

## Research Article

Artur Krowiak\*

# Solving biharmonic problems on irregular domains by stably evaluated Gaussian kernel

<https://doi.org/10.1515/cls-2020-0006>

Received Apr 22, 2020; accepted Jun 02, 2020

**Abstract:** The paper extends recently developed idea of stable evaluation of the Gaussian kernel. Owing to this, the Gaussian radial basis function that is sensitive to the shape parameter can be stably evaluated and applied to interpolation problems as well as to solve differential equations, giving highly accurate results. But it can be done only with grids being the Cartesian product of sets of points, what limits the use of this idea to rectangular domains. In the present paper, by the association of an appropriate transformation with the mentioned method, the latter is applied to solve biharmonic problems on quadrilateral irregular domains. As an example, in the present work this approach is applied to solve bending as well as free vibration problem of thin plates. In the paper some strategies for the implementation of the boundary conditions are also presented and examined due to the accuracy. The numerical tests show high accuracy and usefulness of the method.

**Keywords:** stable RBF basis, Mercer series, Gaussian kernel, blending functions, biharmonic equation

## 1 Introduction

In recent years a significant development of the meshless methods can be observed. Many formulations of the methods have been developed [1–3]. The main their advantage is the flexibility in the discretization of the domain – scattered nodes can be used to this end. It simplifies preprocessing procedures and the use of adaptation algorithms, providing powerful tool for analyzing complex-shaped engineering structures. Some of these methods make use of the radial basis functions (RBFs) to form the interpolant (sought solution), since the RBFs are found to be very useful in problems of scattered data approximation [4–6].

The RBFs fall into a more general category of functions, called kernels. Most of RBFs contain a constant parameter, which controls their flatness. The value of the shape parameter has significant influence on the stability of the solution process and finally on the accuracy of the results. Some information on the issue can be found in several papers [7–10].

Unfortunately, the values that theoretically should lead to more accurate results make the system ill-conditioned, what causes numerical difficulties in the solution process and finally inaccurate results. A significant effort has been done to find an optimal value of the parameter [11–14], but most of the approaches are a kind of trade-off between stability and accuracy.

Another group of the approaches aims to find a new stable basis spanned over the space created by the RBFs.

In [15, 16] the new basis is obtained by QR or SVD factorization of kernel matrix. In [17] another interesting approach is presented. It differs from the mentioned ones by the fact that the kernel matrix, which is prone to be ill-conditioned, is not formed. In order to obtain the stable basis appropriate factorization of the kernel matrix is done on the base of its approximation following from Mercer's theorem. Thus far, this method has been successfully applied in some approximation problems and in solving differential equations [18–20]. The results obtained show that the instability due to the shape parameter is removed and the parameter can be tuned according to the theoretical findings, giving very accurate results.

The method gives promising results, but its variant in [18] can be applied in higher dimensions only on grids that are the Cartesian product of one-dimensional sets of points. It significantly limits the applicability of the method – the scattered nodes cannot be further used as in typical meshless techniques. It makes a serious problem in solving tasks on irregular domains that are often encountered in engineering problems.

To preserve the stable solution and allow the method to be useful for problems defined on irregular domains, in the present paper the method developed in [17] and modified in [18] is appropriately extended. To this end, the method is associated with a domain transformation tech-

\*Corresponding Author: Artur Krowiak: Institute of Computer Science, Mechanical Faculty, Cracow University of Technology, Al. Jana Pawła II 37, 31-864 Kraków, Poland; Email: artur.krowiak@pk.edu.pl, Tel.: +48 12 628 36 33

nique. The latter takes advantage of the blending function interpolation used to transform the quadrilateral regions [21]. So far, this technique has been successfully associated with some pseudospectral methods [22, 23].

In the paper, the method is used to solve biharmonic equations, which can describe some engineering problems, e.g. associated with analysis of thin plates. Using collocation methods to solve such type of problems one has to overcome an inconvenience following from the existence of multiple boundary conditions. In the paper two strategies to implement such boundary conditions are proposed and examined.

The paper is organized as follows: in section 2 the basic information on the kernel-based methods as well as on the derivation of the stable basis is provided, in section 3 two strategies to implement boundary conditions are proposed, in section 4 the treatment of irregular domains with the aid of the mentioned transformation is explained, while in section 5 the numerical results with some comments are presented. Finally in section 6 concluding remarks are drawn.

## 2 Stable basis for Gaussian kernel

The kernel function is understood as a real-valued function of two variables coming from  $d$ -dimensional space, i.e.  $K : \Omega \times \Omega \rightarrow \Omega \in \mathbb{R}^d$ . Using such a function one can conveniently construct a data dependent basis  $\{K(\cdot, \mathbf{x}_1), \dots, K(\cdot, \mathbf{x}_N)\}$  that are a crucial point of kernel-based methods. The latter are very useful in solving interpolation problems as well as differential equations defined on domains in higher dimensions.

### 2.1 Kernel-based method

Let us consider a boundary value problem in a general form as

$$Lu = f \text{ in } \Omega, \quad Bu = g \text{ on } \partial\Omega \quad (1)$$

where  $L$  and  $B$  denote linear differential operators imposed on the sought function  $u$  in the domain  $\Omega$  and on the boundary  $\partial\Omega$ , respectively and  $f, g$  are known functions.

Alternatively one can introduce corresponding eigenvalue problem as follows

$$Lu = \lambda u \text{ in } \Omega, \quad Bu = 0 \text{ on } \partial\Omega \quad (2)$$

where  $\lambda$  is an eigenparameter.

After discretizing the domain as well as the boundary using scatter nodes  $\mathbf{x}_i, i = 1, \dots, N$  we can construct the

sought solution for the differential problem using the kernel basis as

$$u_h(\mathbf{x}) = \sum_{j=1}^N c_j K(\mathbf{x}, \mathbf{x}_j) = \mathbf{k}(\mathbf{x})^T \mathbf{c} \quad (3)$$

where  $\mathbf{c}$  denotes the vector of interpolation coefficients to be determined.

By introducing Eq. (3) into Eq. (1) and by applying collocation procedure one obtains

$$\mathbf{K}_{LB} \cdot \mathbf{c} = \mathbf{F} \quad (4)$$

The structures of system matrix  $\mathbf{K}_{LB}$  and vector  $\mathbf{F}$  are as follows

$$\mathbf{K}_{LB} = \begin{bmatrix} \mathbf{K}_L \\ \mathbf{K}_B \end{bmatrix},$$

$$\text{where } \mathbf{K}_L = \begin{bmatrix} L\mathbf{k}(\mathbf{x}_1^i)^T \\ \vdots \\ L\mathbf{k}(\mathbf{x}_{N_i}^i)^T \end{bmatrix}, \quad \mathbf{K}_B = \begin{bmatrix} B\mathbf{k}(\mathbf{x}_1^b)^T \\ \vdots \\ B\mathbf{k}(\mathbf{x}_{N_b}^b)^T \end{bmatrix}$$

$$\mathbf{F} = \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix},$$

$$\text{where } \mathbf{f} = \begin{bmatrix} f(\mathbf{x}_1^i) \\ \vdots \\ f(\mathbf{x}_{N_i}^i) \end{bmatrix}, \quad \mathbf{g} = \begin{bmatrix} g(\mathbf{x}_1^b) \\ \vdots \\ g(\mathbf{x}_{N_b}^b) \end{bmatrix}$$

Similarly, the eigenvalue equation can be discretized by incorporating Eq. (3) into Eq. (2), yielding the generalized algebraic eigenvalue problem

$$\mathbf{K}_{LB} \cdot \mathbf{c} = \lambda \bar{\mathbf{K}} \cdot \mathbf{c} \quad (5)$$

where  $\bar{\mathbf{K}}$  is the extended kernel matrix of the form

$$\bar{\mathbf{K}} = \begin{bmatrix} \mathbf{K} \\ \mathbf{0} \end{bmatrix},$$

$$\text{where } \mathbf{K} = \begin{bmatrix} \mathbf{k}(\mathbf{x}_1^i)^T \\ \vdots \\ \mathbf{k}(\mathbf{x}_{N_i}^i)^T \end{bmatrix}.$$

In the above objects, there is the distinction between interior nodes  $\mathbf{x}_j^i, j = 1, \dots, N_i$  and boundary nodes  $\mathbf{x}_j^b, j = 1, \dots, N_b$ . Note that the total number of nodes is  $N = N_i + N_b$ .

If only the system matrix of Eq. (4) is not singular, the interpolation coefficients can be computed and the approximate solution in the form of Eq. (3) is determined. As one can see, the procedure of the kernel-based method applied to the solution of differential equations is straightforward.

Moreover, the solution is obtained in analytic form, what can be viewed as another advantage of the method. As the kernel functions, the RBFs are willingly used.

The main drawback of the method is high ill-conditioning of the system (Eq. (4)). When using RBFs this ill-conditioning can be controlled by properly determined shape parameter but then the theoretically estimated high accuracy of the method is not achievable.

In order to maintain the possibility of achieving spectral accuracy, in the present paper another way to overcome the instability is applied. To this end a stable basis for RBFs that has been derived from Mercer's theorem is used.

## 2.2 Derivation of stable basis

According to Mercer's theorem, every positive definite kernel can be represented by an infinite series composed of eigenvalues  $\lambda_n$  and eigenfunctions  $\varphi_n$  of associated Hilbert-Schmidt eigenvalue problem [17]. This representation can be put as follows

$$K(\mathbf{x}, \mathbf{z}) = \sum_{n=1}^{\infty} \lambda_n \varphi_n(\mathbf{x}) \varphi_n(\mathbf{z}) \quad (6)$$

Since the Gaussian RBF  $e^{-\varepsilon^2 \|\mathbf{x}-\mathbf{z}\|^2}$  is a positive definite kernel it can be presented in such a way. Generally, if one operates in  $d$ -dimensional spaces,  $d$ -dimensional eigenvalues and eigenfunctions for Gaussian kernel follow immediately from the univariate ones via tensor product form of this kernel. In most common one- or two-dimensional cases ( $\Omega \in \mathcal{R}$  or  $\Omega \in \mathcal{R}^2$ ), Eq. (6) for Gaussian kernel takes the form

$$e^{-\varepsilon^2(x-z)^2} = \sum_{n=1}^{\infty} \lambda_n \varphi_n(x) \varphi_n(z) \quad (7)$$

or

$$\begin{aligned} e^{-\varepsilon^2 \|\mathbf{x}-\mathbf{z}\|^2} &= \sum_{\mathbf{n} \in \mathcal{R}^2} \lambda_{\mathbf{n}} \varphi_{\mathbf{n}}(\mathbf{x}) \varphi_{\mathbf{n}}(\mathbf{z}) \\ &= \sum_{n_1=1}^{\infty} \sum_{n_2=1}^{\infty} \lambda_{n_2} \lambda_{n_1} \varphi_{n_1}(x_1) \varphi_{n_2}(x_2) \varphi_{n_1}(z_1) \varphi_{n_2}(z_2) \end{aligned} \quad (8)$$

where  $\mathbf{x} = (x_1, x_2)^T$ ,  $\mathbf{z} = (z_1, z_2)^T$ . The eigenvalues and eigenfunctions from Eq. (7) and (8) assume the following form

$$\varphi_n(x) = \gamma_n e^{-\delta^2 x^2} H_{n-1}(\alpha \beta x) \quad (9)$$

$$\lambda_n = \sqrt{\frac{\alpha^2}{\alpha^2 + \delta^2 + \varepsilon^2}} \left( \frac{\varepsilon^2}{\alpha^2 + \delta^2 + \varepsilon^2} \right)^{n-1} \quad (10)$$

where  $H_n$  is the Hermite polynomial of degree  $n$ , and

$$\begin{aligned} \beta &= \left( 1 + \left( \frac{2\varepsilon}{\alpha} \right)^2 \right)^{\frac{1}{4}}, \quad \gamma_n = \sqrt{\frac{\beta}{2^{n-1} \Gamma(n)}}, \\ \delta^2 &= \frac{\alpha^2}{2} (\beta^2 - 1) \end{aligned} \quad (11)$$

are constants including:  $\varepsilon$  – the shape parameter of the Gaussian function,  $\alpha$  – a parameter, which parametrizes the weight function in the Hilbert-Schmidt integral operator and the associated inner product.

To take advantage of Mercer's theorem in numerical computation the series has to be truncated. The number of terms, which is taken into account, in relation to number of nodes leads to three general approaches for derivation of alternate basis. These approaches are described in detail in [24]. In the present work the case that the number of terms equals the number of nodes is considered. It was found [20] that this case give a good trade-off between accuracy and economy of computation.

In this case the Mercer series given by Eq. (6) assumes the form

$$K(\mathbf{x}, \mathbf{x}_j) = \sum_{n=1}^N \lambda_n \varphi_n(\mathbf{x}) \varphi_n(\mathbf{x}_j) = \boldsymbol{\Phi}(\mathbf{x})^T \boldsymbol{\Lambda} \boldsymbol{\Phi}(\mathbf{x}_j) \quad (12)$$

$$\text{where } \boldsymbol{\Phi}(\mathbf{x}) = \begin{bmatrix} \varphi_1(\mathbf{x}) \\ \vdots \\ \varphi_N(\mathbf{x}) \end{bmatrix} \text{ and } \boldsymbol{\Lambda} = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_N \end{bmatrix}.$$

With this assumption the base vector from Eq. (3) can be put as

$$\begin{aligned} \mathbf{k}(\mathbf{x})^T &= \boldsymbol{\Phi}(\mathbf{x})^T \boldsymbol{\Lambda} [\boldsymbol{\Phi}(\mathbf{x}_1), \dots, \boldsymbol{\Phi}(\mathbf{x}_N)] \\ &= \boldsymbol{\Phi}(\mathbf{x})^T \boldsymbol{\Lambda} \boldsymbol{\Phi}^T \end{aligned} \quad (13)$$

and the sought solution from Eq. (3) takes the form

$$u_h(\mathbf{x}) = \boldsymbol{\Phi}(\mathbf{x})^T \boldsymbol{\Lambda} \boldsymbol{\Phi}^T \cdot \mathbf{c} \quad (14)$$

In above equations matrix  $\boldsymbol{\Phi}$  is as follows

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\Phi}(\mathbf{x}_1)^T \\ \vdots \\ \boldsymbol{\Phi}(\mathbf{x}_N)^T \end{bmatrix}.$$

Following the idea of the kernel-based method all we have to do is to introduce Eq. (14) into Eq. (1) or into Eq. (2), dependently on the problem considered, and collocate the governing equation at the interior nodes and the boundary condition at the boundary nodes, what, in the case of the boundary-value problem, yields

$$\boldsymbol{\Phi}_{LB} \boldsymbol{\Lambda} \boldsymbol{\Phi}^T \cdot \mathbf{c} = \mathbf{F} \quad (15)$$

In above equations matrix  $\Phi_{LB}$  has the structure

$$\Phi_{LB} = \begin{bmatrix} \Phi_L \\ \Phi_B \end{bmatrix},$$

$$\text{where } \Phi_L = \begin{bmatrix} L\phi(\mathbf{x}_1^i)^T \\ \vdots \\ L\phi(\mathbf{x}_{N_i}^i)^T \end{bmatrix}, \Phi_B = \begin{bmatrix} B\phi(\mathbf{x}_1^b)^T \\ \vdots \\ B\phi(\mathbf{x}_{N_b}^b)^T \end{bmatrix}.$$

Solving Eq. (15) with respect to interpolation coefficients  $\mathbf{c}$  and evaluating the sought solution (14) one obtains

$$u_h(\mathbf{x}) = \phi(\mathbf{x})^T \Lambda \Phi^T \Phi^{-T} \Lambda^{-1} \Phi_{LB}^{-1} \mathbf{F} = \phi(\mathbf{x})^T \Phi_{LB}^{-1} \mathbf{F} \quad (16)$$

The main conclusion following from Eq. (16) is that the eigenvalue matrix, which is the main source of ill-conditioning [17], is algebraically removed. The eigenfunctions contained in vector  $\phi(\mathbf{x})$  play the role of the basis for the assumed solution. To determine the solution one only needs to take the number of eigenfunctions that equals the number of nodes. With the use of these eigenfunctions, base vector  $\phi(\mathbf{x})$  and matrix  $\Phi_{LB}$  can be formulated and the solution can be constructed using formula (16). Since the main source of ill-conditioning (matrix  $\Lambda$ ) is removed the process of the inversion of  $\Phi_{LB}$  is much more stable than using Gaussian kernel, even for very small values of the shape parameter. It leads to theoretically proved spectral accuracy. Therefore vector  $\phi(\mathbf{x})$  can be treated as a vector of stable basis for assumed solution. The solution described by Eq. (14) can be alternatively put as

$$u_h(\mathbf{x}) = \phi(\mathbf{x})^T \cdot \mathbf{b} \quad (17)$$

where  $\mathbf{b}$  is the vector containing new interpolation coefficients. It has the form  $\mathbf{b} = \Lambda \Phi^T \cdot \mathbf{c}$ .

The eigenvalue equation (2), written with the use of expression given by Eq. (17), takes the form

$$\Phi_{LB} \cdot \mathbf{b} = \lambda \bar{\Phi} \cdot \mathbf{b} \quad (18)$$

where matrix  $\bar{\Phi}$  is a modification of matrix  $\Phi$  introduced in Eqs. (13) and (14), and has the structure

$$\bar{\Phi} = \begin{bmatrix} \phi(\mathbf{x}_1^i)^T \\ \vdots \\ \phi(\mathbf{x}_{N_i}^i)^T \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}.$$

Solving such defined eigenvalue problem one does not operate highly ill-conditioned matrix  $\Lambda$ , therefore eigenpairs  $(\lambda_i, \mathbf{b}_i)$ ,  $i = 1 \dots N_i$  can be stably and accurately obtained.

In higher dimensions, e.g.  $\mathbf{x} \in \mathcal{R}^2$ , the kernel method from [18] associated with stably evaluated Gaussian kernel work only on grids being the result of the Cartesian product. It is a serious limitation comparing to RBF kernels that work properly on scattered nodes.

Moreover, the use of the collocation procedure requires a special treatment of problems possessing multiple boundary conditions, such as biharmonic problems.

In the next sections some approaches are introduced to extend the possibility of the application of the method in biharmonic problems defined on irregular quadrilateral domains.

### 3 The use of the stable basis in solving biharmonic problems

Let us introduce a biharmonic problem in the form

$$\begin{aligned} \Delta^2 u &= f \text{ in } \Omega \in \mathcal{R}^2, \\ B_1 u &= g_1, B_2 u = g_2 \text{ on } \partial\Omega \end{aligned} \quad (19)$$

Since  $\Delta^2$  is the biharmonic operator, two boundary conditions on the boundary are required to make the problem well-posed. Similarly one can consider corresponding eigenvalue problem as

$$\begin{aligned} \Delta^2 u &= \lambda u \text{ in } \Omega \in \mathcal{R}^2, \\ B_1 u &= 0, B_2 u = 0 \text{ on } \partial\Omega \end{aligned} \quad (20)$$

where  $\lambda$  is the eigenparameter.

If  $\Omega$  is a rectangular area, the kernel-based method with the stable basis can be used to obtain analytic, approximate solution. In order to implement two boundary conditions at each boundary nodes two strategies are proposed.

#### 3.1 Implementation of boundary conditions – strategy 1

In this strategy the distinction is made between not only interior and boundary nodes but also between nodes adjacent to the boundary, denoted in Figure 1 by crosses.

At the interior nodes ( $\mathbf{x}_k^i$  – unfilled circle) the governing equation is applied while at the boundary nodes ( $\mathbf{x}_k^b$  – filled circle) the first boundary condition is implemented. The nodes denoted by crosses are unused during the collocation procedure. Instead of this the second boundary condition is written at these boundary nodes which are adjacent to the unused points. This gives the same number

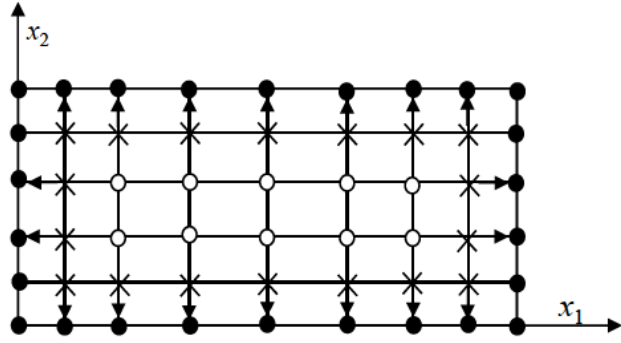


Figure 1: Discretization of the domain in strategy 1

of equations as the number of nodes and the latter corresponds to the number of unknowns.

In this strategy the solution is described by Eq. (16), where vector of base functions  $\Phi(\mathbf{x})$  is unchanged, the only difference is in some entries of system matrix  $\Phi_{LB}$ . The latter has now following form

$$\Phi_{LB} = \begin{bmatrix} \Phi_L \\ \Phi_{B_1} \\ \Phi_{B_2} \end{bmatrix}, \quad (21)$$

$$\text{where } \Phi_L = \begin{bmatrix} L\Phi(\mathbf{x}_1^i)^T \\ \vdots \\ L\Phi(\mathbf{x}_{N_i}^i)^T \end{bmatrix}, \quad \Phi_{B_1} = \begin{bmatrix} B_1\Phi(\mathbf{x}_1^b)^T \\ \vdots \\ B_1\Phi(\mathbf{x}_{N_b}^b)^T \end{bmatrix}, \quad \Phi_{B_2} = \begin{bmatrix} B_2\Phi(\mathbf{x}_1^{b_2})^T \\ \vdots \\ B_2\Phi(\mathbf{x}_{N_{b_2}}^{b_2})^T \end{bmatrix}$$

Note, that nodes  $\mathbf{x}_k^{b_2}$  are a subset of all boundary nodes. The subset contains these boundary nodes that are the projection of the adjacent nodes on the proper boundary, as it is shown in Figure 1. The approach leads to the system matrix of size  $N \times N$ .

The associated eigenvalue problem is also defined by matrix from Eq. (21) as well as by matrix  $\bar{\Phi}$  introduced in Eq. (18).

### 3.2 Implementation of boundary conditions – strategy 2

In this strategy the distinction is made only between interior and boundary nodes as Figure 2 suggests.

At the interior nodes the governing equation is applied, while at the boundary nodes two boundary conditions are implemented. To maintain the square system matrix  $\Phi_{LB}$ , the number of eigenfunctions has to be increased.

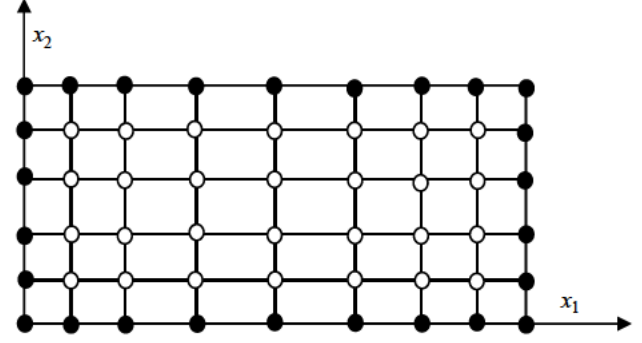


Figure 2: Discretization of the domain in strategy 2.

Instead of  $N$  eigenfunctions that correspond to total number of nodes, now their number is increased to  $N+N_b$ . The vector of basis functions assumes the form

$$\Phi(\mathbf{x}) = \begin{bmatrix} \varphi_1(\mathbf{x}) \\ \vdots \\ \varphi_{N+N_b}(\mathbf{x}) \end{bmatrix}$$

The structures and entries of the objects included in Eq. (16) and in the associated eigenvalue problem are similar as previously (Eq. (21)) with the difference that matrix  $\Phi_{B_2}$  is created with the use of all boundary nodes  $\mathbf{x}_k^b$ . The approach leads to system matrix of size  $(N+N_b) \times (N+N_b)$ .

## 4 Treatment of irregular domain

The kernel-based method with stable basis can be applied on grids being a result of the Cartesian product. It implies that the domain of the problem has a rectangular shape. When the boundaries are curved the direct application of the method is impossible. To extent the usefulness of the method the advantage of the domain transformation can be taken.

In this approach, graphically presented in Figure 3, one needs a transformation between regular domain in  $(\xi, \eta)$  plane and irregular one in  $(x, y)$  plane, what can be gen-

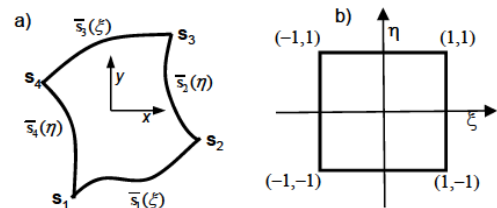


Figure 3: Domains: (a) physical domain; (b) computational one

erally written as

$$\begin{cases} x = x(\xi, \eta) \\ y = y(\xi, \eta) \end{cases} \quad (22)$$

To this end, a method that uses so-called blending functions [21] can be applied. In the method the transformation between the regions is done by following interpolation function

$$\begin{aligned} \mathbf{s} = & \frac{1}{2} \left[ (1 - \eta) \bar{\mathbf{s}}_1(\xi) + (1 + \xi) \bar{\mathbf{s}}_2(\eta) \right. \\ & \left. + (1 + \eta) \bar{\mathbf{s}}_3(\xi) + (1 - \xi) \bar{\mathbf{s}}_4(\eta) \right] - \frac{1}{4} \left[ (1 - \xi)(1 - \eta) \mathbf{s}_1 \right. \\ & \left. + (1 + \xi)(1 - \eta) \mathbf{s}_2 + (1 + \eta)(1 + \xi) \mathbf{s}_3 + (1 - \xi)(1 + \eta) \mathbf{s}_4 \right] \end{aligned} \quad (23)$$

where  $\mathbf{s} = [x, y]^T$ ,  $\bar{\mathbf{s}}_i$  are parametric curves that represent the curvilinear boundaries and  $\mathbf{s}_i$  are Cartesian coordinates of the corner points of the quadrilateral region in  $(x, y)$  plane.

If only it is possible to find parametric representations of the curves the method exactly maps computational domain onto physical one.

By the use of this transformation all spatial derivatives with respect to physical variables, contained in the equation, have to be written as derivatives with respect to  $\xi$  and  $\eta$ .

With the chain rule of differentiation the transformation of the first derivatives is obtained as

$$\begin{Bmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{Bmatrix} = \begin{bmatrix} x_\xi & y_\xi \\ x_\eta & y_\eta \end{bmatrix} \begin{Bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{Bmatrix} = \mathbf{J} \begin{Bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{Bmatrix} \quad (24)$$

where  $\mathbf{J}$  denotes the Jacobian matrix.

Then, the invers transformation is as follows

$$\begin{Bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{Bmatrix} = \mathbf{J}^{-1} \begin{Bmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{Bmatrix} \quad (25)$$

where the invers of the Jacobian matrix has the form

$$\mathbf{J}^{-1} = \begin{bmatrix} \xi_x & \eta_x \\ \xi_y & \eta_y \end{bmatrix} = \frac{1}{|\mathbf{J}|} \begin{bmatrix} y_\eta & -y_\xi \\ -x_\eta & x_\xi \end{bmatrix} \quad (26)$$

In Eq. (26)  $|\mathbf{J}|$  denotes determinant of the Jacobian matrix.

Similarly higher order derivatives with respect to the physical coordinates, in terms of the computational ones, can be obtained. In this way one is able to derive the transformation formulas for all partial derivatives contained in the biharmonic equation. This equation, i.e. Eq. (19), assumes the form

$$\begin{aligned} & D^{(41)} u_{,\xi\xi\xi\xi} + D^{(42)} u_{,\xi\xi\xi\eta} + D^{(43)} u_{,\xi\xi\eta\eta} + D^{(44)} u_{,\xi\eta\eta\eta} \quad (27) \\ & + D^{(45)} u_{,\eta\eta\eta\eta} + D^{(31)} u_{,\xi\xi\xi} + D^{(32)} u_{,\xi\xi\eta} + D^{(33)} u_{,\xi\eta\eta} \\ & + D^{(34)} u_{,\eta\eta\eta} + D^{(21)} u_{,\xi\xi} + D^{(22)} u_{,\xi\eta} + D^{(23)} u_{,\eta\eta} \end{aligned}$$

$$+ D^{(11)} u_{,\xi} + D^{(12)} u_{,\eta} = f$$

Similarly the eigenvalue problem, corresponding to Eq. (20), can be written.

The coefficients contained in Eq. (27) depend on the derivatives of physical variables with respect to computational ones and their explicit form can be found in [22].

Since Eq. (27) physically can describe the deflection of thin plate under load represented by function  $f$ , the associated boundary conditions can follow from the type of constraints on plate edges.

Typical homogenous boundary conditions are associated with the clamped edge (vanishing normal rotation to the edge) or with simply-supported edge (vanishing normal moment to the edge). They have the following form in  $(x, y)$  coordinates

$$u = 0, \quad u_x \cos \vartheta + u_y \sin \vartheta = 0 \quad (28)$$

or

$$\begin{aligned} u = 0, \quad (29) \\ \left( \cos^2 \vartheta + \nu \sin^2 \vartheta \right) u_{xx} + \left( \sin^2 \vartheta + \nu \cos^2 \vartheta \right) u_{yy} \\ + 2(1 - \nu) \cos \vartheta \sin \vartheta u_{xy} = 0 \end{aligned}$$

In above equations  $\vartheta$  is the angle between the normal to the boundary and  $x$  axes, while  $\nu$  is Poisson's ratio.

These boundary conditions transformed onto computational region in  $(\xi, \eta)$  coordinates assume the form

$$\begin{aligned} u = 0, \quad u_\xi = 0 \text{ along } \xi = \text{const}, \quad (30) \\ u_\eta = 0 \text{ along } \eta = \text{const} \end{aligned}$$

or

$$\begin{aligned} u = 0, \quad u_{\xi\xi} - \frac{2b}{a} u_{\xi\eta} + s u_\xi = 0 \text{ along } \xi = \text{const}, \quad (31) \\ u_{\eta\eta} - \frac{2b}{g} u_{\xi\eta} + t u_\eta = 0 \text{ along } \eta = \text{const} \end{aligned}$$

The explicit form of all coefficients contained in Eqs. (30) and (31) is shown in [22].

Once the transformation between the domains is established one can take advantage of the above equations. Following the procedure presented in section 2, the sought solution, now written as a linear combination of eigenfunctions of Gaussian kernel formed in computational coordinates

$$u_n(\boldsymbol{\zeta}) = \boldsymbol{\phi}(\boldsymbol{\zeta})^T \cdot \mathbf{b}, \quad (32)$$

where  $\boldsymbol{\zeta} = [\xi, \eta]^T$  should be introduced into Eq. (27) or into the corresponding eigenvalue problem as well as into appropriate boundary conditions (Eq. (30) or (31)). After the collocation procedure, where the chosen strategy for the implementation of boundary conditions is involved, interpolation coefficients  $\mathbf{b}$  can be found, giving the analytic approximate solution.

## 5 Numerical results

To validate the method the biharmonic problems have been solved on two irregular domains presented in Figure 4.

As a measure of the quality of the results the mean relative error (MRE) has been assumed. For the boundary-value problem the error is obtained by the formula

$$MRE = \frac{1}{N} \sum_{i=1}^N \frac{|u_h(\mathbf{x}_i) - u_i^{ref}|}{u_i^{ref}} \quad (33)$$

while for the eigenvalue one

$$MRE = \frac{1}{5} \sum_{i=1}^5 \frac{|\lambda_i - \lambda_i^{ref}|}{\lambda_i^{ref}} \quad (34)$$

In Eqs. (33) and (34)  $u_i^{ref}$  and  $\lambda_i^{ref}$  are the reference values computed by the differential quadrature method [25], which is a kind of pure numerical, pseudospectral technique. To compute the reference results this technique was also associated with domain transformation.

### 5.1 Quarter section of the elliptical domain

In this case the transformation between physical and computational domain is done by the function

$$\begin{aligned} x &= (0.6255 \cdot \eta + 2.6255) \cdot \cos(0.25 \cdot (\xi + 1) \cdot \pi), \\ y &= (0.8755 \cdot \eta + 1.8755) \cdot \sin(0.25 \cdot (\xi + 1) \cdot \pi) \end{aligned} \quad (35)$$

The results obtained are presented in Table 1-4 dependently on the type of the boundary conditions and the strategy of their implementation.

In Figure 5 two first vibration modes are presented.

In order to show the stability of the method due to the shape parameter, in Figure 6 the influence of this parameter on the accuracy as well as on the condition number of the system is shown, when solving BVP. For comparison, similar results obtained by Gaussian kernel are also presented in the figure.

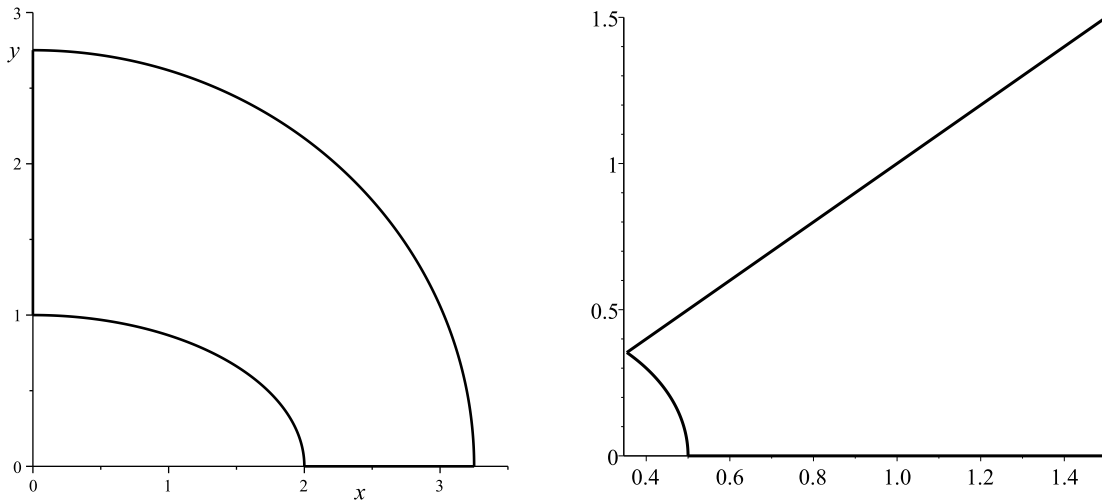


Figure 4: Shapes of the domains: quarter section of the elliptical domain (left), triangular domain with corner cut-out (right).

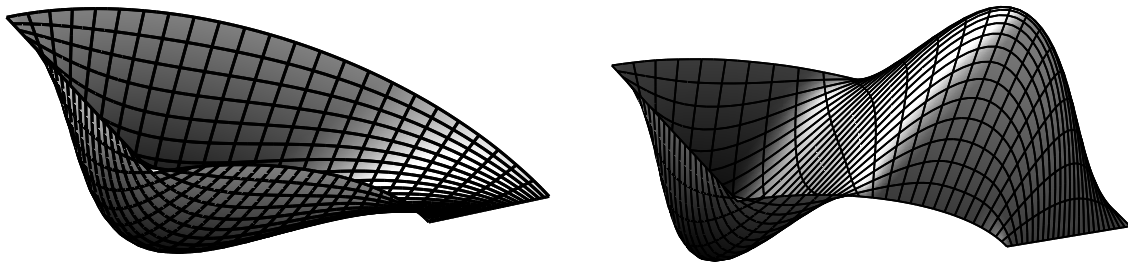
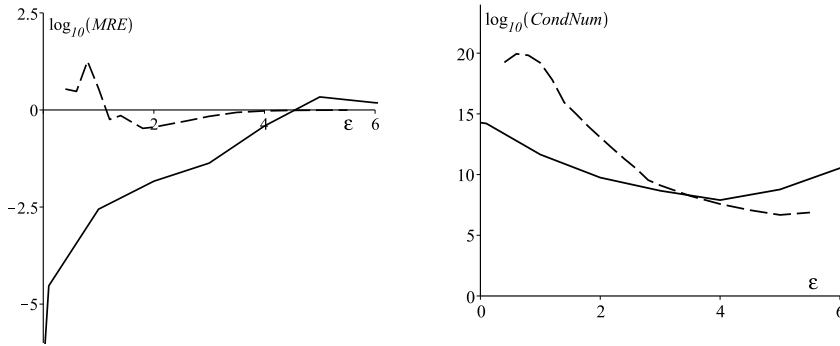


Figure 5: The first (left) and the second (right) vibration mode for the simply-supported quarter section of the elliptical plate.



**Figure 6:** Influence of the shape parameter on the accuracy – left, and condition number – right. Stable basis – solid line, Gaussian kernel – dash line. Results for the BVP on the quarter section of the elliptical domain with boundary conditions given by Eq. (28) implemented by strategy 1.

**Table 1:** The results from solving the biharmonic problems on the quarter section of the elliptical domain. The boundary conditions given by Eq. (28) implemented using strategy 1.

$N$	BVP			Eigenvalue problem				
	$u_{max}$	MRE	$\lambda_1$	$\lambda_2$	$\lambda_3$	$\lambda_4$	$\lambda_5$	MRE
$11 \times 11$	1.686e-2	<b>2.970e-7</b>	9.598	12.742	16.910	22.144	24.093	<b>9.545e-7</b>
Reference results	1.686e-2		9.598	12.742	16.910	22.144	24.093	
$13 \times 13$	1.657e-2	<b>5.057e-5</b>	9.595	12.715	16.721	22.382	24.071	<b>3.462e-7</b>
Reference results	1.657e-2		9.595	12.715	16.721	22.382	24.071	
$15 \times 15$	1.686e-2	<b>1.271e-5</b>	9.591	12.718	16.767	22.251	24.039	<b>6.153e-4</b>
Reference results	1.686e-2		9.595	12.717	16.745	22.250	24.068	

**Table 2:** The results from solving the biharmonic problems on the quarter section of the elliptical domain. The boundary conditions given by Eq. (28) implemented using strategy 2.

$N$	BVP			Eigenvalue problem				
	$u_{max}$	MRE	$\lambda_1$	$\lambda_2$	$\lambda_3$	$\lambda_4$	$\lambda_5$	MRE
$11 \times 11$	1.685e-2	<b>1.084e-3</b>	9.595	12.717	16.737	22.085	24.067	<b>3.257e-3</b>
Reference results	1.686e-2		9.598	12.742	16.910	22.144	24.093	
$13 \times 13$	1.657e-2	<b>2.116e-3</b>	9.601	12.714	16.761	22.275	24.068	<b>1.620e-3</b>
Reference results	1.657e-2		9.595	12.715	16.721	22.382	24.071	
$15 \times 15$	1.686e-2	<b>1.019e-2</b>	9.592	12.717	16.363	22.225	24.007	<b>5.356e-3</b>
Reference results	1.686e-2		9.595	12.717	16.745	22.250	24.068	

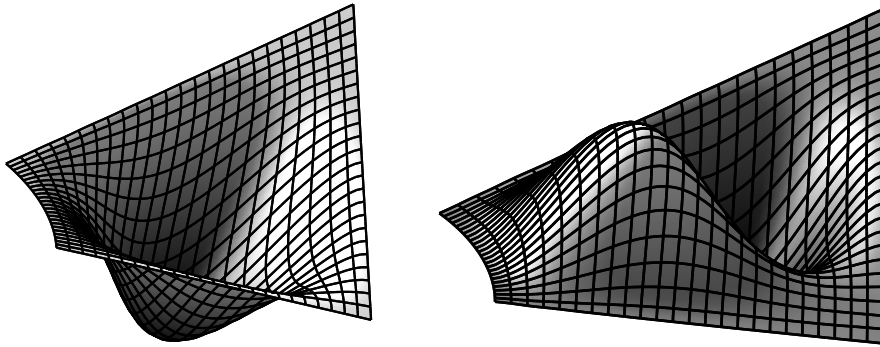
**Table 3:** The results from solving the biharmonic problems on the quarter section of the elliptical domain. The boundary conditions given by Eq. (29) implemented using strategy 1.

$N$	BVP			Eigenvalue problem				
	$u_{max}$	MRE	$\lambda_1$	$\lambda_2$	$\lambda_3$	$\lambda_4$	$\lambda_5$	MRE
$11 \times 11$	6.641e-2	<b>1.878e-7</b>	3.989	7.429	10.090	16.002	18.007	<b>8.231e-2</b>
Reference results	6.641e-2		4.841	7.626	11.661	16.007	16.757	
$13 \times 13$	6.608e-2	<b>3.687e-7</b>	4.918	7.625	11.496	16.008	16.988	<b>6.214e-3</b>
Reference results	6.608e-2		4.841	7.592	11.394	16.007	16.954	
$15 \times 15$	6.557e-2	<b>2.449e-4</b>	4.967	7.750	11.344	16.004	16.654	<b>1.206e-2</b>
Reference results	6.556e-2		4.842	7.595	11.444	16.008	16.569	

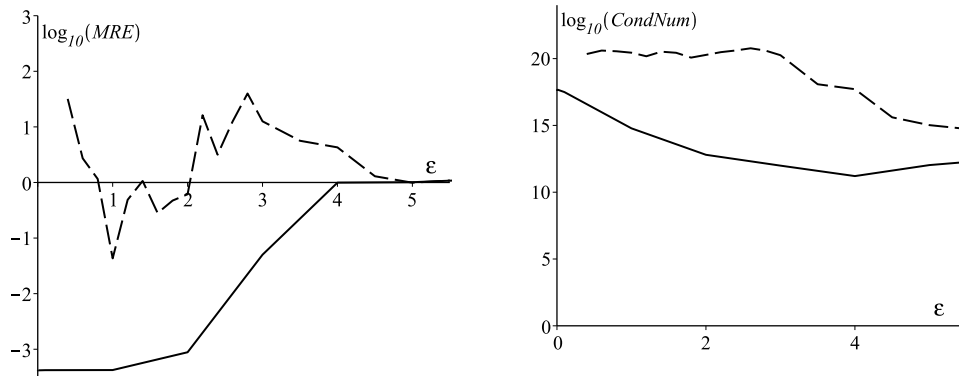


**Table 4:** The results from solving the biharmonic problems on the quarter section of the elliptical domain. The boundary conditions given by Eq. (29) implemented using strategy 2.

$N$	BVP		Eigenvalue problem					MRE
	$u_{max}$	MRE	$\lambda_1$	$\lambda_2$	$\lambda_3$	$\lambda_4$	$\lambda_5$	
$11 \times 11$	$6.642e-2$	<b><math>4.833e-4</math></b>	4.849	7.599	11.450	16.008	16.581	<b><math>6.761e-3</math></b>
Reference results	$6.641e-2$		4.841	7.626	11.661	16.007	16.757	
$13 \times 13$	$6.608e-2$	<b><math>1.300e-4</math></b>	5.356	7.640	10.884	16.004	16.621	<b><math>3.542e-2</math></b>
Reference results	$6.608e-2$		4.841	7.592	11.394	16.007	16.954	
$15 \times 15$	$6.557e-2$	<b><math>1.214e-4</math></b>						<b><math>2.062e-2</math></b>
Reference results	$6.556e-2$		4.842	7.595	11.444	16.008	16.569	



**Figure 7:** The first (left) and the second (right) vibration mode for the clamped triangular plate with corner cut-out.



**Figure 8:** Influence of the shape parameter on the accuracy – left, and condition number – right. Stable basis – solid line, Gaussian kernel – dash line. Results for the BVP on the triangular domain with corner cut-out with boundary conditions given by Eq. (28) implemented by strategy 1.

## 5.2 Triangular domain with corner cut-out

In this case the transformation between physical and computational domain is done by the function

$$\begin{aligned}
 x &= 0.25 \cdot \cos(0.125 \cdot (\eta + 1) \cdot \pi) (1 - \xi) & (36) \\
 &+ 0.75 \cdot (\xi + 1) \\
 y &= 0.25 \cdot \sin(0.125 \cdot (\eta + 1) \cdot \pi) (1 - \xi) \\
 &+ 0.75 \cdot (\xi + \eta + \xi\eta + 1)
 \end{aligned}$$

The results obtained are presented in Tables 5 – 8 dependently on the type of the boundary conditions and the strategy of their implementation.

In Figure 7 two first vibration modes are presented.

The influence of the shape parameter on the accuracy and condition number is shown in Figure 8.

**Table 5:** The results from solving the biharmonic problems on the triangular domain with corner cut-out. The boundary conditions given by Eq. (28) implemented using strategy 1.

$N$	BVP			Eigenvalue problem				
	$u_{max}$	MRE	$\lambda_1$	$\lambda_2$	$\lambda_3$	$\lambda_4$	$\lambda_5$	MRE
$11 \times 11$	$8.964e-4$	<b><math>4.623e-7</math></b>	41.788	71.228	87.881	110.452	130.841	<b><math>4.518e-7</math></b>
Reference results	$8.964e-4$		41.788	71.228	87.881	110.452	130.841	
$13 \times 13$	$8.967e-4$	<b><math>7.227e-3</math></b>	41.787	71.263	87.898	110.702	130.377	<b><math>1.746e-6</math></b>
Reference results	$8.965e-4$		41.787	71.263	87.898	110.702	130.377	
$15 \times 15$	$8.991e-4$	<b><math>1.779e-1</math></b>	41.938	71.256	87.704	110.975	130.417	<b><math>1.695e-3</math></b>
Reference results	$8.983e-4$		41.786	71.257	87.895	110.688	130.426	

**Table 6:** The results from solving the biharmonic problems on the triangular domain with corner cut-out. The boundary conditions given by Eq. (28) implemented using strategy 2.

$N$	BVP			Eigenvalue problem				
	$u_{max}$	MRE	$\lambda_1$	$\lambda_2$	$\lambda_3$	$\lambda_4$	$\lambda_5$	MRE
$11 \times 11$	$8.961e-4$	<b><math>8.379e-2</math></b>	41.791	71.261	87.905	110.344	130.471	<b><math>9.256e-4</math></b>
Reference results	$8.964e-4$		41.788	71.228	87.881	110.452	130.841	
$13 \times 13$	$8.9604$	<b><math>1.055e-2</math></b>	41.772	71.255	87.893	110.610	130.451	<b><math>3.840e-4</math></b>
Reference results	$8.965e-4$		41.787	71.263	87.898	110.702	130.377	
$15 \times 15$	$8.982e-4$	<b><math>3.380e-2</math></b>	42.711	71.548	82.654	115.608	130.435	<b><math>2.607e-2</math></b>
Reference results	$8.983e-4$		41.786	71.257	87.895	110.688	130.426	

**Table 7:** The results from solving the biharmonic problems on the triangular domain with corner cut-out. The boundary conditions given by Eq. (29) implemented using strategy 1.

$N$	BVP			Eigenvalue problem				
	$u_{max}$	MRE	$\lambda_1$	$\lambda_2$	$\lambda_3$	$\lambda_4$	$\lambda_5$	MRE
$11 \times 11$	$3.133e-3$	<b><math>7.001e-8</math></b>	22.247	45.621	58.890	77.534	96.667	<b><math>1.075e-6</math></b>
Reference results	$3.133e-3$		22.247	45.621	58.890	77.534	96.667	
$13 \times 13$	$3.133e-3$	<b><math>2.721e-5</math></b>	22.250	45.700	58.940	77.872	96.147	<b><math>1.394e-7</math></b>
Reference results	$3.133e-3$		22.250	45.700	58.940	77.872	96.147	
$15 \times 15$	$3.134e-3$	<b><math>6.387e-4</math></b>	22.251	45.699	58.940	7.853	96.263	<b><math>1.874e-6</math></b>
Reference results	$3.138e-3$		22.251	45.699	58.940	77.853	96.263	

**Table 8:** The results from solving the biharmonic problems on the triangular domain with corner cut-out. The boundary conditions given by Eq. (29) implemented using strategy 2.

$N$	BVP			Eigenvalue problem				
	$u_{max}$	MRE	$\lambda_1$	$\lambda_2$	$\lambda_3$	$\lambda_4$	$\lambda_5$	MRE
$11 \times 11$	$3.133e-3$	<b><math>4.171e-4</math></b>	22.251	45.699	58.940	77.853	96.264	<b><math>2.205e-3</math></b>
Reference results	$3.133e-3$		22.247	45.621	58.890	77.534	96.667	
$13 \times 13$	$3.133e-3$	<b><math>2.002e-4</math></b>	22.251	45.699	58.940	77.853	96.251	<b><math>2.795e-4</math></b>
Reference results	$3.133e-3$		22.250	45.700	58.940	77.872	96.147	
$15 \times 15$	$3.138e-3$	<b><math>9.752e-4</math></b>	22.251	45.699	58.940	77.853	96.263	<b><math>3.063e-5</math></b>
Reference results	$3.138e-3$		22.251	45.699	58.940	77.853	96.263	

### 5.3 Comments to the results

It should be mentioned that all the results contained in the tables have been obtained using Chebyshev-Gauss-Lobatto grid (points concentrated near boundaries) and with the shape parameter  $\varepsilon = 0.01$ .

General analysis of the results obtained indicates that the method provides solutions that are very close to the reference ones.

Comparing MRE between appropriate tables it is easy to notice that the implementation of boundary conditions with the use of strategy 1 leads to more accurate results, despite of the fact that this strategy employs smaller number of interior nodes to discretize the governing equation than strategy 2. The reason may lie in conditioning of the system. Strategy 2 uses more base functions than strategy 1 for the same number of total points. It means that the Hermite polynomials of higher degrees are employed in the discretization process, what makes conditioning of the system worse.

This fact also means that increasing the number of nodes does not always lead to more accurate results, regardless of the strategy used.

Analyzing Figures 6 and 8 one can conclude that the derived bases is stable due to the shape parameter – the influence of this parameter on the conditioning of the system is not as significant as for the conventional Gaussian kernel. Therefore very small values of the parameter still allow to carry out stable computation and give very accurate results, according to the theoretical findings on RBF kernels.

## 6 Conclusion

Although the modified kernel method with stable basis is an improvement of the well-known RBF collocation methods it limits their applicability. The RBF kernels easily allow to take advantage of the scattered nodes, while the mentioned method does not. The method can be applied on the Cartesian product grids. It means that the method cannot be used directly to solve the problems with curvilinear boundaries.

In the present paper an extension of the applicability of the kernel method with stable basis has been shown. The method, which can be directly used for problems on regular domains has been applied on irregular ones. By using the blending functions, the transformation between irregular and computational domain has been done. Therefore the computation can be carried out on the regular grid.

Solving higher order equations such as biharmonic ones, the problem with implementation of multiple boundary conditions appears, when using collocation methods. In the paper this problem has been solved by the use of two strategies.

The results obtained indicate that the method provides very accurate results, maintaining its main feature – stability due to the shape parameter. It allows, according to the theoretical findings, to take very small value of this parameter and expect very accurate results.

On the other hand more collocation points require to use basis functions of higher degrees, what worsen the conditioning of the system and the results may not be improved in this case.

Therefore the reasonable choice is to use a coarse grid and control the accuracy by the shape parameter.

It is worth to notice that the use of the method requires to derive, operate and evaluate complicated mathematical expressions, what can viewed as a drawback. But with the use of modern computer algebra systems this task can be significantly simplified.

## References

- [1] Belytschko T., Krongauz Y., Organ D., Fleming M., Krysl P., Meshless methods: An overview and recent developments, *Comput. Methods Appl. Mech. Eng.*, 1996, 139, 3-47.
- [2] Liu G.R., *Meshfree methods : moving beyond the finite element method*, 2nd ed., CRC Press, Boca Raton, 2018.
- [3] Chen W., Fu Z.J., Chen C.S., *Recent advances in radial basis function collocation methods*, Springer Verlag, 2014.
- [4] Buhmann M.D., *Radial Basis Functions: Theory and Implementations*, Cambridge University Press, Cambridge, 2003.
- [5] Wendland H., *Scattered data approximation*, Cambridge University Press, Cambridge, 2010.
- [6] Fasshauer G.E., *Meshfree Approximation Methods with Matlab*, World Scientific Publishing, Singapore, 2007.
- [7] Madych W.R., Miscellaneous error bounds for multiquadric and related interpolators, *Comput. Math. with Appl.*, 1992, 24, 121-138.
- [8] Wu Z., Schaback R., Local Error Estimates for Radial Basis Function Interpolation of Scattered Data, *IMA J. Numer. Anal.*, 1992, 13, 13-27.
- [9] Schaback R., Error estimates and condition numbers for radial basis function interpolation, *Adv. Comput. Math.*, 1995, 3, 251-264.
- [10] Cheng A.H.D., Multiquadric and its shape parameter - A numerical investigation of error estimate, condition number, and round-off error by arbitrary precision computation, *Eng. Anal. Bound. Elem.*, 2012, 36, 220-239.
- [11] Rippa S., An algorithm for selecting a good value for the parameter  $c$  in radial basis function interpolation, *Adv. Comput. Math.*, 1999, 11, 193-210.

- [12] Tsai C.H., Kolibal J., Li M., The golden section search algorithm for finding a good shape parameter for meshless collocation methods, *Eng. Anal. Bound. Elem.*, 2010, 34, 738-746.
- [13] Krowiak A., Domain-type RBF Collocation Methods for Biharmonic Problems, *Int. J. Comput. Methods.*, 2018, 15, 1850078:1-20.
- [14] Sarra S.A., Sturgill D., A random variable shape parameter strategy for radial basis function approximation methods, *Eng. Anal. Bound. Elem.*, 2009, 33, 1239-1245.
- [15] De Marchi S., Santin G., A new stable basis for radial basis function interpolation, *J. Comput. Appl. Math.*, 2013, 253, 1-13.
- [16] Pazouki M., Schaback R., Bases for kernel-based spaces, in: *J. Comput. Appl. Math.*, 2011, 236, 575-588.
- [17] Fasshauer G.E., Mccourt M.J., Stable evaluation of gaussian rbf interpolants, *SIAM J. Sci. Comput.*, 2012, 34, A737-A762.
- [18] Rashidinia J., Fasshauer G.E., Khasi M., A stable method for the evaluation of Gaussian radial basis function solutions of interpolation and collocation problems, *Comput. Math. with Appl.*, 2016, 72, 178-193.
- [19] Rashidinia J., Khasi M., Fasshauer G.E., A stable Gaussian radial basis function method for solving nonlinear unsteady convection–diffusion–reaction equations, *Comput. Math. with Appl.*, 2018, 75, 1831-1850.
- [20] Krowiak A., Podgórski J., Stable evaluation of gaussian kernel applied to interface problems, In: M. A. Loja et al. (Eds.), *Proceedings of Symcomp 2019 Conference (11-12 April 2019, Porto, Portugal)*, *Eccomas Conference Proceedings*, 2019, 85-96.
- [21] Gordon W.J., Hall C.A., Transfinite element methods: Blending-function interpolation over arbitrary curved element domains, *Numer. Math.*, 1973, 21, 109-129.
- [22] Shu C., Chen W., Du H., Free Vibration Analysis of Curvilinear Quadrilateral Plates by the Differential Quadrature Method, *J. Comput. Phys.*, 2000, 163, 452-466.
- [23] Hessari P., Kim S.D., Shin B-C., Numerical Solution for Elliptic Interface Problems Using Spectral Element Collocation Method, *Abstr. Appl. Anal.*, 2014, 2014, 780769: 1-11.
- [24] Fasshauer G., McCourt M., *Kernel-based Approximation Methods using Matlab*, World Scientific Publishing, Singapore 2015.
- [25] Shu C., *Differential Quadrature and Its Application in Engineering*, Springer, London, 2000.