TECHNICAL TRANSACTIONS 7/2017
CZASOPISMO TECHNICZNE 7/2017
MECHANICS
DOI: 10.4467/2353737XCT.17.115.6656

Artur Krowiak (krowiak@mech.pk.edu.pl)
Institute of Computer Science, Faculty of Mechanical Engineering, Cracow University of Technology

Jordan Podgórski
Student of Applied Informatics, Faculty of Mechanical Engineering, Cracow University of Technology

# The adaptation of the cross validation aproach for RBF-based collocation methods

## Adaptacja podejscia krzyżowego sprawdzania do metod bazujących na funkcjach RBF

**Abstract**

The paper shows the adaptation of the cross validation approach, known from interpolation problems, for estimating the value of a shape parameter for radial basis functions. The latter are involved in two collocation techniques used on an unstructured grid to find approximate solution of differential equations. To obtain accurate results, the shape parameter should be chosen as a result of a trade-off between accuracy and conditioning of the system. The cross validation approach called "leave one out" takes these aspects into consideration. The numerical examples that summarize the investigations show the usefulness of the approach.

**Keywords:** radial basis functions, shape parameter, cross-validation approach

**Streszczenie**

W artykule pokazano adaptację algorytmu krzyżowego sprawdzania, znanego z zagadnień statystyki i interpolacji, do wyznaczenia wartości współczynnika kształtu w radialnych funkcjach bazowych. Funkcje te użyto w dwóch typach technik kolokacyjnych stosowanych na nieregularnej siatce do przybliżonego rozwiązywania równań różniczkowych. Aby otrzymać rezultaty o odpowiedniej dokładności, współczynnik kształtu powinien być dobrany na bazie kompromisu pomiędzy dokładnością a uwarunkowaniem układu równań. Przedstawiony algorytm, zwany "leave one out", bierze te aspekty pod uwagę. Podsumowaniem artykułu są numeryczne testy, które pokazują użyteczność tego podejścia.

**Słowa kluczowe:** radialne funkcje bazowe, współczynnik kształtu, algorytm cross-validation

## 1. Introduction

In recent years a significant development of numerical methods for solving differential equations with the use of an unstructured grid has been observed. All these techniques are called meshless or meshfree methods as opposed to well-known mesh based methods such as finite element, finite difference, or finite volume method. The meshless methods can overcome some drawbacks of mesh based techniques associated with grid distortion, remeshing in adaptation approaches and handling problems characterized by complicated geometries. There are several formulations of meshless methods [1, 2]. Among them, one can distinguish methods that apply interpolant composed of radial basis functions (RBF) and use the collocation technique in order to discretize a differential problem – the Kansa method and RBF-based pseudospectral method (RBF-PS). There are many papers devoted to these methods [3–7]. The problem of choosing a respective value of the shape parameter for RBF is the issue which appears in almost all of these papers. This parameter is responsible for the flatness of RBF and influences the accuracy of the methods as well as conditioning of the system of algebraic equations that follows from the discretization procedure. To achieve high accuracy, the value of the parameter should be large but this leads to an ill-conditioned system, which cannot be easily solved [8]. To estimate a respective value, a trade-off is needed. This value is estimated mostly on the base of researcher's experience, but there are also a few more sophisticated approaches [9, 10].

The present paper shows the adaptation of a kind of cross validation algorithm to this end. The latter is called "leave one out" and it has been used in statistics and interpolation problems [5]. The paper presents an easy implementation of the approach in the Kansa method and demonstrates that the same value of the parameter is valid for the RBF-PS method. The layout of the paper is as follows: in section 2 two RBF-based numerical techniques are briefly described, in section 3 the use of the "leave one out" algorithm is demonstrated and finally in section 4 the numerical tests are shown.

## 2. RBF-based collocation methods for partial differential equations

There are several numerical methods for solving boundary-value problem. The latter can be written in a general form as:

$$\begin{aligned} Lu &= f \quad \text{in } \Omega, \\ Bu &= g \quad \text{on } \partial\Omega \end{aligned} \tag{1}$$

where $L$ and $B$ denote linear differential operators imposed on the sought function $u$ in the domain $\Omega$ and on the boundaries $\partial\Omega$, respectively, and $f$, $g$ are known functions. Among the methods that take advantage of irregularly distributed nodes for domain discretization, one can distinguish collocation techniques that employs interpolant consisting of RBF. Such interpolant has a general form, which is as follows:

$$u(\mathbf{x}) = \sum_{j=1}^{N} \alpha_j \varphi\left(\left\|\mathbf{x} - \mathbf{x}_j\right\|\right) \tag{2}$$

where $\mathbf{x}_i$, $i = 1, \ldots, N$ represent a set of irregularly distributed nodes in the domain as well as on the boundary. Among them one can distinguish interior nodes $\mathbf{x}_i^I, i = 1,...,N^I$ and the nodes imposed on the boundary $\mathbf{x}_i^B, i = 1,...,N^B$. In Eq. (2) $\varphi_j(\mathbf{x}) = \varphi\left(\left\|\mathbf{x} - \mathbf{x}_j\right\|\right)$ represents RBF and $\alpha_j$ are the interpolation coefficients.

One of the approach which falls into the mentioned category called the Kansa method, introduces function (2) to problem (1) and by the collocation technique transforms differential problem into a set of algebraic equations. Another one uses the RBF interpolant to determine discrete approximation of derivatives included in , thereby obtaining algebraic approximation of the considered problem – the RBF- PS method. The details of these methods are given below.

### 2.1. Kansa method

By introducing interpolation function described by Eq. (2) into Eq. (1) and by collocating at each node in the domain including boundaries, one gets:

$$\begin{aligned} &\sum_{j=1}^{N} \alpha_j \left[ L\varphi\left(\left\|\mathbf{x} - \mathbf{x}_j\right\|\right)\right]_{\mathbf{x}=\mathbf{x}_i^I} = f\left(\mathbf{x}_i\right), \quad i = 1,...,N^I, \\ &\sum_{j=1}^{N} \alpha_j \left[ B\varphi\left(\left\|\mathbf{x} - \mathbf{x}_j\right\|\right)\right]_{\mathbf{x}=\mathbf{x}_i^B} = g\left(\mathbf{x}_i\right), \quad i = 1,...,N^B \end{aligned} \tag{3}$$

Using matrix notation one can put Eq. (3) in the following way:

$$\mathbf{A}_{LB}\boldsymbol{\alpha} = \mathbf{b} \tag{4}$$

In Eq. $\mathbf{A}_{LB} = \begin{bmatrix} \mathbf{A}_L \\ \mathbf{A}_B \end{bmatrix}$, $\mathbf{b} = \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix}$, where $\left(\mathbf{A}_L\right)_{ij} = \left[ L\varphi\left(\left\|\mathbf{x} - \mathbf{x}_j\right\|\right)\right]_{\mathbf{x}=\mathbf{x}_i^I}, i = 1,...,N^I, j = 1,...,N$

$\left(\mathbf{A}_B\right)_{ij} = \left[ B\varphi\left(\left\|\mathbf{x} - \mathbf{x}_j\right\|\right)\right]_{\mathbf{x}=\mathbf{x}_i^B}, i = 1,...,N^B, j = 1,...,N$,

$\mathbf{f}_i = f(\mathbf{x}_i^I), i = 1,...,N^I, \mathbf{g}_i = g(\mathbf{x}_i^B), i = 1,...,N^B$

The interpolation coefficients are obtained from Eq. if only $\mathbf{A}_{LB}$ is invertible:

$$\boldsymbol{\alpha} = \mathbf{A}_{LB}^{-1}\mathbf{b} \tag{5}$$

The study on the invertibility of the Kansa matrix ($\mathbf{A}_{LB}$) can be found in several papers. One can conclude that although there are numerical examples showing that the matrix can be singular for arbitrary center locations [11], these cases are rare and many other works [12] indicate a successful application of the method.

Since the interpolation coefficients are determined, the approximate solution is described by interpolant (2).

### 2.2. RBF-PS method

RBF-PS is a combination of RBF interpolation with the pseudospectral technique. In this approach, the interpolant in the form of (2) is used to determine the discrete approximation of differential operator from Eq. (1). To this end, the interpolation conditions are taken into account:

$$\sum_{j=1}^{N} \alpha_j \varphi\left(\left\|\mathbf{x}_i - \mathbf{x}_j\right\|\right) = u_i, \quad i = 1,...,N \tag{6}$$

It allows for presenting the coefficients $\alpha_i$ in terms of the values of the sought function, which can be put in the matrix notation in the following way:

$$\boldsymbol{\alpha} = \mathbf{A}^{-1}\mathbf{u} \tag{7}$$

where $\boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 & \cdots & \alpha_N \end{bmatrix}^T$, $\mathbf{u} = \begin{bmatrix} u_1 & \cdots & u_N \end{bmatrix}^T$ and $\mathbf{A}_{ij} = \varphi(\left\|\mathbf{x}_i - \mathbf{x}_j\right\|)$, $i, j = 1,...,N$.

Then, by imposing an appropriate differential operator on the interpolant and evaluating it at each interior as well as boundary node, one gets:

$$\mathbf{u}_L = \mathbf{A}_L \boldsymbol{\alpha} \tag{8}$$

$$\mathbf{u}_B = \mathbf{A}_B \boldsymbol{\alpha} \tag{9}$$

where $\mathbf{u}_L$, $\mathbf{u}_B$ are discrete representation of appropriate derivatives and $\mathbf{A}_L$, $\mathbf{A}_B$ are the same matrices that appear in Eq. (4) .

Using Eq. (7) one can express derivatives $\mathbf{u}_L$ and $\mathbf{u}_B$ in Eqs. (8) and (9) in terms of the sought function values from the whole domain as:

$$\mathbf{u}_{LB} = \mathbf{A}_{LB}\mathbf{A}^{-1}\mathbf{u} \tag{10}$$

where $\mathbf{u}_{LB} = [\mathbf{u}_L, \mathbf{u}_B]^T$ and $\mathbf{A}_{LB}$ matrix is composed of $\mathbf{A}_L$, $\mathbf{A}_B$ in the the same way as in the Kansa method (Eq. (4)).

Matrix $\mathbf{A}_{LB}\mathbf{A}^{-1}$ is a discrete form of differential operators $L, B$ and is called as differentiation matrix in the nomenclature of pseudospectral methods. With the use of this matrix Eq. (1) can be easily discretized:

$$\mathbf{A}_{LB}\mathbf{A}^{-1}\mathbf{u} = \mathbf{b} \tag{11}$$

and solved for unknown function values, which yields:

$$\mathbf{u} = \mathbf{A}\mathbf{A}_{LB}^{-1}\mathbf{b} \tag{12}$$

From the above it can be clearly seen that the solvability of the problem using the RBF-PS is conditioned by the inevitability of the same matrix as in the Kansa approach.

As one can notice, the approach presented is similar to the Kansa method. The main difference between the RBF-PS and Kansa method is that in the latter we introduce the

interpolation function directly into differential equation obtaining the interpolation coefficients. With these coefficients, the interpolation function approximates the solution at any point of the domain. In the RBF-PS we use the same interpolation function to derive a discrete approximation of a differential operator at each node and then this approximation in the form of a differentiation matrix is used to discretize the equation. Finally, in the RBF-PS, the function values at the nodes are obtained as the solution.

Since the RBF-PS operates directly with function values (does not need to evaluate the interpolant) it is more efficient in non-linear problems and in time dependent problems, where a kind of iterations are required to obtain the approximate function values.

## 3. Adopting "leave-one-out" algorithm for boundary-value-problem

It was found [5, 6, 8] that the shape parameter has a significant influence on accuracy. A larger value of this parameter theoretically should make the solution more accurate but leads to an ill-conditioned system, which may not be accurately solved. Therefore, an important issue in using RBF based methods is the choice of the appropriate value of *c*.

One of the approaches that can be employed comes from the interpolation problem:

$$\mathbf{A}\boldsymbol{\alpha} = \mathbf{u} \tag{13}$$

where **A**, **α** and **u** are the RBF interpolation matrix, vector of interpolation coefficients and vector of nodal function values, respectively. Eq. is a result of the application of the interpolation conditions. In this case values of **u** are known.

The optimal value of *c* depends on the number of nodes and on the pattern of their distribution, on the right-hand-side vector and precision of computation. All these factors are taken into consideration in the approach as it is reported in [13]. The approach is based on cross-validation and is called "leave-one-out". In this algorithm, an optimal value of *c* is obtained by minimizing the error of an interpolant based on the data from which one of the nodes was "left out". The error at the *k*th node, which was left out can be obtained as:

$$E_k = u_k - u^{[k]}(\mathbf{x}_k) \tag{14}$$

where $u_k$ is the function value at this node and $u^{[k]}(\mathbf{x}_k) = \sum_{\substack{j=1 \\ j\neq k}}^{N} \alpha_j^{[k]} \varphi\left(\left\|\mathbf{x}-\mathbf{x}_j\right\|\right)$ is the RBF interpolant to the data $\mathbf{u} = [u_1, \ldots, u_{k-1}, u_{k+1}, \ldots, u_N]$.

Removing in turn each of the nodes, the vector of errors $\mathbf{E} = \begin{bmatrix} E_1 & \cdots & E_N \end{bmatrix}^{\mathrm{T}}$ can be composed. The norm of this vector indicates the quality of the fit, which depends on the shape parameter. By repeating calculations for different values of *c,* one can choose the optimal one – which minimizes the $\|\mathbf{E}\|$ norm.

Since the implementation of his strategy is rather expensive, Rippa [13] showed that $E_k$ can be computed in a simpler way as:

$$E_k = \frac{\alpha_k}{\mathbf{A}^{-1}_{kk}} \tag{15}$$

where $\alpha_k$ is the $k$th coefficient in the interpolant $u$ based on a full set of nodes and $\mathbf{A}^{-1}_{kk}$ is the $k$th element in the diagonal of the inverse of the interpolation matrix.

In the present paper, the above method for estimating the optimal $c$ is adopted to methods of solving differential equations presented in sections 2.1 and 2.2. Here, we follow directly the idea presented by Rippa, understanding the problem described by Eq. (1) as a kind of interpolation problem, but defined for derivatives.

At first, let us consider the method from section 2.1, where the discretized boundary-value-problem is approximated by Eq.(4). We can consider this system of equations as the interpolation conditions such as Eq.(13), but written for derivative of the sought function. In this case the interpolant applied at the interior nodes assumes the form of:

$$u_L(\mathbf{x}) = Lu(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i L\varphi_i\left(\left\|\mathbf{x}-\mathbf{x}_i\right\|\right) \tag{16}$$

and interpolant for approximation of boundary values is as follows:

$$u_B(\mathbf{x}) = Bu(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i B\varphi_i\left(\left\|\mathbf{x}-\mathbf{x}_i\right\|\right) \tag{17}$$

One can make use of the "leave one out" algorithm to obtain the solution of this interpolation problem in the case, where the $k$th node is omitted:

$$\boldsymbol{\alpha}^{[k]} = \left(\mathbf{A}^{[k]}_{LB}\right)^{-1}\mathbf{b}^{[k]} \tag{18}$$

With the obtained coefficients, the interpolant for derivative at the $k$th node is evaluated

$$u^{[k]}_L(\mathbf{x}^I_k) = \sum_{i=1,i\neq k}^{N} \alpha^{[k]}_i\left[L\varphi_i\left(\left\|\mathbf{x}-\mathbf{x}_i\right\|\right)\right]_{\mathbf{x}=\mathbf{x}^I_k} \quad \text{or} \quad u^{[k]}_B(\mathbf{x}^B_k) = \sum_{i=1,i\neq k}^{N} \alpha^{[k]}_i\left[B\varphi_i\left(\left\|\mathbf{x}-\mathbf{x}_i\right\|\right)\right]_{\mathbf{x}=\mathbf{x}^B_k} \tag{19}$$

yielding the error at the $k$th node as:

$$E_k = b_k - u^{[k]}_L(\mathbf{x}^I_k) \quad \text{or} \quad E_k = b_k - u^{[k]}_B(\mathbf{x}^B_k) \tag{20}$$

where $b_k$ is the $k$th element of the right-hand-side vector introduced by Eq.(4).

Making use of Rippa's acceleration (Eq.(15)), the above error can be computed faster by:

$$E_k = \frac{\alpha_k}{\left(\mathbf{A}_{LB}\right)^{-1}_{kk}} = \frac{\left[\left(\mathbf{A}_{LB}\right)^{-1}\cdot\mathbf{b}\right]_k}{\left(\mathbf{A}_{LB}\right)^{-1}_{kk}} \tag{21}$$

where $\mathbf{A}_{LB}$ is the inverse of derivative interpolation matrix based on full set of nodes, introduced by Eq.(4).

For the method presented in section 2.2, Eq. (11) can be considered as the interpolation problem defined for the derivatives and therefore can be the starting point for derivation of the formula for error in the context of "leave one out" algorithm. Unfortunately, in the RBF-PS approach we are not able to find directly the value of $u_L^{[k]}(\mathbf{x}_k)$. By solving the system in the form of Eq. (11), but defined by omitting the $k$th node, we obtain $u_i$, $i = 1, \ldots, k\text{-}1, k\text{+}1, \ldots, N$, which can be used by Eq. (10) to approximate the derivatives at all the nodes without the $k$th node. Therefore, formula (20) cannot be directly obtained – it requires the value of the derivative at the $k$th node.

To overcome this inconvenience, with the values of $\mathbf{u}^{[k]} = \mathbf{A}^{[k]}\left(\mathbf{A}_{LB}^{-1}\right)^{[k]}\mathbf{b}^{[k]}$ at all nodes without the $k$th node, one should make a step back and use Eq. (7) to calculate $\alpha_i^{[k]}$ yielding:

$$\boldsymbol{\alpha}^{[k]} = \left(\mathbf{A}^{[k]}\right)^{-1}\mathbf{u}^{[k]} = \left(\mathbf{A}^{[k]}\right)^{-1}\mathbf{A}^{[k]}\left(\mathbf{A}_{LB}^{[k]}\right)^{-1}\mathbf{b} = \left(\mathbf{A}_{LB}^{[k]}\right)^{-1}\mathbf{b} \tag{22}$$

where $\mathbf{A}_{LB}^{[k]}$ is a Kansa matrix derived without the $k$th node. It is obvious that the same coefficients as those used in Eq. (18) are obtained, which allows for evaluating the interpolant for derivative at the $k$th node and leads to the same formula for the error as in the Kansa approach – (Eq.(21)).

## 4. Numerical tests

To show the usefulness of the approach proposed in the last section several equations in 2D space have been solved with the Kansa and RBF-PS methods. Due to the limited space, the results of two of them are presented.

Example1. Poisson equation with Dirichlet boundary conditions:

$$\begin{aligned} &\nabla^2 u(x,y) = \sin(\pi\cdot x)\sin(\pi\cdot y), \quad (x,y)\in\Omega = [0,1]\times[0,1] \\ &u(x,y) = 0, \quad (x,y)\in\partial\Omega \end{aligned} \tag{23}$$

for which the analytical solution has the form of:

$$u = -\frac{1}{2\pi^2}\sin(\pi\cdot x)\sin(\pi\cdot y) \tag{24}$$

Example 2. 2D modified Helmholtz equation with non-homogeneous boundary conditions:

$$\begin{aligned} &\nabla^2 u(x,y) - u(x,y) = -(\pi^2+1)(y\sin(\pi\cdot x) + x\cos(\pi\cdot y)), \quad (x,y)\in\Omega = [0,1]\times[0,1] \\ &u(0,y) = 0,\ u(1,y) = \cos(\pi\cdot y),\ u(x,0) = x,\ u(x,1) = \sin(\pi\cdot x) - x \end{aligned} \tag{25}$$

whose exact solution has the form:

$$u(x,y) = y\sin(\pi x) + x\cos(y) \tag{26}$$

As a measure of the quality of results, an error norm in the following form has been introduced $\varepsilon = \sqrt{\sum_{i=1}^{N}\left(u_i^n - u_i^e\right)^2} \Big/ \sqrt{\sum_{i=1}^{N}\left(u_i^e\right)^2} \cdot 100\%$ , where $u^n$ is numerical solution, $u^e$ – the exact one. The obtained results are shown in Tables 1–4.

Table 1. Results for example 1 solved by Kansa's method

| $N$ (regular grid) | $c_{opt}$ | $\varepsilon_{min}$ | "leave one out" algorithm | |
|---|---|---|---|---|
| | | | $c$ | $\varepsilon$ |
| 81 | 2.2846 | 5.3802e-04 | 1.0670 | 5.6308e-04 |
| 121 | 0.9818 | 1.5387e-04 | 0.9518 | 8.7747e-04 |
| 169 | 0.9167 | 6.0927e-05 | 0.9217 | 1.2719e-04 |
| 225 | 0.8916 | 3.4936e-05 | 0.8816 | 5.8035e-05 |
| $N$ (irrreg. grid) | | | | |
| 81 | 1.8035 | 1.8135e-04 | 1.0370 | 4.6253e-04 |
| 121 | 1.3676 | 8.3723e-05 | 0.9367 | 2.7855e-04 |
| 169 | 0.9868 | 3.3615e-05 | 0.9367 | 3.8024e-05 |
| 225 | 0.8215 | 1.3580e-05 | 0.8265 | 1.6603e-05 |

Table 2. Results for example 1 solved by the RBF-PS method

| $N$ (regular grid) | $c_{opt}$ | $\varepsilon_{min}$ | "leave one out" algorithm | |
|---|---|---|---|---|
| | | | $c$ | $\varepsilon$ |
| 81 | 1.0720 | 5.3918e-04 | 1.0670 | 5.6319e-04 |
| 121 | 0.9818 | 1.5665e-04 | 0.9518 | 8.8616e-04 |
| 169 | 0.9167 | 1.0287e-04 | 0.9217 | 1.8572e-04 |
| 225 | 0.8866 | 3.9077e-05 | 0.8816 | 5.0806e-05 |
| $N$ (irreg.grid) | | | | |
| 81 | 1.7935 | 1.9638e-04 | 1.0370 | 4.6253e-04 |
| 121 | 1.2775 | 8.3239e-05 | 0.9367 | 2.7856e-04 |
| 169 | 0.9718 | 3.2367e-05 | 0.9367 | 3.6655e-05 |
| 225 | 0.8716 | 1.4629e-05 | 0.8265 | 2.0200e-05 |

It is obvious that the presented algorithm for finding a good value of the shape parameter gives the same results for Kansa's method as well as for the RBF-PS when applied on the same grid, since it makes use of the same matrix. Therefore, appropriate columns presenting

a good value of $c$ possess the same values, comparing between two discretization methods, but for clear comparison of the results they are included in both tables. For comparison, the optimal value of $c$ obtained on the base of the exact solution is also included in the tables. By analyzing the results one can conclude that the presented approach gives the values of $c$ that lead to acceptable results.

Table 3. Results for example 2 solved by Kansa's method

| $N$ (regular grid) | $c_{opt}$ | $\varepsilon_{min}$ | "leave one out" algorithm | |
|---|---|---|---|---|
| | | | $c$ | $\varepsilon$ |
| 81 | 2.4349 | 4.9988e-04 | 1.3075 | 9.000e-03 |
| 121 | 1.8186 | 2.6106e-04 | 1.1622 | 2.5000e-03 |
| 169 | 1.3626 | 1.4130e-04 | 1.0921 | 5.8126e-04 |
| 225 | 1.1973 | 9.0023e-05 | 0.9067 | 4.6734e-04 |
| $N$ (irrreg. grid) | | | | |
| 81 | 1.8587 | 1.2793e-04 | 1.5931 | 6.5335e-04 |
| 121 | 1.5230 | 1.0177e-04 | 0.9167 | 3.8001e-03 |
| 169 | 1.1171 | 9.9409e-05 | 1.1722 | 1.5911e-04 |
| 225 | 1.0019 | 7.0534e-05 | 0.9969 | 4.8329e-04 |

Table 4. Results for example 2 solved by the RBF-PS method

| $N$ (regular grid) | $c_{opt}$ | $\varepsilon_{min}$ | "leave one out" algorithm | |
|---|---|---|---|---|
| | | | $c$ | $\varepsilon$ |
| 81 | 2.3547 | 7.8125e-04 | 1.3075 | 9.000e-03 |
| 121 | 1.6733 | 3.0915e-04 | 1.1622 | 2.4174e-03 |
| 169 | 1.5680 | 2.0440e-04 | 1.0921 | 3.7698e-03 |
| 225 | 1.0620 | 1.5164e-04 | 0.9067 | 5.3632e-04 |
| $N$ (irreg.grid) | | | | |
| 81 | 1.8937 | 1.8087e-04 | 1.5931 | 6.6200e-04 |
| 121 | 1.4878 | 1.8640e-04 | 0.9167 | 3.7853e-03 |
| 169 | 0.9768 | 1.1707e-04 | 1.1722 | 4.7312e-04 |
| 225 | 0.8766 | 7.9021e-05 | 0.9969 | 3.3314e-04 |

## 5. Conclusion

In the paper, the problem of accuracy of two meshless collocation methods that employ RBF interpolation is considered. It is well-known that the accuracy of such methods depends on the value of the shape parameter included in RBF. The paper applies a kind of cross validation approach, known form interpolation problems, to find respective value of this parameter.

To this end, the system of algebraic equations following from the application of the Kansa or RBF-PS method is treated as a kind of interpolation problem. The "leave one out" approach takes into consideration several discretization and computational parameters to find the value of *c*, which is a great value of this approach. The use of this algorithm requires to set a range, which is searched for the optimal value of *c* and many evaluations of system matrix. These can be considered as some weaknesses of the algorithm that should be improved in future work.

**References**


[1] Belytschko T., Krongauz Y., Organ D., Flrming M., Krysl P., *Meshless methods: an overview and recent developments*, Computer Methods in Applied Mechanics and Engineering, vol. 139, 1996, 3-47.

[2] Liu G.R., *Meshlees Methods – Moving beyond the Finite Element Method*, CRC Press, Boca Raton, Florida 2003.

[3] Kansa E., *Multiquadrics – A scattered data approximation scheme with applications to computational fluid dynamics I: Surface approximations and partial derivative estimates*, Computers and Mathematics with Applications, vol. 19, 1990, 127-145.

[4] Kansa E., *Multiquadrics – A scattered data approximation scheme with applications to computational fluid dynamics I: Solutions to parabolic, hyperbolic, and elliptic partial differential equations*, Computers and Mathematics with Applications, vol. 19, 1990, 147-161.

[5] Fasshauer G.E., *Meshfree Approximation Methods with Matlab*, World Scientific Publishing, Singapore, 2007.

[6] Cheng A.H.D., *Multiquadrics and its shape parameter – a numerical investigation of error estimate, condition number and round-off error by arbitrary precision computation*, Engineering analysis with boundary elements, vol. 36, 2012, 220-239.

[7] Ferreira A.J.M, *A formulation of the multiquadric radial basis function method for the analysis of laminated composite plates*, Composit Structures, vol. 59, 2003, 385-92.

[8] Schaback R., *Error estimates and condition numbers for radial basis function interpolation*, Advances in Computational Mathematics, vol. 3, 1995, 251-264.

[9] Krowiak A., *Radial basis function-based pseudospectral method for static analysis of thin plates*, Engineering Analysis with Boundary Elements, vol. 71, 2016, 50-58.

[10] Krowiak A., *On choosing a value of shape parameter in Radial Basis Function collocation methods*, Numerical Methods for Partial Differential Equations, submitted for publication.

[11] Hon Y.C., Schaback R., *On nonsymmetric collocation by radial basis functions*, Appl. Math. Comput., vol. 119, 2001, 177-186.

[12] Chen W., Fu Z.J., Chen C.S., *Recent Advances in Radial Basis Function Collocation* Methods, Springer, 2014.

[13] Rippa S., *An algorithm for selecting a good value for the parameter c in radial basis function interpolation*, Adv. in Comput. Math., vol. 11, 1999, 193-210.