is examined. The number of nodes in the cover with the lowest impact on goal profit for each given mesh size is determined. Subsequently, it is removed from the output list and so on until two classes, as shown in Table 5. By deleting the outputs one by one; the final results are minimally influenced, in the meantime, convergence, and precision of the system are much better. However, four classes are selected when the impact on profit is yielded by 1%.

3.2.3. *ANN configuration*

Several algorithm optimizations are available, and the Adam algorithm is used in this study [33]. Moreover, the evaluation is made by the mean square error (MSE). The MSE is defined as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\widehat{X_i} - X_i)^2 = \frac{1}{n}\sum_{i=1}^{n}(e_i)^2. \tag{10}$$

Here, n, $\widehat{X_i}$, and X_i, present the number of input data, numerical result, and the calculated value respectively from ANN model. The data vector (X_g, X_i, W_i) is defined for several mesh sizes and used as an input to the MLP system. These data sets are divided into three parts: training, validation, and testing. Training routine is an essential learning step for the model to determine the appropriate weight functions. In this step input and output data are used. Therefore, 70% of the data are employed for the training and validation step and 30% for the testing. Figure 6

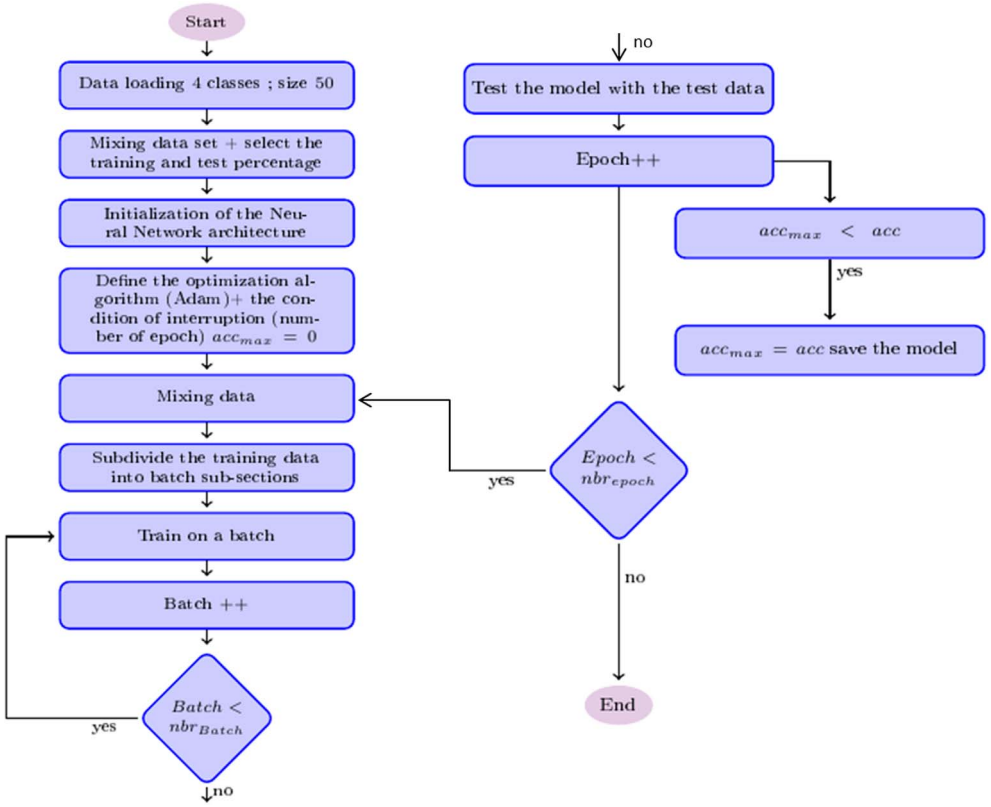

Figure 6. Flowchart optimization process with ANN.

Table 4. Some MLP model scenarios

$N°$	Epoches	Activity function	Architecture	Convergence	Test	Mean improvement
1	5000	*Relu*	2000	89%	73%	3,08%
2	30000	*Sigmoid*	2000	87%	72%	2,67%
3	10000	*Relu*	3000	92%	72%	3,07%
4	30000	*Sigmoid*	150|150	94%	77%	2,94%
5	10000	*Tanh*	150| 150	96%	77%	3,06%
6	5000	*Relu*	350| 150	99%	76%	2,80%
7	5000	*Tanh*	150| 150	90%	76%	2,81%
8	5000	*Relu*	250|150|150	99%	77%	2,88%
9	5000	*Relu*	350| 150|50	99%	78%	2,17%
10	5000	*Relu*	150| 50|50	92%	74%	2,38%

presents the flowchart of the optimization algorithm process. The program is implemented with the Python software.

3.2.4. *ANN construction*

The hidden layer number, the neuron number, and the activation functions are changed to produce the perfect ANN model with maximum accuracy, and given in the flowchart 6. Consequently, several scenarios are performed to provide the best model. Table 4 shows some

Table 5. Output data analysis

	goal profit			
	−11,167%	−8,895%	−9,543%	−7,440%
	Step meshing			
N° of classes	0.20	0.18	0.15	0.12
12	7,94E-03	7,53E-03	7,58E-03	7,59E-03
11	0,001%	0,002%	0,004%	0,001%
10	0,023%	0,018%	0,010%	0,005%
9	0,088%	0,055%	0,039%	0,020%
8	0,235%	0,147%	0,499%	0,076%
7	0,342%	0,246%	0,574%	0,140%
6	0,639%	0,308%	0,647%	0,206%
5	0,661%	0,337%	0,669%	0,221%
4	**1,029%**	**1,269%**	**0,885%**	**0,412%**
3	2,986%	2,817%	1,949%	1,336%
2	7,896%	7,389%	7,293%	5,812%

Table 6. Constructed files and classes for the MLP model

		Vector number / Class			
Mesh step	Total vector number	$C_1(5)$	$C_2(7)$	$C_3(12)$	$C_4(16)$
0.07	7257	1573	845	1625	1589
0.08	5631	1541	871	1621	1598
0.09	4422	1177	713	1236	1296
0.1	3576	923	570	984	1099
0.11	2828	688	462	719	959
0.12	2436	577	412	636	811
0.13	1991	457	354	465	715
0.14	1790	385	287	451	667
0.15	1497	296	246	391	564
0.18	1047	181	192	273	401
0.2	825	135	121	233	336

Table 7. Dimension of the plate with an elliptic hole considered for the study

Length, height	$L = 70$ mm, $D = 50$ mm
Crack	$a = 20$ mm, $b = 5$ mm
Poisson's ratio	$\nu = 0.3$
Young's modulus	$E = 2e11$ Pa
Loading	$P = 100$ N

studied examples and that the convergence training reaches 99% and that the test process achieves an outstanding value of 78%.

Approximately thirty tests have been conducted, and the selected model contains one hidden layer with 2000 neurons. The relu activation function is used for the hidden layer, whereas softmax is used for the output layer of all scenarios. Figure 9 presents the software interface with the input, output data, and the configuration of the model (optimization algorithm, % of

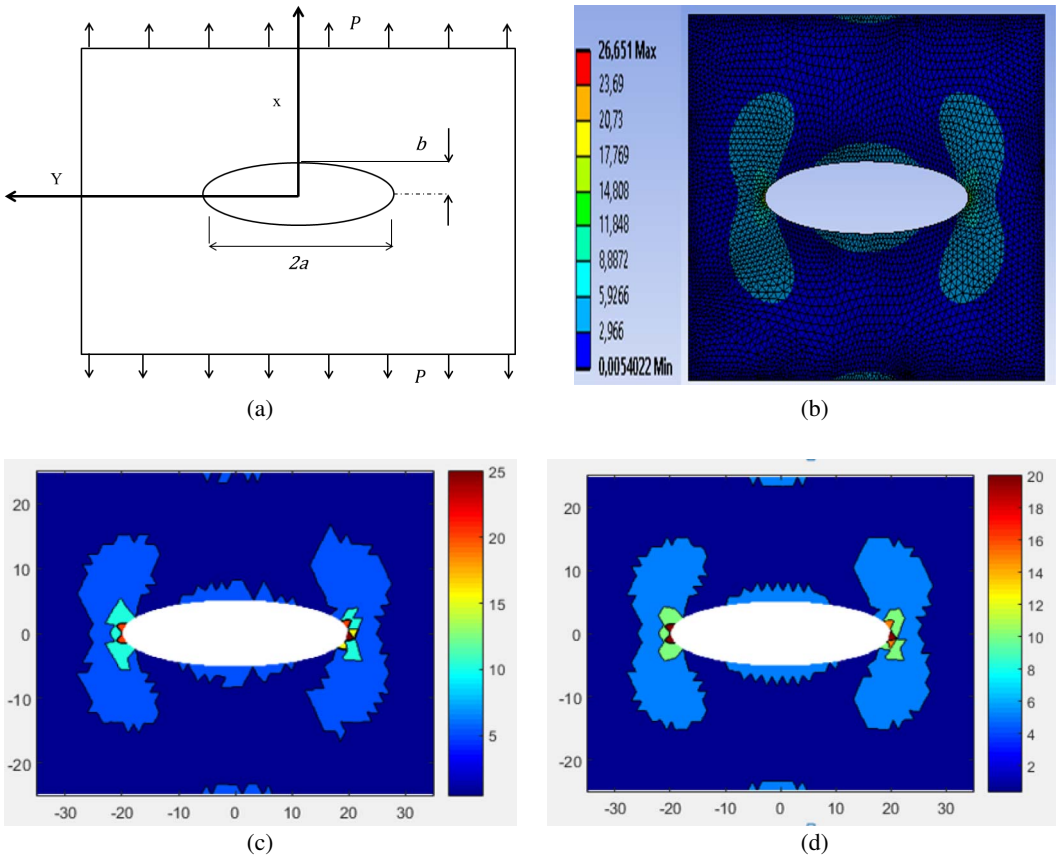

Figure 7. VM stresses for the plate with an elliptic hole. (a) Elastic plate with an elliptic hole. (b) $\sigma_{\text{VM}}^{\text{FEM}}$. (c) The use of ANN model in the approximation. (d) Fixed node in the influence domain (8 nodes).

training). The selected architecture to build the model in the data network and train rubrics are also presented in the same figure.

4. Application

An elastic square plate similar to the previous example is always present in the geometries of complex mechanical parts. In the present case, the elliptical crack is studied to validate the developed ANN model. It is subjected to uniform stress at two different ends and is depicted in Figure 7a.

The FEM simulation is performed by ANSYS, the results are shown in Figure 7b, and the ANN model is used to test its performance. Figure 7b illustrates that the maximum VM stresses reaches 26 MPa. The closest result is shown in Figure 7c, which uses the ANN model in the solution approximation.

5. Results and discussion

Several MLP architectures were tested, and the best scenarios with the minimum MSE are mentioned in Table 4. Figure 8 depicts the global error in the energy norm against the mesh

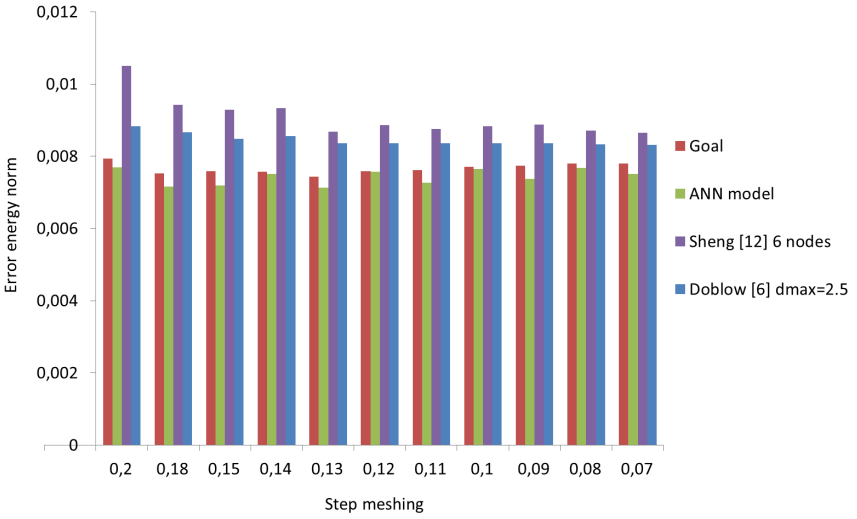

Figure 8. MLP model versus other approaches.

Table 8. Computing time for the formulation of the EFG solution with the ANN model

		Time (s)		
Gauss point	Node	ANN	[22]	[11]
1047	203	3.97	2.81	11.59
Memory (Mo)		520	517	523

step used in the MLP model and the desired goal (minimum global error). The results of the developed MLP model are identical to the goal (as shown in Figure 8) that signifies the efficiency, precision, and generics of the ANN model. Moreover, in comparison with other approaches, the MLP model is given the number of nodes directly in the cover for each interest point. Thus, the only computation needed is to select the number of nodes from the interest point. The performance of the ANN approach in the CPU is shown in Table 8, with the application on the plate with a circular hole. The example is run on the following hardware: Core i5-2430M CPU 2.4 GHz and 8 Go RAM, within MATLAB R2015 a in x64 Windows 10.

6. Conclusion

The present research mainly aims to improve the EFG meshless method through an interest in the influence domain. Most of the existing methods in the literature use equations to compute the size of the influence domain, but the proposed method transforms a calculation problem into a classification problem, and each point of interest is assigned to a class. Four different classes at 5, 7, 12, and 16, are represented by the number of nodes in the influence domain, and a practical approach is developed to determine the size of the influence domain on the basis of an optimal number of nodes in the cover. Thus, an ANN system is trained to optimize between the accuracy and the computation cost of the numerical results. The ANN model is provided a matrix system witch assigns to each interest point an optimal number of node in the influence domain. The method is applied to a 2D elastic problem (plates with circular and elliptical holes) and is founded that the results are accurate compared to other existing methods. The performance in the formulation of the EFG numerical solution in terms of CPU in the proposed method is

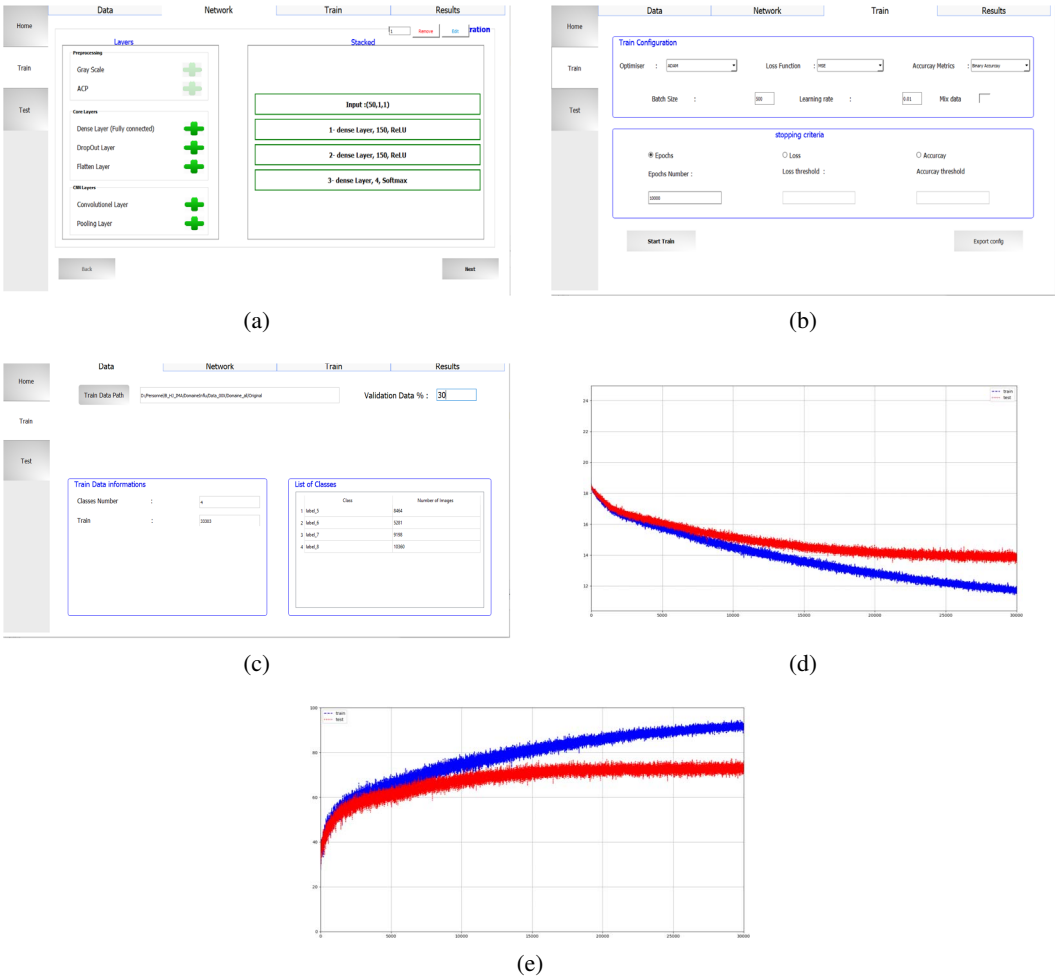

Figure 9. Python software interfaces for the MLP model. (a) Network construction. (b) Training data configuration. (c) Input and output configurations. (d) MSE. (e) Model accuracy.

slightly above the approach by fixing the number of nodes in all supports and also better than Belystchko's approach. In terms of accuracy, the ANN model is more precise as it is trained by the size of the influence domain, which results in a minimum error. The system is trained by more than thirty-three thousand (33,000) interest points, which makes it generic for the application on different meshing sizes and geometries.

References

[1] T. Mnasri, R. B. Younès, A. Mazioud, J. F. Durastanti, "FVM-BEM method based on the Green's function theory for the heat transfer problem in buried co-axial exchanger", *C. R. Mécanique* **338** (2010), no. 4, p. 220-229.

[2] B. Ullah, "Structural topology optimisation based on the Boundary Element and Level Set methods", Thesis, Durham University, May 2014.

[3] P. Profizi, A. Combescure, K. Ogawa, "SPH modeling of adhesion in fast dynamics: Application to the Cold Spray process", *C. R. Méc.* **344** (2016), no. 4, p. 211-224.

[4] Y. Lu, T. Belytschko, L. Gu, "A new implementation of the element free Galerkin method", *Comput. Methods Appl. Mech. Eng.* **113** (1994), no. 3–4, p. 397-414.

[5] L. B. Lucy, "A numerical approach to the testing of the fission hypothesis", *Astron. J.* **82** (1977), p. 1013-1024.
[6] B. Nayroles, G. Touzot, P. Villon, "Generalizing the finite element method: diffuse approximation and diffuse elements", *Comput. Mech.* **10** (1992), no. 5, p. 307-318.
[7] W. K. Liu, S. Jun, Y. F. Zhang, "Reproducing kernel particle methods", *Int. J. Numer. Methods Fluids* **20** (1995), no. 8-9, p. 1081-1106.
[8] S. N. Atluri, H.-G. Kim, J. Y. Cho, "A critical assessment of the truly Meshless Local Petrov-Galerkin (MLPG), and Local Boundary Integral Equation (LBIE) methods", *Comput. Mech.* **24** (1999), no. 5, p. 348-372.
[9] G. Liu, Y. Gu, "A local radial point interpolation method (LRPIM) for free vibration analyses of 2-D solids", *J. Sound Vib.* **246** (2001), no. 1, p. 29-46.
[10] G.-R. Liu, Y. Gu, "A point interpolation method for two-dimensional solids", *Int. J. Numer. Methods Eng.* **50** (2001), no. 4, p. 937-951.
[11] T. Belytschko, Y. Y. Lu, L. Gu, "Element-free Galerkin methods", *Int. J. Numer. Methods Eng.* **37** (1994), no. 2, p. 229-256.
[12] S.-H. Lee, Y.-C. Yoon, "An improved crack analysis technique by element-free Galerkin method with auxiliary supports", *Int. J. Numer. Methods Eng.* **56** (2003), no. 9, p. 1291-1314.
[13] Y. Belaasilia, A. Timesli, B. Braikat, M. Jamal, "A numerical mesh-free model for elasto-plastic contact problems", *Eng. Anal. Bound. Elem.* **82** (2017), p. 68-78.
[14] W. Khan, S. Ul-Islam, B. Ullah, "Analysis of meshless weak and strong formulations for boundary value problems", *Eng. Anal. Bound. Elem.* **80** (2017), p. 1-17.
[15] P. Lancaster, K. Salkauskas, "Surface generated by moving least squares methods", *Math. Comput.* **37** (1981), no. 155, p. 141-158.
[16] I. Hajjout, M. Haddouch, E. M. Boudi, "An optimal radius of influence domain in element-free Galerkin method", in *2018 6th International Renewable and Sustainable Energy Conference (IRSEC)* (A. ElHibaoui, M. Essaaidi, Y. Zaz, eds.), IEEE, 345 E 47TH ST, New York, NY 10017 USA, 2018, p. 911-914.
[17] J. Dolbow, T. Belytschko, "An introduction to programming the meshless element free Galerkin method", *Arch. Comput. Methods Eng.* **207-241** (1998), no. 5, p. 207-241.
[18] X. Zhuang, C. Heaney, C. Augarde, "On error control in the element-free Galerkin method", *Eng. Anal. Bound. Elem.* **36** (2012), no. 3, p. 351-360.
[19] G. Liu, Z. Tu, "An adaptive procedure based on background cells for meshless methods", *Comput. Methods Appl. Mech. Eng.* **191** (2002), no. 17-18, p. 1923-1943.
[20] X. Zhang, P. Zhang, L. Zhang, "A simple technique to improve computational efficiency of meshless methods", *Procedia Eng.* **31** (2012), p. 1102-1107.
[21] Y. Cai, H. Zhu, "Independent cover meshless method using a polynomial approximation", *Int. J. Fract.* **203** (2017), no. 1-2, p. 63-80.
[22] M. Sheng, G. Li, S. Shah, "A modified method to determine the radius of influence domain in element-free Galerkin method", *Proc. Inst. Mech. Eng. C* **229** (2015), no. 5, p. 795-805.
[23] R. Rossi, M. K. Alves, "An h-adaptive modified element-free Galerkin method", *Eur. J. Mech. - A* **24** (2005), no. 5, p. 782-799.
[24] S. Fernández-Méndez, A. Huerta, "Imposing essential boundary conditions in mesh-free methods", *Comput. Methods Appl. Mech. Eng.* **193** (2004), no. 12-14, p. 1257-1275.
[25] W. Khan, S. Ul-Islam, B. Ullah, "Structural optimization based on meshless element free Galerkin and level set methods", *Comput. Methods Appl. Mech. Eng.* **344** (2019), p. 144-163.
[26] L. G.R, G. Y.T, *An Introduction to Meshfree Methods and Their Programming*, Springer-Verlag, Berlin/Heidelberg, 2005.
[27] H.-J. Chung, T. Belytschko, "An error estimate in the EFG method", *Comput. Mech.* **21** (1998), no. 2, p. 91-100.
[28] S. Timoshenko, J. N. Goodier, *Theory of Elasticity*, Engineering Societies Monographs, 1951.
[29] D. D. Cox, T. Dean, "Neural networks and neuroscience-inspired computer vision", *Curr. Biol.* **24** (2014), no. 18, p. R921-R929.
[30] J. S. R. Jang, "ANFIS: adaptive-network-based fuzzy inference system", *IEEE Trans. Syst. Man Cybern.* **23** (1993), no. 3, p. 665-685.
[31] V. Vapnik, *The Nature of Statistical Learning Theory*, 2nd ed., Springer-Verlag, New York, 2000.
[32] H. You, Z. Ma, Y. Tang, Y. Wang, J. Yan, M. Ni, K. Cen, Q. Huang, "Comparison of ANN (MLP), ANFIS, SVM, and RF models for the online classification of heating value of burning municipal solid waste in circulating fluidized bed incinerators", *Waste Manag.* **68** (2017), p. 186-197.
[33] D. P. Kingma, J. Ba, Adam: a method for stochastic optimization, arXiv:1412.6980 [cs] (2014).