3-21-2015

A Kansa-Radial Basis Function Method for Elliptic Boundary Value Problems in Annular Domains

Xiao Yan Liu
Taiyuan University of Technology

Andreas Karageorghis
University of Cyprus

C. S. Chen
Taiyuan University of Technology

Recommended Citation

Liu, X., Karageorghis, A., Chen, C. (2015). A Kansa-Radial Basis Function Method for Elliptic Boundary Value Problems in Annular Domains. *Journal of Scientific Computing, 65*(3), 1240-1269.
Available at: https://aquila.usm.edu/fac_pubs/18569

A KANSA-RADIAL BASIS FUNCTION METHOD FOR ELLIPTIC BOUNDARY VALUE PROBLEMS IN ANNULAR DOMAINS

XIAO-YAN LIU, ANDREAS KARAGEORGHIS, AND C. S. CHEN

Abstract. We employ a Kansa-radial basis function (RBF) method for the numerical solution of elliptic boundary value problems in annular domains. This discretization leads, with an appropriate selection of collocation points and for any choice of RBF, to linear systems in which the matrices possess block circulant structures. These linear systems can be solved efficiently using matrix decomposition algorithms (MDAs) and fast Fourier transforms (FFTs). A suitable value for the shape parameter in the various RBFs used is found using the leave-one-out cross validation (LOOCV) algorithm. In particular, we consider problems governed by the Poisson equation, the inhomogeneous biharmonic equation and the inhomogeneous Cauchy-Navier equations of elasticity. In addition to its simplicity, the proposed method can both achieve high accuracy and solve large-scale problems. The feasibility of the proposed techniques is illustrated by several numerical examples.

1. Introduction

During the past decade, radial basis function (RBF) collocation methods have attracted great attention in scientific computing and have been widely applied to solve a large class of problems in science and engineering. Among the various types of RBF collocation methods, the Kansa method [14], proposed in early 1990s is the most known. One of the main attractions of this method is its simplicity, since in it neither a boundary nor a domain discretization are required. This feature is especially useful for solving high dimensional problems in complex geometries. Furthermore, we have the freedom of choosing an RBF as the basis function which does not necessary satisfy the governing equation. Hence, the Kansa method is especially attractive for solving nonhomogeneous equations. In general, the multiquadric (MQ) function is the most commonly used RBF in the implementation of the Kansa method. Using the MQ, an exponential convergence rate has been observed and thus high accuracy beyond the reach of the traditional numerical methods such as finite element and finite difference methods can be achieved. Despite all the advantages mentioned above, the Kansa method has a number of unfavorable features. It is known that the determination of the optimal shape parameter of various RBFs is still a challenge. Currently, there are a variety of techniques [5, 11, 15, 25, 32] available for the determination of a suitable shape parameter. In this paper, we will use the so called leave-one-out cross validation (LOOCV) proposed by Rippa [32] to find a sub-optimal shape parameter. Another potential problem of the RBF collocation methods is the ill-conditioning of the resultant matrix. On the one hand, when the number of interpolation points is increased, the accuracy improves, while, on the other hand, the condition number becomes larger. This phenomenon is referred to as *the principle of uncertainty* by Schaback [33, 34]. Eventually, when the number of interpolation points becomes too large, the condition number becomes enormous and the solution breaks down. When using globally supported RBFs such as the MQ, the resultant matrix of the Kansa method is dense and ill-conditioned. Not only do we have a stability problem, but also the computational cost becomes very high. The Kansa method is thus not suitable for the solution of large-scale problems which require the use of a large number of interpolation points. In recent years, the *localized Kansa method* [26] has been developed for handling large-scale problems. In this approach, a local

Date: January 20, 2015.

2000 *Mathematics Subject Classification.* Primary 65N35; Secondary 65N21, 65N38.

Key words and phrases. radial basis functions, Poisson equation, biharmonic equation, Cauchy-Navier equations of elasticity, Fast Fourier transforms, Kansa method.

influence domain is established for each interpolation point and only a small number of neighbouring points is used to approximate the solution. The resultant matrix is thus sparse allowing for the use of a large number of collocation points. However, the accuracy of the localized Kansa method is rather low because the local influence domain only contains a small number of interpolation points.

In this paper, we couple the Kansa method and a matrix decomposition technique for solving problems using a large number of interpolation points. Since global RBFs are used as basis functions, we can also achieve high accuracy in numerical results. To be more specific, we consider the discretization of elliptic boundary value problems in annular domains using the Kansa method. For *any choice* of RBF, such appropriate discretizations lead to linear systems in which the coefficient matrices possess block circulant structures and which are solved efficiently using *matrix decomposition algorithms (MDAs)*. An MDA [1] is a direct method which reduces the solution of an algebraic problem to the solution of a set of independent systems of lower dimension with, often, the use of fast Fourier Transforms (FFTs). MDAs lead to considerable savings in computational cost and storage. This decomposition technique not only allows us to handle large-scale matrices but also makes it possible to implement the LOOCV technique to find the sub-optimal shape parameter of the RBFs used. It should be noted that the LOOCV technique is not suitable when the size of the matrix is too large. Such MDAs have been used in the past in various applications of the method of fundamental solutions (MFS) to boundary value problems in geometries possessing radial symmetry, see e.g., [16, 17]. A similar MDA was also applied for the approximation of functions and their derivatives using RBFs in circular domains in [20], see also, [13]. Some preliminary results for the Dirichlet Poisson problem have recently been presented in [19].

The paper is organized as follows. In Section 2 we present the Kansa method and the proposed MDA for both the Dirichlet and the mixed Dirichlet-Neumann Poisson problem. The Kansa-RBF discretization for the first and second biharmonic problems is given in Section 3. The corresponding discretization and MDA for the Dirichlet and the mixed Dirichlet-Neumann boundary value problems for the more challenging Cauchy-Navier equations of elasticity is described in Section 4. In Section 5 we present the results of three numerical experiments. In the first example various RBFs are tested and the normalized MQ turns out to be the most effective one. As a result, in the other examples, we focus on using the normalized MQ. We also show that the LOOCV technique is very effective for finding a good shape parameter of the RBFs. All three examples show that we can solve large-scale problems with high accuracy. Finally, in Section 6 some conclusions and ideas for future work are outlined.

2. The Poisson equation

2.1. **The problem.**

We first consider the Poisson equation

$$\Delta u = f \quad \text{in} \quad \Omega, \tag{2.1a}$$

subject to the Dirichlet boundary conditions

$$u = g_1 \quad \text{on} \quad \partial\Omega_1, \tag{2.1b}$$

$$u = g_2 \quad \text{on} \quad \partial\Omega_2, \tag{2.1c}$$

in the annulus

$$\Omega = \{\boldsymbol{x} \in \mathbb{R} : \varrho_1 < |\boldsymbol{x}| < \varrho_2\}\,. \tag{2.2}$$

The boundary $\partial\Omega = \partial\Omega_1 \cup \partial\Omega_2$, $\partial\Omega_1 \cap \partial\Omega_2 = \emptyset$ where $\partial\Omega_1 = \left\{\boldsymbol{x} \in \mathbb{R}^2 : |\boldsymbol{x}| = \varrho_1\right\}$ and $\partial\Omega_2 = \left\{\boldsymbol{x} \in \mathbb{R}^2 : |\boldsymbol{x}| = \varrho_2\right\}$.

2.2. **Kansa's method.** In Kansa's method [14] we approximate the solution u of boundary value problem (2.1) by a linear combination of RBFs [3, 7]

$$u_{\mathsf{N}}(x,y)=\sum_{\mathsf{n}=1}^{\mathsf{N}} a_{\mathsf{n}}\phi_{\mathsf{n}}(x,y), \qquad (x,y)\in\bar{\Omega}. \tag{2.3}$$

The RBFs $\phi_{\mathsf{n}}(x,y), \mathsf{n}=1,\dots,\mathsf{N}$ can be expressed in the form

$$\phi_{\mathsf{n}}(x,y)=\Phi(r_{\mathsf{n}}), \quad \text{where} \quad r_{\mathsf{n}}^2=(x-x_{\mathsf{n}})^2+(y-y_{\mathsf{n}})^2. \tag{2.4}$$

Thus each RBF ϕ_{n} is associated with a point $(x_{\mathsf{n}},y_{\mathsf{n}})$. These points $\{(x_{\mathsf{n}},y_{\mathsf{n}})\}_{\mathsf{n}=1}^{\mathsf{N}}$ are usually referred to as *centers.* The coefficients $\{a_{\mathsf{n}}\}_{\mathsf{n}=1}^{\mathsf{N}}$ in equation (2.3) are determined from the collocation equations

$$\Delta u(x_{\mathsf{m}},y_{\mathsf{m}}) \;=\; f(x_{\mathsf{m}},y_{\mathsf{m}}), \quad \mathsf{m}=1,\dots\mathsf{M}_{\text{int}}, \tag{2.5a}$$

$$u(x_{\mathsf{m}},y_{\mathsf{m}}) \;=\; g_1(x_{\mathsf{m}},y_{\mathsf{m}}), \quad \mathsf{m}=\mathsf{M}_{\text{int}}+1,\dots,\mathsf{M}_{\text{int}}+\mathsf{M}_{\text{bry}_1}, \tag{2.5b}$$

$$u(x_{\mathsf{m}},y_{\mathsf{m}}) \;=\; g_2(x_{\mathsf{m}},y_{\mathsf{m}}), \quad \mathsf{m}=\mathsf{M}_{\text{int}}+\mathsf{M}_{\text{bry}_1}+1,\dots,\mathsf{M}_{\text{int}}+\mathsf{M}_{\text{bry}}, \tag{2.5c}$$

where $\mathsf{M}_{\text{int}}+\mathsf{M}_{\text{bry}}=\mathsf{M}$ and the points $\{(x_{\mathsf{m}},y_{\mathsf{m}})\}_{\mathsf{m}=1}^{\mathsf{M}}\in\bar{\Omega}$ are the *collocation points.* Note that, in general, the collocation points are not the same as the centers and $\mathsf{M}\geq\mathsf{N}$.

In this work, however, we shall assume that $\mathsf{M}=\mathsf{N}$ and that the centres are the same as the collocation points. In particular, we define the M angles

$$\vartheta_m=\frac{2\pi(m-1)}{M}, \quad m=1,\dots,M, \tag{2.6}$$

and the N radii

$$\mathsf{r}_n=\varrho_1+(\varrho_2-\varrho_1)\,\frac{n-1}{N-1}, \quad n=1,\dots,N. \tag{2.7}$$

The collocation points $\{(x_{mn},y_{mn})\}_{m=1,n=1}^{M,N}$ are defined as follows:

$$x_{mn}=\mathsf{r}_n\cos(\vartheta_m+\frac{2\pi\alpha_n}{N}), \quad y_{mn}=\mathsf{r}_n\sin(\vartheta_m+\frac{2\pi\alpha_n}{N}), \quad m=1,\dots,M,\; n=1,\dots,N. \tag{2.8}$$

In (2.8) the parameters $\{\alpha_n\}_{n=1}^{N}\in[-1/2,1/2]$ correspond to rotations of the collocation points and may be used to produce more uniform distributions. Typical distributions of collocation points without rotation ($\alpha_n=0, n=1,\dots,n$) and with rotation are given in Figure 1. In the current application of Kansa's method, we take

$$u_{MN}(x,y)=\sum_{m=1}^{M}\sum_{n=1}^{N}a_{mn}\phi_{mn}(x,y), \qquad (x,y)\in\bar{\Omega}, \tag{2.9}$$

where the $\mathsf{M}=MN$ coefficients $\{(a_{mn})\}_{m=1,n=1}^{M,N}$ are unknown. These coefficients are determined by collocating the differential equation (2.1a) and the boundary conditions (2.1b)-(2.1c) in the following way:

$$\begin{gathered}\Delta u_{MN}(x_{mn},y_{mn})=f(x_{mn},y_{mn}), \quad m=1,\dots,M,\; n=2,\dots,N-1,\\ u_{MN}(x_{m1},y_{m1})=g_1(x_{m1},y_{m1}), \quad u_{MN}(x_{mN},y_{mN})=g_2(x_{mN},y_{mN}), \quad m=1,\dots,M,\end{gathered} \tag{2.10}$$

yielding a total of $\mathsf{M}=MN$ equations.
By vectorizing the arrays of unknown coefficients and collocation points from

$$\mathsf{a}_{(n-1)M+m}=a_{mn}, \quad \mathsf{x}_{(n-1)M+m}=x_{mn}, \quad \mathsf{y}_{(n-1)M+m}=y_{mn}, \quad m=1,\dots,M,\; n=1,\dots,N, \tag{2.11}$$

equations (2.10) yield a linear system of the form

$$A\,\boldsymbol{a}=\begin{pmatrix} A_{1,1} & A_{1,2} & \dots & A_{1,N}\\ A_{2,1} & A_{2,2} & \dots & A_{2,N}\\ \vdots & \vdots & \ddots & \vdots\\ A_{N,1} & A_{N,2} & \dots & A_{N,N}\end{pmatrix}\begin{pmatrix}\boldsymbol{a}_1\\ \boldsymbol{a}_2\\ \vdots\\ \boldsymbol{a}_N\end{pmatrix}=\begin{pmatrix}\boldsymbol{b}_1\\ \boldsymbol{b}_2\\ \vdots\\ \boldsymbol{b}_N\end{pmatrix}=\boldsymbol{b}\,. \tag{2.12}$$

The $M \times M$ submatrices $A_{n_1,n_2},\ n_1, n_2 = 1, \ldots, N$ are defined as follows:

$$\left(A_{n_1,n_2}\right)_{m_1,m_2} = \Delta\phi_{m_2,n_2}(x_{m_1,n_1}, y_{m_1,n_1}),\ m_1, m_2 = 1, \ldots, M,\ n_1 = 2, \ldots, N-1,\ n_2 = 1, \ldots, N, \tag{2.13a}$$

$$\left(A_{1,n}\right)_{m_1,m_2} = \phi_{m_2,n}(x_{m_1,1}, y_{m_1,1}), \tag{2.13b}$$

$$m_1, m_2 = 1, \ldots, M,\ n = 1, \ldots, N,$$

$$\left(A_{N,n}\right)_{m_1,m_2} = \phi_{m_2,n}(x_{m_1,N}, y_{m_1,N}), \tag{2.13c}$$

while the $M \times 1$ vectors $\boldsymbol{a}_n, \boldsymbol{b}_n, n = 1, \ldots, N$ are defined as

$$\left(\boldsymbol{a}_n\right)_m = a_{mn},\ m = 1, \ldots, M,\ N = 1, \ldots, N,$$

$$\left(\boldsymbol{b}_n\right)_m = f(x_{mn}, y_{mn}), \quad m = 1, \ldots, M,\ n = 2, \ldots, N-1,$$

$$\left(\boldsymbol{b}_1\right)_m = g_1(x_{m1}, y_{m1}), \quad \left(\boldsymbol{b}_N\right)_m = g_2(x_{mN}, y_{mN}), \quad m = 1, \ldots, M.$$

2.2.1. *Neumann boundary conditions.* Suppose that instead of boundary condition (2.1b) we had the Neumann boundary condition

$$\frac{\partial u}{\partial \mathrm{n}} = g_1 \quad \text{on} \quad \partial\Omega_1, \tag{2.14}$$

where $\mathbf{n}(x, y) = (\mathrm{n}_x, \mathrm{n}_y)$ denotes the outward normal vector to the boundary at the point (x, y). The corresponding submatrices $\left(A_{1,n}\right)$, $n = 1, \ldots, N$, in (2.13b) are now defined by

$$\left(A_{1,n}\right)_{m_1,m_2} = \frac{\partial\phi_{m_2,n}}{\partial \mathrm{n}}(x_{m_1,1}, y_{m_1,1}), \quad m_1, m_2 = 1, \ldots, M. \tag{2.15}$$

As proved in the Appendix (Lemma 1) each of the $M \times M$ submatrices $A_{n_1,n_2},\ n_1, n_2 = 1, \ldots, N$, in the coefficient matrix in (2.12) is circulant [6]. Hence matrix A in system (2.12) is *block circulant*.

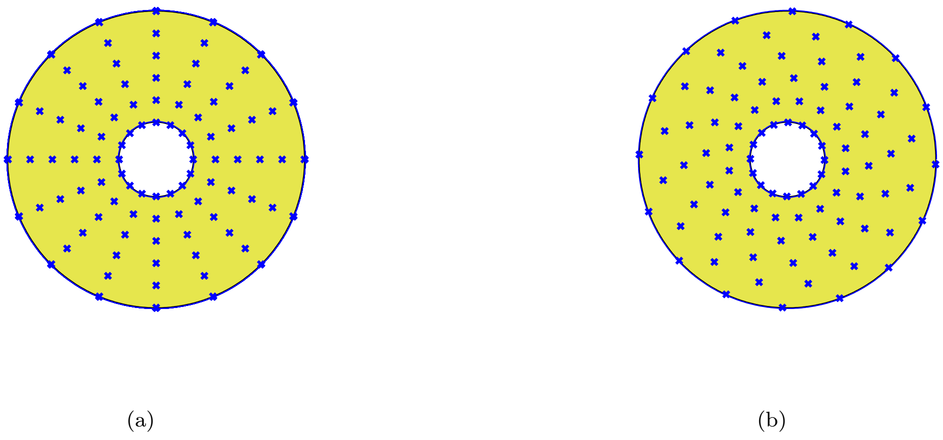

FIGURE 1. Typical discretization of the domain with (a) no rotation of the collocation points and (b) with rotation of the collocation points. The crosses (+) denote the collocation points.

2.3. **Matrix decomposition algorithm.** First, we define the unitary $M \times M$ Fourier matrix

$$U_M = \frac{1}{\sqrt{M}} \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \bar{\omega} & \bar{\omega}^2 & \cdots & \bar{\omega}^{M-1} \\ 1 & \bar{\omega}^2 & \bar{\omega}^4 & \cdots & \bar{\omega}^{2(M-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \bar{\omega}^{M-1} & \bar{\omega}^{2(M-1)} & \cdots & \bar{\omega}^{(M-1)(M-1)} \end{pmatrix}, \quad \text{where} \quad \omega = \mathrm{e}^{2\pi \mathrm{i}/M},\ \mathrm{i}^2 = -1. \tag{2.16}$$

If I_N is the $N \times N$ identity matrix, pre–multiplication of system (2.12) by $I_N \otimes U_M$ yields

$$(I_N \otimes U_M)\, A\, (I_N \otimes U_M^*)\, (I_N \otimes U_M)\, \boldsymbol{a} = (I_N \otimes U_M)\, \boldsymbol{b} \quad \text{or} \quad \tilde{A}\tilde{\boldsymbol{a}} = \tilde{\boldsymbol{b}}, \tag{2.17}$$

where

$$\begin{aligned} \tilde{A} &= (I_N \otimes U_M)\, A\, (I_N \otimes U_M^*) \\ &= \begin{pmatrix} U_M A_{1,1} U_M^* & U_M A_{1,2} U_M^* & \cdots & U_M A_{1,N} U_M^* \\ U_M A_{2,1} U_M^* & U_M A_{2,2} U_M^* & \cdots & U_M A_{2,N} U_M^* \\ \vdots & \vdots & & \vdots \\ U_M A_{N,1} U_M^* & U_M A_{N,2} U_M^* & \cdots & U_M A_{N,N} U_M^* \end{pmatrix} = \begin{pmatrix} D_{1,1} & D_{1,2} & \cdots & D_{1,N} \\ D_{2,1} & D_{2,2} & \cdots & D_{2,N} \\ \vdots & \vdots & & \vdots \\ D_{N,1} & D_{N,2} & \cdots & D_{N,N} \end{pmatrix}, \end{aligned} \tag{2.18}$$

and

$$\tilde{\boldsymbol{a}} = (I_N \otimes U_M)\, \boldsymbol{a} = \begin{pmatrix} U_M \boldsymbol{a}_1 \\ U_M \boldsymbol{a}_2 \\ \vdots \\ U_M \boldsymbol{a}_N \end{pmatrix} = \begin{pmatrix} \tilde{\boldsymbol{a}}_1 \\ \tilde{\boldsymbol{a}}_2 \\ \vdots \\ \tilde{\boldsymbol{a}}_N \end{pmatrix}, \quad \tilde{\boldsymbol{f}} = (I_N \otimes U_M)\, \boldsymbol{b} = \begin{pmatrix} U_M \boldsymbol{b}_1 \\ U_M \boldsymbol{b}_2 \\ \vdots \\ U_M \boldsymbol{b}_N \end{pmatrix} = \begin{pmatrix} \tilde{\boldsymbol{b}}_1 \\ \tilde{\boldsymbol{b}}_2 \\ \vdots \\ \tilde{\boldsymbol{b}}_N \end{pmatrix}. \tag{2.19}$$

From the properties of circulant matrices [6], each of the $M \times M$ matrices D_{n_1,n_2}, $n_1, n_2 = 1, \cdots, N$, is diagonal. If, in particular

$$D_{n_1,n_2} = \operatorname{diag}\left(D_{n_1,n_{2_1}}, D_{n_1,n_{2_2}}, \ldots, D_{n_1,n_{2_M}}\right) \quad \text{and} \quad A_{n_1,n_2} = \operatorname{circ}\left(A_{n_1,n_{2_1}}, A_{n_1,n_{2_2}} \ldots, A_{n_1,n_{2_M}}\right),$$

we have, for $n_1, n_2 = 1, \cdots, N$,

$$D_{n_1,n_{2_m}} = \sum_{k=1}^{M} A_{n_1,n_{2_k}} \omega^{(k-1)(m-1)},\ m = 1, \cdots, M. \tag{2.20}$$

Since the matrix $\tilde{A}$ consists of N^2 blocks of order M, each of which is diagonal, the solution of system (2.17) can be decomposed into solving the M independent systems of order N

$$E_m\, \boldsymbol{x}_m = \boldsymbol{y}_m, \quad m = 1, \cdots, M, \tag{2.21}$$

where

$$(E_m)_{n_1,n_2} = D_{n_1,n_{2_m}},\ n_1, n_2 = 1, \cdots, N,$$

and

$$(\boldsymbol{x}_m)_n = (\tilde{\boldsymbol{a}}_n)_m,\ (\boldsymbol{y}_m)_n = \left(\tilde{\boldsymbol{b}}_n\right)_m,\ n = 1, \cdots, N. \tag{2.22}$$

Having obtained the vectors $\boldsymbol{x}_m$, $m = 1, \cdots, M$, we can recover the vectors $\tilde{\boldsymbol{a}}_n$, $n = 1, \cdots, N$ and, subsequently, the vector $\boldsymbol{a}$ from (2.19), i.e.

$$\boldsymbol{a} = \begin{pmatrix} \boldsymbol{a}_1 \\ \boldsymbol{a}_2 \\ \vdots \\ \boldsymbol{a}_N \end{pmatrix} = (I_N \otimes U_M^*)\, \tilde{\boldsymbol{a}} = \begin{pmatrix} U_M^* \tilde{\boldsymbol{a}}_1 \\ U_M^* \tilde{\boldsymbol{a}}_2 \\ \vdots \\ U_M^* \tilde{\boldsymbol{a}}_N \end{pmatrix}. \tag{2.23}$$

In conclusion, the MDA can be summarized as follows:

```
Algorithm 1
Step 1:    Compute $\tilde{\boldsymbol{b}}_n$ $\tilde{\boldsymbol{b}}_n = U_M \boldsymbol{b}_n$, $n = 1, \cdots, N$.
Step 2:    Construct the diagonal matrices $D_{n_1,n_2}$ from (2.20).
Step 3:    Solve the $M$, $N \times N$ systems (2.21) to obtain the $\{\boldsymbol{x}_m\}_{m=1}^{M}$,
           and subsequently the $\{\tilde{\boldsymbol{a}}_m\}_{m=1}^{N}$ from (2.22).
Step 4:    Recover the vector of coefficients $\boldsymbol{a}$ from (2.23).
```

In Steps 1, 2 and 4 FFTs are used while the most expensive part of the algorithm is the solution of M linear systems, each of order N. The FFTs are carried out using the MATLAB© [31] commands `fft` and `ifft`. In addition to the savings in computational cost, considerable savings in storage are achieved since only one row of the circulant matrices involved needs to be stored.

3. The Biharmonic equation

3.1. **The problem.** We next consider the biharmonic equation

$$\Delta^2 u = f \quad \text{in} \quad \Omega, \tag{3.1a}$$

subject to the boundary conditions

$$u = g_1 \quad \text{and} \quad \frac{\partial u}{\partial \mathrm{n}} = h_1 \quad \text{on} \quad \partial\Omega_1, \tag{3.1b}$$

$$u = g_2 \quad \text{and} \quad \frac{\partial u}{\partial \mathrm{n}} = h_2 \quad \text{on} \quad \partial\Omega_2, \tag{3.1c}$$

where Ω is the annulus (2.2). This problem is known as the *first biharmonic problem.*

3.2. **Kansa's method.** We approximate the solution u of boundary value problem (3.1) by (2.9).
The coefficients are determined by collocating the differential equation (3.1a) and the boundary conditions (3.1b)-(3.1c) in the following way:

$$\begin{aligned} \Delta^2 u_{MN}(x_{mn}, y_{mn}) &= f(x_{mn}, y_{mn}), \quad m = 1, \ldots, M, \; n = 3, \ldots, N-2, \\ u_{MN}(x_{m1}, y_{m1}) = g_1(x_{m1}, y_{m1}) \quad \text{and} \quad \frac{\partial u}{\partial \mathrm{n}}(x_{m1}, y_{m1}) &= h_1(x_{m1}, y_{m1}), \quad m = 1, \ldots, M, \\ u_{MN}(x_{mN}, y_{mN}) = g_2(x_{mN}, y_{mN}) \quad \text{and} \quad \frac{\partial u}{\partial \mathrm{n}} u_{MN}(x_{mN}, y_{mN}) &= h_2(x_{mN}, y_{mN}), \quad m = 1, \ldots, M, \end{aligned} \tag{3.2}$$

yielding a total of MN equations.
By vectorizing the arrays of unknown coefficients and collocation points via (2.11), equations (3.2) yield a system of the form (2.12). In this case, the $M \times M$ submatrices A_{n_1,n_2}, $n_1, n_2 = 1, \ldots, N$ are defined as follows:

$$\begin{aligned} &\left(A_{n_1,n_2}\right)_{m_1,m_2} = \Delta^2 \phi_{m_2,n_2}(x_{m_1,n_1}, y_{m_1,n_1}), \; m_1, m_2 = 1, \ldots, M, \; n_1 = 3, \ldots, N-2, \; n_2 = 1, \ldots, N, \\ &\left(A_{1,n}\right)_{m_1,m_2} = \phi_{m_2,n}(x_{m_1,1}, y_{m_1,1}), \quad \left(A_{2,n}\right)_{m_1,m_2} = \frac{\partial \phi_{m_2,n}}{\partial \mathrm{n}}(x_{m_1,1}, y_{m_1,1}), \\ &\qquad m_1, m_2 = 1, \ldots, M, \quad n = 1, \ldots, N, \\ &\left(A_{N,n}\right)_{m_1,m_2} = \phi_{m_2,n}(x_{m_1,N}, y_{m_1,N}), \quad \left(A_{N-1,n}\right)_{m_1,m_2} = \frac{\partial \phi_{m_2,n}}{\partial \mathrm{n}}(x_{m_1,N}, y_{m_1,N}), \end{aligned} \tag{3.3}$$

while the $M \times 1$ vectors $\boldsymbol{a}_n, \boldsymbol{b}_n, n = 1, \ldots, N$ are defined as

$$\begin{gathered} \left(\boldsymbol{a}_n\right)_m = a_{mn}, \; m = 1, \ldots, M, \; N = 1, \ldots, N, \\ \left(\boldsymbol{b}_n\right)_m = f(x_{mn}, y_{mn}), \quad m = 1, \ldots, M, \; n = 3, \ldots, N-2, \\ \left(\boldsymbol{b}_1\right)_m = g_1(x_{m1}, y_{m1}), \quad \left(\boldsymbol{b}_2\right)_m = h_1(x_{m1}, y_{m1}), \quad m = 1, \ldots, M, \\ \left(\boldsymbol{b}_N\right)_m = g_2(x_{mN}, y_{mN}), \quad \left(\boldsymbol{b}_{N-1}\right)_m = h_2(x_{mN}, y_{mN}), \quad m = 1, \ldots, M. \end{gathered}$$

3.2.1. *Second biharmonic problem.* Suppose that instead of boundary conditions (3.1b)-(3.1c) we had the boundary conditions

$$u = g_1 \quad \text{and} \quad \Delta u = h_1 \quad \text{on} \quad \partial\Omega_1, \tag{3.4}$$

$$u = g_2 \quad \text{and} \quad \Delta u = h_2 \quad \text{on} \quad \partial\Omega_2. \tag{3.5}$$

This problem is known as the *second biharmonic problem.*
The corresponding submatrices $(A_{2,n})$, $(A_{N-1,n})$, $n = 1, \dots, N$, in (3.3) are now defined by

$$\left(A_{2,n}\right)_{m_1,m_2} = \Delta\phi_{m_2,n}(x_{m_1,1}, y_{m_1,1}), \quad \left(A_{N-1,n}\right)_{m_1,m_2} = \Delta\phi_{m_2,n}(x_{m_1,N}, y_{m_1,N}), \quad m_1, m_2 = 1, \dots, M. \tag{3.6}$$

As shown in the Appendix (Lemma 2), as in the case of the Poisson equation in Section 2.2, each of the $M \times M$ submatrices A_{n_1,n_2}, $n_1, n_2 = 1, \dots, N$, is circulant and hence the matrix A is block circulant. The resulting system can therefore be solved efficiently using the MDA described in Section 2.3.

4. The Cauchy-Navier equations of elasticity

4.1. **The problem.** We finally consider the Cauchy–Navier system for the displacements (u_1, u_2) in the form (see, e.g. [12])

$$\left\{ \begin{aligned} \mathcal{L}_1(u_1, u_2) &\equiv \mu\Delta u_1 + \frac{\mu}{1-2\nu}\left(\frac{\partial^2 u_1}{\partial x^2} + \frac{\partial^2 u_2}{\partial x \partial y}\right) = f_1, \\ \mathcal{L}_2(u_1, u_2) &\equiv \frac{\mu}{1-2\nu}\left(\frac{\partial^2 u_1}{\partial x \partial y} + \frac{\partial^2 u_2}{\partial y^2}\right) + \mu\Delta u_2 = f_2, \end{aligned} \right. \quad \text{in} \quad \Omega, \tag{4.1a}$$

subject to the Dirichlet boundary conditions

$$u_1 = g_1 \quad \text{and} \quad u_2 = h_1 \quad \text{on} \quad \partial\Omega_1, \tag{4.1b}$$

$$u_1 = g_2 \quad \text{and} \quad u_2 = h_2 \quad \text{on} \quad \partial\Omega_2, \tag{4.1c}$$

where Ω in the annulus (2.2). In (4.1a) the constant $\nu \in [0, 1/2)$ is Poisson's ratio and $\mu > 0$ is the shear modulus.

4.2. **Kansa's method.** We approximate the solution (u_1, u_2) of boundary value problem (3.1) by $(u_{MN}^{(1)}, u_{MN}^{(2)})$ where

$$u_{MN}^{(\ell)}(x, y) = \sum_{m=1}^{M} \sum_{n=1}^{N} a_{mn}^{(\ell)} \phi_{mn}(x, y), \ \ell = 1, 2, \qquad (x, y) \in \bar{\Omega}, \tag{4.2}$$

and the $2MN$ coefficients $\left\{(a_{mn}^{(\ell)})\right\}_{m=1,n=1}^{M,N}$, $\ell = 1, 2$, are unknown.
The unknown coefficients are determined by collocating the differential equations (4.1a) and the boundary conditions (4.1b)-(4.1c) in the following way:

$$\begin{gathered} \mathcal{L}_\ell(u_{MN}^{(1)}, u_{MN}^{(2)})(x_{mn}, y_{mn}) = f_\ell(x_{mn}, y_{mn}), \ \ell = 1, 2, \quad m = 1, \dots, M, \ n = 2, \dots, N-1, \\ u_{MN}^{(1)}(x_{m1}, y_{m1}) = g_1(x_{m1}, y_{m1}) \ \text{ and } \ u_{MN}^{(2)}(x_{m1}, y_{m1}) = h_1(x_{m1}, y_{m1}), \quad m = 1, \dots, M, \\ u_{MN}^{(1)}(x_{mN}, y_{mN}) = g_2(x_{mN}, y_{mN}) \ \text{ and } \ u_{MN}^{(2)}(x_{mN}, y_{mN}) = h_2(x_{mN}, y_{mN}), \quad m = 1, \dots, M, \end{gathered} \tag{4.3}$$

yielding a total of $2MN$ equations.
By vectorizing the arrays of unknown coefficients and collocation points as in (2.11), equations (4.3) yield a $2MN \times 2MN$ system of the form (2.12). The $2M \times 2M$ submatrices A_{n_1,n_2}, $n_1, n_2 = 1, \dots, N$ are now defined as follows: (Note that we are now defining the matrix and vector elements in (2.12) as 2×2 and 2×1 *arrays*, respectively.)

$$\left(A_{n_1,n_2}\right)_{m_1,m_2} =$$

$$\begin{pmatrix} \mu\Delta\phi_{m_2,n_2}(x_{m_1,n_1},y_{m_1,n_1}) + \dfrac{\mu}{1-2\nu}\dfrac{\partial^2\phi_{m_2,n_2}}{\partial x^2}(x_{m_1,n_1},y_{m_1,n_1}) & \dfrac{\mu}{1-2\nu}\dfrac{\partial^2\phi_{m_2,n_2}}{\partial x\partial y}(x_{m_1,n_1},y_{m_1,n_1}) \\ \dfrac{\mu}{1-2\nu}\dfrac{\partial^2\phi_{m_2,n_2}}{\partial x\partial y}(x_{m_1,n_1},y_{m_1,n_1}) & \mu\Delta\phi_{m_2,n_2}(x_{m_1,n_1},y_{m_1,n_1}) + \dfrac{\mu}{1-2\nu}\dfrac{\partial^2\phi_{m_2,n_2}}{\partial y^2}(x_{m_1,n_1},y_{m_1,n_1}) \end{pmatrix},$$

$$m_1, m_2 = 1,\dots,M,\ n_1 = 2,\dots,N-1,\ n_2 = 1,\dots,N, \tag{4.4a}$$

$$\left(A_{1,n}\right)_{m_1,m_2} = \begin{pmatrix} \phi_{m_2,n}(x_{m_1,1},y_{m_1,1}) & 0 \\ 0 & \phi_{m_2,n}(x_{m_1,1},y_{m_1,1}) \end{pmatrix}, \tag{4.4b}$$

$$m_1, m_2 = 1,\dots,M,\ n = 1,\dots,N,$$

$$\left(A_{N,n}\right)_{m_1,m_2} = \begin{pmatrix} \phi_{m_2,n}(x_{m_1,N},y_{m_1,N}) & 0 \\ 0 & \phi_{m_2,n}(x_{m_1,N},y_{m_1,N}) \end{pmatrix}, \tag{4.4c}$$

while the $2M \times 1$ vectors $\boldsymbol{a}_n, \boldsymbol{b}_n, n = 1,\dots,N$ are defined as

$$\left(\boldsymbol{a}_n\right)_m = \begin{pmatrix} a_{mn}^{(1)} \\ a_{mn}^{(2)} \end{pmatrix},\ m = 1,\dots,M,\ n = 1,\dots,N,$$

$$\left(\boldsymbol{b}_n\right)_m = \begin{pmatrix} f_1(x_{mn},y_{mn}) \\ f_2(x_{mn},y_{mn}) \end{pmatrix}, \quad m = 1,\dots,M,\ n = 2,\dots,N-1,$$

$$\left(\boldsymbol{b}_1\right)_m = \begin{pmatrix} g_1(x_{m1},y_{m1}) \\ h_1(x_{m1},y_{m1}) \end{pmatrix}, \quad \left(\boldsymbol{b}_N\right)_m = \begin{pmatrix} g_2(x_{mN},y_{mN}) \\ h_2(x_{mN},y_{mN}) \end{pmatrix}, \quad m = 1,\dots,M.$$

4.2.1. *Neumann boundary conditions.* Suppose that instead of the Dirichlet boundary conditions (4.1b) we had the Neumann boundary conditions

$$t_1 = g_1 \quad \text{and} \quad t_2 = h_1 \quad \text{on} \quad \partial\Omega_1, \tag{4.5}$$

where (t_1, t_2) are the tractions defined by ([12])

$$t_1 = 2\mu\left[\left(\frac{1-\nu}{1-2\nu}\right)\frac{\partial u_1}{\partial x} + \left(\frac{\nu}{1-2\nu}\right)\frac{\partial u_2}{\partial y}\right]\mathrm{n}_x + \mu\left[\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x}\right]\mathrm{n}_y,$$

$$t_2 = \mu\left[\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x}\right]\mathrm{n}_x + 2\mu\left[\left(\frac{\nu}{1-2\nu}\right)\frac{\partial u_1}{\partial x} + \left(\frac{1-\nu}{1-2\nu}\right)\frac{\partial u_2}{\partial y}\right]\mathrm{n}_y.$$

In this case, we have, instead of (4.4b), that the submatrices $\left(A_{1,n}\right)_{m_1,m_2}, m_1, m_2 = 1,\dots,M,\ n = 1,\dots,N$, are defined by

$$\left(A_{1,n}\right)_{m_1,m_2} = \mu\begin{pmatrix} 2\left(\dfrac{1-\nu}{1-2\nu}\right)\dfrac{\partial\phi_{m_2,n}}{\partial x}\mathrm{n}_x + \dfrac{\partial\phi_{m_2,n}}{\partial y}\mathrm{n}_y & \left(\dfrac{2\nu}{1-2\nu}\right)\dfrac{\partial\phi_{m_2,n}}{\partial y}\mathrm{n}_x + \dfrac{\partial\phi_{m_2,n}}{\partial x}\mathrm{n}_y \\ \dfrac{\partial\phi_{m_2,n}}{\partial y}\mathrm{n}_x + \left(\dfrac{2\nu}{1-2\nu}\right)\dfrac{\partial\phi_{m_2,n}}{\partial x}\mathrm{n}_y & \dfrac{\partial\phi_{m_2,n}}{\partial x}\mathrm{n}_x + 2\left(\dfrac{1-\nu}{1-2\nu}\right)\dfrac{\partial\phi_{m_2,n}}{\partial y}\mathrm{n}_y \end{pmatrix} \tag{4.6}$$

$$= \mu\left(\frac{\partial\phi_{m_2,n}}{\partial x}\mathrm{n}_x + \frac{\partial\phi_{m_2,n}}{\partial y}\mathrm{n}_y\right)I_2 + \frac{\mu}{1-2\nu}\begin{pmatrix} \dfrac{\partial\phi_{m_2,n}}{\partial x}\mathrm{n}_x & 2\nu\dfrac{\partial\phi_{m_2,n}}{\partial y}\mathrm{n}_x + (1-2\nu)\dfrac{\partial\phi_{m_2,n}}{\partial x}\mathrm{n}_y \\ (1-2\nu)\dfrac{\partial\phi_{m_2,n}}{\partial y}\mathrm{n}_x + 2\nu\dfrac{\partial\phi_{m_2,n}}{\partial x}\mathrm{n}_y & \dfrac{\partial\phi_{m_2,n}}{\partial y}\mathrm{n}_y \end{pmatrix},$$

with all the quantities in (4.6) evaluated at the boundary point $(x_{m_1,1}, y_{m_1,1})$.

In contrast to the Poisson and biharmonic cases, matrix A in (2.12) does not possess a block circulant structure. However, as described in the context of the MFS, in e.g., [22, 23, 24] a block circulant structure can be obtained by means of a simple transformation.

4.3. **Matrix decomposition algorithm.** We introduce the $2M \times 2M$ matrix

$$R = \left(\begin{array}{c|c|c|c|c|c} R_{\vartheta_1} & 0 & 0 & \cdots & 0 & 0 \\ \hline 0 & R_{\vartheta_2} & 0 & \cdots & 0 & 0 \\ \hline \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \hline 0 & 0 & 0 & \cdots & R_{\vartheta_{M-1}} & 0 \\ \hline 0 & 0 & 0 & \cdots & 0 & R_{\vartheta_M} \end{array} \right), \tag{4.7}$$

where

$$R_{\vartheta_k} = \begin{pmatrix} \cos\vartheta_k & \sin\vartheta_k \\ \sin\vartheta_k & -\cos\vartheta_k \end{pmatrix}, \quad \vartheta_k = \frac{2\pi(k-1)}{M}.$$

Since clearly $R_{\vartheta_k}^2 = I_2$ then $R^2 = I_{2N}$.
We premultiply the $2MN \times 2MN$ system (2.12) ($A\boldsymbol{a} = \boldsymbol{b}$) by the $2MN \times 2MN$ matrix $I_N \otimes R$ to get

$$(I_N \otimes R)\, A\boldsymbol{a} = (I_N \otimes R)\, \boldsymbol{b} \quad \text{or} \quad \tilde{A}\tilde{\boldsymbol{a}} = \tilde{\boldsymbol{b}}, \tag{4.8}$$

where

$$\tilde{A} = (I_N \otimes R)\, A\, (I_N \otimes R), \quad \tilde{\boldsymbol{a}} = (I_N \otimes R)\, \boldsymbol{a}, \quad \tilde{\boldsymbol{b}} = (I_N \otimes R)\, \boldsymbol{b}.$$

The $2MN \times 2MN$ matrix $\tilde{A}$ can be written as

$$\tilde{A} = \begin{pmatrix} \tilde{A}_{1,1} & \tilde{A}_{1,2} & \ldots & \tilde{A}_{1,N} \\ \tilde{A}_{2,1} & \tilde{A}_{2,2} & \ldots & \tilde{A}_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{A}_{N,1} & \tilde{A}_{N,2} & \ldots & \tilde{A}_{N,N} \end{pmatrix}, \tag{4.9}$$

where each of the $2M \times 2M$ submatrices $\tilde{A}_{m,\ell} = RA_{m,\ell}R$.

Moreover, keeping in mind that the elements $\left(\tilde{A}_{n_1,n_2}\right)_{m_1,m_2} = \left(\left(\tilde{A}_{n_1,n_2}\right)_{m_1,m_2}\right)_{i,j=1}^2$ are 2×2 arrays, we have

$$\left(\tilde{A}_{n_1,n_2}\right)_{m_1,m_2} = R_{\vartheta_{m_1}} \left(A_{n_1,n_2}\right)_{m_1,m_2} R_{\vartheta_{m_2}}, \quad m_1, m_2 = 1, \ldots, M,\ n_1, n_2 = 1, \ldots, N. \tag{4.10}$$

Further, as proved in the Appendix (Lemma 3), each submatrix $\tilde{A}_{n_1,n_2}, n_1, n_2 = 1, \ldots, N$, has a block 2×2 block circulant structure. The $2MN \times 1$ vectors $\tilde{\boldsymbol{a}}, \tilde{\boldsymbol{b}}$ are written as

$$\tilde{\boldsymbol{a}} = \begin{pmatrix} \tilde{\boldsymbol{a}}_1 \\ \tilde{\boldsymbol{a}}_2 \\ \vdots \\ \tilde{\boldsymbol{a}}_N \end{pmatrix}, \quad \tilde{\boldsymbol{b}} = \begin{pmatrix} \tilde{\boldsymbol{b}}_1 \\ \tilde{\boldsymbol{b}}_2 \\ \vdots \\ \tilde{\boldsymbol{b}}_N \end{pmatrix},$$

where the $2M \times 1$ subvectors $\tilde{\boldsymbol{a}}_n, \tilde{\boldsymbol{b}}_n, n = 1, \ldots N$, are defined by $\tilde{\boldsymbol{a}}_n = R\boldsymbol{a}_n, \tilde{\boldsymbol{b}}_n = R\boldsymbol{b}_n$ and the 2×1 subvectors $\left(\left(\tilde{\boldsymbol{a}}_n\right)_m\right)_{i=1}^2$, $\left(\left(\tilde{\boldsymbol{b}}_n\right)_m\right)_{i=1}^2$, $m = 1, \ldots, M$, are defined by

$$\left(\tilde{\boldsymbol{a}}_n\right)_m = R_{\vartheta_m} \left(\boldsymbol{a}_n\right)_m, \ \left(\tilde{\boldsymbol{b}}_n\right)_m = R_{\vartheta_m} \left(\boldsymbol{b}_n\right)_m.$$

We next rewrite system (4.8) in the form

$$\left(\begin{array}{c|c} B_{11} & B_{12} \\ \hline B_{21} & B_{22} \end{array} \right) \left(\begin{array}{c} \boldsymbol{c}_1 \\ \hline \boldsymbol{c}_2 \end{array} \right) = \left(\begin{array}{c} \boldsymbol{d}_1 \\ \hline \boldsymbol{d}_2 \end{array} \right), \tag{4.11}$$

where the $MN \times MN$ matrices $B_{ij}, i, j = 1, 2$, are expressed in the form

$$B_{ij} = \begin{pmatrix} \tilde{B}^{ij}_{1,1} & \tilde{B}^{ij}_{1,2} & \cdots & \tilde{B}^{ij}_{1,N} \\ \tilde{B}^{ij}_{2,1} & \tilde{B}^{ij}_{2,2} & \cdots & \tilde{B}^{ij}_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{B}^{ij}_{N,1} & \tilde{B}^{ij}_{N,2} & \cdots & \tilde{B}^{ij}_{N,N} \end{pmatrix}.$$

Each $M \times M$ submatrix $\tilde{B}^{ij}_{n_1,n_2}, i, j = 1, 2,\ n_1, n_2 = 1, \dots, N$, is circulant and defined from

$$\left(\tilde{B}^{ij}_{n_1,n_2}\right)_{m_1,m_2} = \left(\left(\tilde{A}_{n_1,n_2}\right)_{m_1,m_2}\right)_{i,j}, \quad m_1, m_2 = 1, \dots, M. \tag{4.12}$$

Also, the $MN \times 1$ vectors $\boldsymbol{c}_i, \boldsymbol{d}_i, i = 1, 2$, are defined from

$$\boldsymbol{c}_i = \begin{pmatrix} \tilde{\boldsymbol{c}}^i_1 \\ \tilde{\boldsymbol{c}}^i_2 \\ \vdots \\ \tilde{\boldsymbol{c}}^i_N \end{pmatrix}, \quad \boldsymbol{d}_i = \begin{pmatrix} \tilde{\boldsymbol{d}}^i_1 \\ \tilde{\boldsymbol{d}}^i_2 \\ \vdots \\ \tilde{\boldsymbol{d}}^i_N \end{pmatrix},$$

where

$$\left(\tilde{c}^i_n\right)_m = \left((\tilde{a}_n)_m\right)_i, \quad \left(\tilde{d}^i_n\right)_m = \left(\left(\tilde{b}_n\right)_m\right)_i, \ m = 1, \dots, M. \tag{4.13}$$

We premultiply system (4.11) by the matrix $I_2 \otimes I_N \otimes U_M$ to get

$$\left(I_2 \otimes I_N \otimes U_M\right) \left(\begin{array}{c|c} B_{11} & B_{12} \\ \hline B_{21} & B_{22} \end{array}\right) \left(I_2 \otimes I_N \otimes U_M^*\right) \left(I_2 \otimes I_N \otimes U_M\right) \left(\frac{\boldsymbol{c}_1}{\boldsymbol{c}_2}\right) = \left(I_2 \otimes I_N \otimes U_M\right) \left(\frac{\boldsymbol{d}_1}{\boldsymbol{d}_2}\right), \tag{4.14}$$

or

$$\left(\begin{array}{c|c} \tilde{B}_{11} & \tilde{B}_{12} \\ \hline \tilde{B}_{21} & \tilde{B}_{22} \end{array}\right) \left(\frac{\boldsymbol{p}_1}{\boldsymbol{p}_2}\right) = \left(\frac{\boldsymbol{q}_1}{\boldsymbol{q}_2}\right), \tag{4.15}$$

where

$$\left(\frac{\boldsymbol{p}_1}{\boldsymbol{p}_2}\right) = \left(I_2 \otimes I_N \otimes U_M\right) \left(\frac{\boldsymbol{c}_1}{\boldsymbol{c}_2}\right), \quad \left(\frac{\boldsymbol{q}_1}{\boldsymbol{q}_2}\right) = \left(I_2 \otimes I_N \otimes U_M\right) \left(\frac{\boldsymbol{d}_1}{\boldsymbol{d}_2}\right),$$

where for $i = 1, 2$

$$\boldsymbol{p}_i = \begin{pmatrix} \tilde{\boldsymbol{p}}^i_1 \\ \tilde{\boldsymbol{p}}^i_2 \\ \vdots \\ \tilde{\boldsymbol{p}}^i_N \end{pmatrix}, \quad \boldsymbol{q}_i = \begin{pmatrix} \tilde{\boldsymbol{q}}^i_1 \\ \tilde{\boldsymbol{q}}^i_2 \\ \vdots \\ \tilde{\boldsymbol{q}}^i_N \end{pmatrix}, \quad \text{with} \quad \tilde{\boldsymbol{p}}^i_n = U_M \tilde{\boldsymbol{c}}^i_n,\ \tilde{\boldsymbol{q}}^i_n = U_M \tilde{\boldsymbol{d}}^i_n,\ n = 1, \dots, N.$$

The matrices $\tilde{B}_{ij}, i, j = 1, 2$ are given from

$$\tilde{B}_{ij} = \left(I_N \otimes U_M\right) B_{ij} \left(I_N \otimes U_M^*\right),$$

and since each of the matrices $B_{ij}, i, j = 1, 2$ is block circulant, from (2.18) it follows that

$$\tilde{B}_{ij} = \begin{pmatrix} D^{ij}_{1,1} & D^{ij}_{1,2} & \cdots & D^{ij}_{1,N} \\ D_{2,1} & D^{ij}_{2,2} & \cdots & D^{ij}_{2,N} \\ \vdots & \vdots & & \vdots \\ D^{ij}_{N,1} & D^{ij}_{N,2} & \cdots & D^{ij}_{N,N} \end{pmatrix}, \tag{4.16}$$

where each $M \times M$ matrix $D^{ij}_{n_1,n_2}, n_1, n_2 = 1, \dots, N$, is diagonal.

More specifically, if

$$D^{ij}_{n_1,n_2} = \operatorname{diag}\left(D^{ij}_{n_1,n_{2_1}}, D^{ij}_{n_1,n_{2_2}}, \ldots, D^{ij}_{n_1,n_{2_M}}\right) \quad \text{and} \quad \tilde{B}^{ij}_{n_1,n_2} = \operatorname{circ}\left(\tilde{B}^{ij}_{n_1,n_{2_1}}, \tilde{B}^{ij}_{n_1,n_{2_M}} \ldots, \tilde{B}^{ij}_{n_1,n_{2_M}}\right),$$

we have, for $n_1, n_2 = 1, \cdots, N$,

$$D^{ij}_{n_1,n_{2_m}} = \sum_{k=1}^{M} \tilde{B}^{ij}_{n_1,n_{2_k}} \omega^{(k-1)(m-1)}, \ m = 1, \cdots, M. \tag{4.17}$$

Since each matrix $\tilde{B}_{ij}, i, j = 1, 2$, consists of N^2 blocks of order M each of which is diagonal, the solution of system (4.15) can be decomposed into solving the M systems of order $2N$

$$\left(\begin{array}{c|c} E^m_{11} & E^m_{12} \\ \hline E^m_{21} & E^m_{22} \end{array}\right) \left(\begin{array}{c} \boldsymbol{x}^m_1 \\ \hline \boldsymbol{x}^m_2 \end{array}\right) = \left(\begin{array}{c} \boldsymbol{y}^m_1 \\ \hline \boldsymbol{y}^m_2 \end{array}\right), \quad m = 1, \cdots, M, \tag{4.18}$$

where

$$\left(E^m_{ij}\right)_{n_1,n_2} = D^{ij}_{n_1,n_{2_m}}, \ n_1, n_2 = 1, \cdots, N$$

and

$$(\boldsymbol{x}^m_i)_n = \left(\tilde{\boldsymbol{p}}^i_n\right)_m, \ (\boldsymbol{y}^m_i)_n = \left(\tilde{\boldsymbol{q}}^i_n\right)_m, \ \ n = 1, \cdots, N. \tag{4.19}$$

From the vectors $\boldsymbol{x}^m_i, i = 1, 2; m = 1, \ldots, M$ we can obtain the vectors p_1, p_2 and the vectors c_1, c_2, and subsequently the vector $\tilde{\boldsymbol{a}}$, before finally obtaining the vector $\boldsymbol{a}$.

The MDA, in this case, can be summarized as follows:

```
Algorithm 2
Step 1:   Compute  $\tilde{\boldsymbol{b}} = (I_N \otimes R)\boldsymbol{b}$.
Step 2:   Calculate the $2 \times 2$ arrays $\left(\tilde{A}_{n_1,n_2}\right)_{1,m_2}$.
Step 3:   Compute  $\tilde{\boldsymbol{q}}^i_n = U_M \tilde{\boldsymbol{d}}^i_n$, $n = 1, \ldots, N$ and hence $\boldsymbol{y}^m_i$, $m = 1, \ldots, N$ from (4.19).
Step 4:   Construct the diagonal matrices $D^{ij}_{n_1,n_2}$ from (4.17) and hence matrices $E^m_{ij}$ in (4.18).
Step 5:   Solve the $M$, $2N \times 2N$ systems (4.18) to obtain the $\boldsymbol{x}^m_i, i = 1, 2; m = 1, \ldots, M$,
          and subsequently the vectors $\boldsymbol{p}_i, i = 1, 2$.
Step 6:   Recover the vectors $\boldsymbol{c}_i, i = 1, 2$ from $\tilde{\boldsymbol{c}}^i_n = U^*_M \tilde{\boldsymbol{p}}^i_n, n = 1, \ldots, N$.
Step 7:   Reorder vectors $\boldsymbol{c}_i, i = 1, 2$ to obtain vector $\tilde{\boldsymbol{a}}$.
Step 8:   Compute  $\boldsymbol{a} = (I_N \otimes R)\tilde{\boldsymbol{a}}$.
```

In Steps 3, 4 and 6 FFTs are used while the most expensive part of the algorithm is the solution of M linear systems, each of order $2N$. Again, substantial savings in storage are obtained as only the first line of the circulant matrices involved needs to be constructed and stored.

5. Numerical examples

In all numerical examples considered, we took collocation points described by $\alpha_n = (-1)^n/4, n = 1, \ldots, N$ (cf. (2.8)). The inner and outer radii of the annular domain Ω are $\varrho_1 = 0.3, \varrho_2 = 1$, respectively. We calculated the maximum relative error E over $\mathcal{M}\mathcal{N}$ test points in $\overline{\Omega}$ defined by

$$\mathrm{r}_n\left(\cos\vartheta_m, \sin\vartheta_m\right), \text{ where } \vartheta_m = \frac{2\pi(m-1)}{\mathcal{M}}, \ m = 1, \ldots, \mathcal{M}, \ \mathrm{r}_n = \varrho_1 + (\varrho_2 - \varrho_1)\,\frac{n-1}{\mathcal{N}-1}, \ n = 1, \ldots, \mathcal{N}. \tag{5.1}$$

Unless otherwise stated, we choose $\mathcal{N} = 25, \mathcal{M} = 50$ so that the test points are different than the collocation points. The maximum relative error E is defined as

$$E = \frac{||u - u_N||_{\infty,\overline{\Omega}}}{||u||_{\infty,\overline{\Omega}}}. \tag{5.2}$$

The proper selection of an RBF is crucial in obtaining good accuracy. In the first example, we test various commonly used RBFs and then choose the best one to perform the tests for the remaining examples. The determination of a suitable shape parameter remains a major issue in the many applications of RBFs involving such parameters. As we have mentioned earlier, numerous techniques have been proposed for the selection of an appropriate shape parameter. In this work, we apply the so-called LOOCV (*leave-one-out cross validation*) algorithm proposed by Rippa [32] for the identification of a suitable shape parameter. The MATLAB© codes for LOOCV can be found in [9]. The MATLAB© function `fminbnd` is used to find the minimum of a function of one variable within a fixed interval. Since `fminbnd` is a local minimizer, an initial guess of the lower and upper bounds denoted by `min` and `max` needs to be provided so that the search takes place in the interval [`min`, `max`]. One of the attractive features of using LOOCV is that we do not need to know the exact solution of the given problem for the selection of a (sub-optimal) shape parameter. Typically, in the problems considered, there are M systems of equations of order N to be solved, see, e.g., (2.21). It is clearly not cost-effective to apply the LOOCV to each of the M systems and determine the sub-optimal shape parameter for each. We therefore only apply the LOOCV to a randomly chosen system among the M systems. This adds $O(N^3)$ computational complexity to the proposed technique. The sub-optimal shape parameter thus obtained is then used for all the other systems. This technique, as shown by the results, appears to be working well.

All numerical computations were carried out on a MATLAB© 2010a platform in OS Windows 7 (32 bit) with Intel Core(TM) i5 2.4GHz CPU and 4 GB memory.

We have considered the following numerical examples.

5.1. **Example 1.** We consider the Poisson equation (2.1a) with a Neumann boundary condition prescribed on $\partial\Omega_1$ and a Dirichlet boundary condition prescribed on $\partial\Omega_2$. The boundary conditions correspond to the exact solution which is given by $u = \mathrm{e}^{x+y}$.

We first perform some tests with the normalized multiquadric (MQ) basis functions

$$\phi_j(x,y) = \Phi(r_j) = \sqrt{(cr_j)^2+1}, \quad r_j^2 = (x-x_j)^2 + (y-y_j)^2,$$

where c is the *shape parameter*. In Figure 2, we present the maximum relative error in u versus the shape parameter c for the case when $M = 128, N = 64$ using the normalized MQ basis functions. From this figure, we can observe that the optimal shape parameter is $c = 5.208$ and the corresponding error is $2.940(-8)$. The results in Table 1 are obtained using the LOOCV algorithm for various search intervals [`min`, `max`]. When comparing the results in Figure 2 and Table 1, we can see that the (sub-optimal) shape parameter results obtained for various search intervals are very stable and satisfactorily close to the optimal shape parameter. In terms of accuracy, the errors obtained using the LOOCV algorithm are comparable to the ones obtained using the optimal shape parameter. Moreover, in order to obtain the optimal shape parameter in Figure 2, one needs to know the exact solution. In contrast, to obtain the sub-optimal shape parameter using the LOOCV algorithm in Table 1, it is not necessary to know the exact solution *a priori*. The results obtained for the Dirichlet problem (2.1) are very similar.

In Table 2, we present the results obtained using various numbers of collocation points $M = N$. It can be seen that the sub-optimal shape parameter becomes larger when the density of the collocation points increases.

In Figure 3 we present the error convergence plots for $(M,N) = (16,8), (32,16), (64,32)$ and $(128,64)$ for varying c. From these plots we observe that as the number of degrees of freedom increases, the accuracy improves.

We also compared the performance of the proposed MDA Kansa-RBF versus the full Kansa-RBF solution in which the full system (2.12) is solved. In Table 3, we present the errors E, CPU times and condition numbers for the full Kansa-RBF and the corresponding quantities using the proposed MDA Kansa-RBF for various numbers of degrees of freedom, using LOOCV. The most time consuming part in both approaches is the search of a suitable shape parameter using LOOCV, in which systems are solved repeatedly. This makes the advantage of the MDA Kansa-RBF even more pronounced as, for example, the solution of the full Kansa system for $M = N = 50$ once requires 2.82 seconds while with LOOCV the method requires 55.68 seconds. Clearly, the use of LOOCV with the

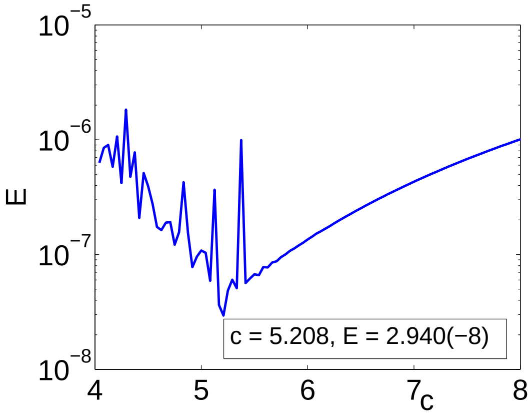

FIGURE 2. Example 1: Maximum relative error versus shape parameter with $M = 128, N = 64$.

[min, max]	sub-optimal c	E
$[0, 8]$	5.013	8.790(−8)
$[0, 10]$	5.005	6.706(−8)
$[0, 12]$	5.078	3.432(−8)
$[0, 14]$	5.001	6.062(−8)
$[0, 16]$	5.002	8.902(−8)
$[0, 18]$	5.006	2.361(−7)
$[0, 20]$	5.007	6.516(−8)

TABLE 1. Example 1: Sub-optimal shape parameters and the corresponding maximum relative errors for various search intervals using fminbnd with $M = 128, N = 64$.

$M = N$	sub-optimal c	E
80	4.714	9.316(−7)
100	6.057	3.361(−7)
120	7.499	2.247(−7)
140	8.896	3.045(−7)
160	10.445	1.949(−7)

TABLE 2. Example 1: Sub-optimal shape parameter and the maximum relative errors for various $M = N$ with initial search interval [0, 20].

MDA Kansa approach does not slow the process as considerably. Note that for $M = N = 100$ and 150 the size of the full Kansa-RBF matrices becomes prohibitive. From the values of the condition numbers in the two approaches we may infer that the *Principle of Uncertainty* [32,33] is more pronounced for the full Kansa-RBF than the MDA Kansa-RBF. The sub-optimal shape parameters are not given in Table 3 since the focus is on the CPU time and accuracy. Similar results were observed for the other examples considered.

Next, we examine and compare the performance of different RBFs using the proposed MDA. The Matérn RBF [29, 30]

$$\Phi(r) = (rc)^n K_n(rc),$$

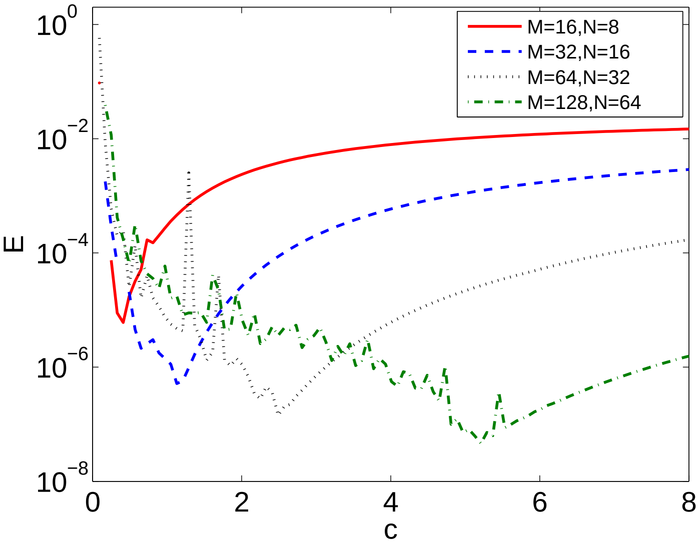

FIGURE 3. Example 1: Maximum relative error versus shape parameter with different numbers of degrees of freedom.

	Full Kansa			MDA Kansa		
$M = N$	E	CPU(sec)	condition #	E	CPU(sec)	condition #
20	5.066(-5)	1.062	1.37(+19)	2.558(-4)	0.143	6.65(+5)
30	4.032(-5)	5.410	6.94(+19)	4.062(-6)	0.309	1.03(+13)
40	2.683(-5)	19.124	2.53(+20)	2.069(-6)	0.638	1.75(+15)
50	7.163(-6)	55.684	4.29(+21)	7.223(-6)	0.947	8.24(+13)
60	3.253(-6)	214.481	8.21(+20)	1.985(-6)	1.608	7.93(+15)
100	–	–	–	4.951(-7)	6.087	1.40(+17)
150	–	–	–	2.672(-6)	21.053	1.03(+18)

TABLE 3. Example 1: Comparison of CPU times and E for full Kansa RBF and MDA Kansa RBF solutions for various $M = N$ with initial search interval [0, 8].

where $n \in \mathbb{Z}, K_n$ is the modified Bessel function of second kind with order n and c is the shape parameter, is known as a highly effective basis function. In [30], the Kansa method using the Matérn RBF was applied to solve Poisson's equation. For large-scale problems, domain decomposition was applied to decompose the domain into smaller domains so that the Kansa method can be applied. In this work, instead of domain decomposition, we use matrix decomposition to handle large-scale problems. For (2.1a) with Dirichlet boundary conditions, the profiles of the maximum relative error versus the shape parameter for the Matérn RBF of orders $n = 4, 5, 6,$ and 7, with $M = 128, N = 64$, are shown in Figure 4. As can be observed, as the order increases so does the value of the optimal shape parameter. Furthermore, the stability of the curve deteriorates as the order becomes higher due to an increase in the higher condition number. As shown in Table 4, when using the Matérn of order 6, the results for various search intervals are compatible with the results in Table 1. Since the value of the optimal shape parameter increases with the order of the Matérn RBF, we use different search intervals for different orders. In Table 5, we present the sub-optimal shape parameters and the corresponding errors for various orders of the Matérn RBF. The accuracy obtained using the Matérn RBF of various orders and the LOOCV algorithm is very close to the results obtained using the actual optimal shape parameters as shown in Figure 4. Even though the results are highly accurate, the drawback of using the Matérn RBF is the high cost of computation due to the presence of the special function $K_n(cr)$.

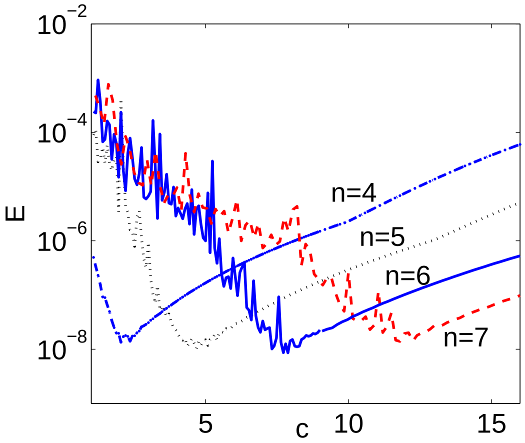

FIGURE 4. Example 1: Maximum relative error versus shape parameter for various orders of the Matérn function with $M = 128, N = 64$.

[min, max]	sub-optimal c	E
$[0, 12]$	9.048	2.026(−8)
$[0, 14]$	9.055	2.246(−8)
$[0, 16]$	9.037	2.010(−8)
$[0, 18]$	9.020	2.335(−8)
$[0, 20]$	9.026	1.985(−8)
$[0, 22]$	8.979	2.046(−8)

TABLE 4. Example 1: Sub-optimal shape parameters and the corresponding maximum relative errors for various search intervals using the Matérn function of order 6 with $M = 128, N = 64$.

order	sub-optimal c	E
4	1.006	8.123(−7)
5	3.735	2.888(−8)
6	9.049	2.300(−8)
7	13.854	2.888(−8)
8	18.466	4.953(−8)

TABLE 5. Example 1: Sub-optimal shape parameter and the maximum relative errors using various orders of the Matérn function with $M = 128$ and $N = 64$.

The Gaussian RBF

$$\Phi(r) = \mathrm{e}^{-cr^2}, \text{ where } c \text{ is the shape parameter,}$$

is another popular and widely used RBF. As shown in Figure 5, we see a very different profile of the shape parameter versus the maximum relative error, for problem (2.1). The errors are fluctuating between 10^{-4} and 10^{-6}. Despite this irregularity, we obtain excellent results using the LOOCV algorithm, as shown in Table 6. We note that all the local minima in Figure 5 have similar accuracy. The MATLAB© function `fminbnd` is capable of finding the local minima and, hence, as we can see in Table 6, for different search intervals we may obtain quite different sub-optimal shape parameters and yet the obtained accuracy is very satisfactory. In conclusion, however, it is preferable to use one of the previous two RBFs which are more predictable in finding a suitable shape parameter.

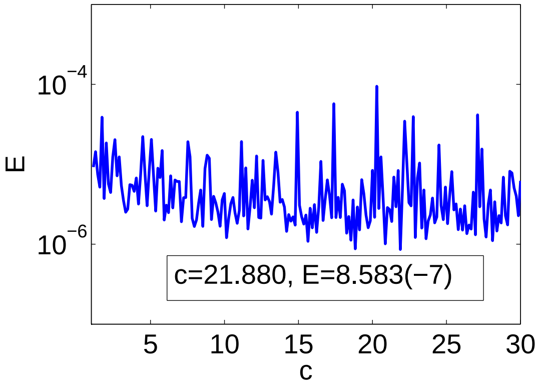

FIGURE 5. Example 1: Maximum relative error versus shape parameter for Gaussian RBF with $M = 128, N = 64$.

[min, max]	sub-optimal c	E
$[0, 6]$	5.657	2.319(−6)
$[0, 8]$	3.028	1.597(−6)
$[0, 10]$	6.176	2.340(−6)
$[0, 12]$	10.493	9.629(−7)
$[0, 14]$	9.076	9.728(−7)
$[0, 16]$	12.262	6.537(−7)
$[0, 18]$	12.480	1.000(−6)
$[0, 20]$	13.055	1.988(−6)
$[0, 22]$	11.603	1.103(−6)

TABLE 6. Example 1: Sub-optimal shape parameters and the corresponding maximum relative errors for various search intervals using the Gaussian RBF with $M = 128, N = 64$.

Finally, we tested the performance of the inverse multiquadric (IMQ) RBF

$$\Phi(r) = \frac{1}{\sqrt{1+(cr)^2}}, \text{ where } c \text{ is the shape parameter,}$$

for the same example. The results obtained were very similar to those obtained using the normalized MQ RBF but the best accuracy recorder in this case was slightly worse at around 10^{-7}.

Among all the RBFs tested in this example, we conclude that the normalized MQ is the best in terms of efficiency, stability, and accuracy. As a result, we will only consider the normalized MQ as an RBF for the next two examples.

Remark. We also considered a distribution of the collocation points which renders their concentration near the boundaries denser (see, e.g., [10]). In particular, instead the of the N uniformly distributed radii defined by (2.7) we used the N Chebyshev-Gauss-Lobatto points on the interval $[\varrho_1, \varrho_2]$, defined by

$$\mathsf{r}_n = \frac{1}{2}\left(\varrho_1 + \varrho_2 + (\varrho_1 - \varrho_2)\cos(\frac{n\pi}{N})\right), \quad n = 1, \ldots, N, \tag{5.3}$$

while the angles are still defined by (2.6) and the collocation points by (2.8). Extensive experimentation revealed no significant difference in the accuracy of the results with the uniform distribution and the Chebyshev-Gauss-Lobatto point distribution. Moreover, it was observed that there was no significant difference between the size of the errors near the boundaries and the interior of the domain.

5.2. **Example 2.** We next consider the first biharmonic boundary value problem (3.1) corresponding to the exact solution $u = \sin(\pi x)\cos(\pi y/2)$. In Table 7 we present the sub-optimal shape parameters and their corresponding errors using the LOOCV algorithm with various search intervals for the case $M = 128, N = 64$. As in Example 1, the results we obtained are very stable irrespective to the initial search interval. Without prior knowledge of the exact solution, we can find a good shape parameter and obtain accuracy which is fairly close to the optimal accuracy, as shown in Figure 6.

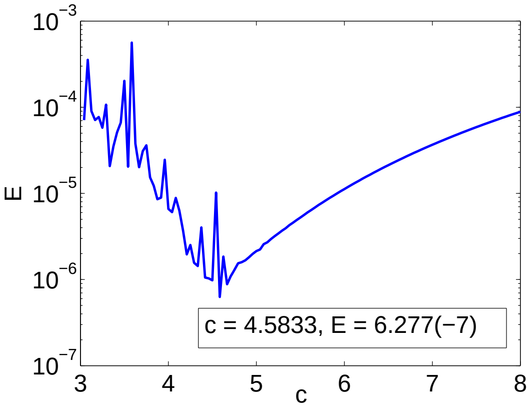

FIGURE 6. Example 2: Maximum relative error versus the shape parameter for $M = 128, N = 64$.

In Table 8 we present the sub-optimal shape parameters and errors in u for various values of $M = N$ using the LOOCV algorithm for the case of Dirichlet and Neumann boundary conditions. The initial search internal is set to [0, 20] and obtain stable and accurate results. We also run a similar test for the corresponding second biharmonic boundary value problem (3.1a), (3.4) and (3.5) in which the initial search interval was set to [0, 10]. The results are shown in Table 9.

5.3. **Example 3.** We finally consider a mixed boundary value problem for the Cauchy-Navier equations of elasticity given by (4.1), (4.5) and (4.1c) and corresponding to the exact solution $u_1 = \mathrm{e}^{x+y}$, $u_2 = \sin(x+y)$. We take Poisson's ratio and the shear modulus to be $\nu = 0.3$ and $\mu = 1$, respectively. In Figure 7 we present the errors in u_1 and u_2 for various numbers of degrees of freedom versus the shape parameter c when the tractions are prescribed on $\partial\Omega_1$ (cf. Section 4.2.1). In this case, the optimal shape parameter is $c = 5.216$ and the corresponding errors in u_1 and u_2 are 3.530(−8) and 1.059(−7), respectively. Note that the errors E_1 and E_2 in u_1 and u_2 are defined using (5.2).

In Table 10 we present results obtained when applying the LOOCV algorithm to find the sub-optimal shape parameter and the corresponding errors in u_1 and u_2. Overall, accurate results are obtained for various initial

[`min`, `max`]	sub-optimal c	E
$[0,\ 8]$	4.433	$1.713(-6)$
$[0, 10]$	4.351	$1.422(-6)$
$[0, 12]$	4.590	$1.066(-6)$
$[0, 14]$	4.211	$4.485(-6)$
$[0, 16]$	4.272	$4.184(-6)$
$[0, 18]$	4.242	$2.411(-6)$
$[0, 20]$	4.235	$1.801(-6)$

Table 7. Example 2: Sub-optimal shape parameters and the corresponding maximum relative errors for various initial search interval using `fminbnd` for $M = 128, N = 64$.

$M = N$	sub-optimal c	E
80	4.003	$1.351(-5)$
100	5.429	$5.353(-6)$
120	7.311	$3.583(-6)$
140	7.857	$5.250(-6)$
160	9.219	$9.647(-6)$

Table 8. Example 2: Sub-optimal shape parameters and the maximum relative errors for various $M = N$.

$M = N$	sub-optimal c	E
80	3.678	$1.651(-5)$
100	4.991	$4, 172(-5)$
120	6.774	$1.839(-4)$
140	7.329	$8.462(-5)$
160	8.451	$9.012(-5)$

Table 9. Example 2: Sub-optimal shape parameters and the maximum relative errors for various $M = N$ for the second biharmonic problem.

search intervals. If we choose the interval properly, the accuracy can be very close to the optimal one as shown in Figure 7. In Table 11 we present results obtained using various $M = N$ for a search interval of [0,20]. We also considered the corresponding Dirichlet boundary value problem (4.1) and in Table 12 we present results obtained using various $M = N$ for a search interval of [0,20] which are very similar to those in Table 11. Moreover, these results are consistent with the corresponding results obtained in Examples 1 and 2.

6. Conclusions

We have applied a Kansa-RBF method for the solution of elliptic boundary value problems in annular domains. With an appropriate choice of collocation points, the discretization of such problems governed by the Poisson, inhomogeneous biharmonic or the Cauchy-Navier equations of elasticity leads to linear systems in which the coefficient matrices are either block circulant or can be easily transformed into block circulant matrices. Thus these systems may be solved efficiently using MDAs leading to substantial savings in computational cost and storage. In addition, a major advantage of the proposed technique, other than its simplicity, is that it is applicable for any RBF and, as shown in our numerical tests, it is applicable to large-scale problems achieving high accuracy. Moreover, the use of the LOOCV algorithm enables us to select a good shape parameter which is critical to ensure appropriate

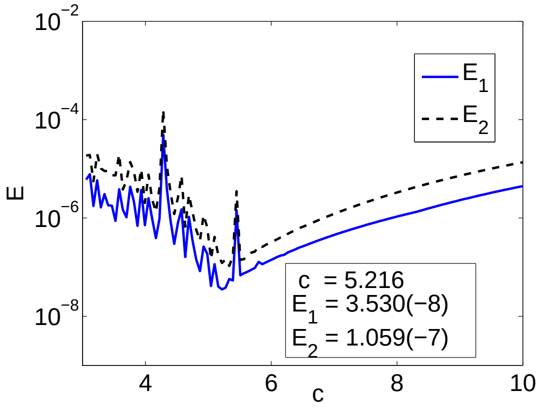

FIGURE 7. Example 3: Maximum relative error versus the shape parameter with $M = 128, N = 64$.

[min, max]	sub-optimal c	E_1	E_2
[2, 8]	6.791	3.647(−7)	9.501(−7)
[3, 8]	5.631	7.482(−8)	1.732(−7)
[0, 8]	4.944	1.654(−7)	3.806(−7)
[0, 10]	8.439	1.487(−6)	4.767(−6)
[0, 12]	8.440	1.488(−6)	4.771(−6)
[0, 14]	8.427	1.472(−6)	4.722(−6)
[0, 16]	8.421	1.464(−6)	4.697(−6)
[0, 18]	8.443	1.491(−6)	4.778(−6)
[0, 20]	8.432	1.477(−6)	4.738(−6)

TABLE 10. Example 3: Sub-optimal shape parameters and the corresponding maximum relative errors for various initial search intervals using fminbnd with $M = 128, N = 64$.

$M = N$	sub-optimal c	E_1	E_2
80	6.391	5.987(−6)	2.484(−5)
100	7.585	4.755(−6)	6.914(−6)
120	9.202	1.728(−6)	5.171(−6)
140	12.071	4.045(−6)	1.431(−5)
160	12.121	1.966(−6)	2.943(−6)

TABLE 11. Example 3: Sub-optimal shape parameters and the maximum relative errors for various $M = N$.

numerical accuracy. We would like to indicate that the proposed method can be extended to solving a large class of partial differential equations without difficulty. The algorithm proposed in this paper can be easily applied to other RBF collocation methods [4, 8, 28]. In the spirit of reproducible research and for the convenience of interested readers the MATLAB© code for Example 1, as described by Algorithm 1 in Section 2.3, maybe accessed at [35].

Possible areas of future research include

- The application of the method using compactly supported RBF, see, e.g., [2].
- The application of the method using a localized RBF collocation, see, e.g., [27].

$M = N$	sub-optimal c	E_1	E_2
80	7.227	1.368(−5)	5.191(−5)
100	8.741	1.212(−5)	1.736(−5)
120	9.905	2.877(−6)	8.399(−6)
140	11.695	3.262(−6)	1.173(−5)
160	13.471	4.089(−6)	6.032(−6)

Table 12. Example 3: Sub-optimal shape parameters and the maximum relative errors for various $M = N$ for Dirichlet boundary value problem.

- The extension of the current method for the solution of three-dimensional axisymmetric elliptic boundary value problems, see, e.g., [21].
- The extension of the current method to problems in which the type of boundary conditions alternates on the circular segments of the boundary, see, e.g., [18].

Acknowledgements

The authors wish to thank the referees for their helpful comments and suggestions which resulted in an improved manuscript.

Appendix

We consider the RBFs ϕ_{mn}, $m = 1, \ldots, M, n = 1, \ldots, N$. These satisfy, see, e.g., [7, Appendix D],

$$\phi_{mn}(x, y) = \Phi(r_{mn}), \quad r_{mn}^2(x, y) = (x - x_{mn})^2 + (y - y_{mn})^2. \tag{A.1}$$

It can be easily seen that

$$\frac{\partial \phi_{mn}}{\partial x}(x, y) = \frac{\Phi'(r_{mn})}{r_{mn}}(x - x_{mn}), \quad \frac{\partial \phi_{mn}}{\partial y}(x, y) = \frac{\Phi'(r_{mn})}{r_{mn}}(y - y_{mn}), \tag{A.2}$$

$$\frac{\partial^2 \phi_{mn}}{\partial x^2}(x, y) = \frac{\Phi'(r_{mn})}{r_{mn}} + \left(\frac{r_{mn}\Phi''(r_{mn}) - \Phi'(r_{mn})}{r_{mn}^3}\right)(x - x_{mn})^2, \tag{A.3}$$

$$\frac{\partial^2 \phi_{mn}}{\partial y^2}(x, y) = \frac{\Phi'(r_{mn})}{r_{mn}} + \left(\frac{r_{mn}\Phi''(r_{mn}) - \Phi'(r_{mn})}{r_{mn}^3}\right)(y - y_{mn})^2, \tag{A.4}$$

and

$$\frac{\partial^2 \phi_{mn}}{\partial x \partial y}(x, y) = \left(\frac{r_{mn}\Phi''(r_{mn}) - \Phi'(r_{mn})}{r_{mn}^3}\right)(x - x_{mn})(y - y_{mn}). \tag{A.5}$$

Also,

$$\Delta \phi_{mn}(x, y) = \Phi''(r_{mn}) + \frac{\Phi'(r_{mn})}{r_{mn}} \tag{A.6}$$

and

$$\Delta^2 \phi_{mn}(x, y) = \Phi''''(r_{mn}) + \frac{2\Phi'''(r_{mn})}{r_{mn}} - \frac{\Phi''(r_{mn})}{r_{mn}^2} + \frac{\Phi'(r_{mn})}{r_{mn}^3}. \tag{A.7}$$

In the sequel, we shall be using the notation (cf. (2.8))

$$\begin{aligned} \delta_{m_1,m_2} x &= x_{m_1,n_1} - x_{m_2,n_2} = \mathsf{r}_{n_1} \cos(\vartheta_{m_1} + \frac{2\pi\alpha_{n_1}}{N}) - \mathsf{r}_{n_2} \cos(\vartheta_{m_2} + \frac{2\pi\alpha_{n_2}}{N}), \\ \delta_{m_1,m_2} y &= y_{m_1,n_1} - y_{m_2,n_2} = \mathsf{r}_{n_1} \sin(\vartheta_{m_1} + \frac{2\pi\alpha_{n_1}}{N}) - \mathsf{r}_{n_2} \sin(\vartheta_{m_2} + \frac{2\pi\alpha_{n_2}}{N}). \end{aligned} \tag{A.8}$$

We shall also need the normal derivatives on the boundary $\partial\Omega_1$ which are given by

$$(\mathrm{n}_x, \mathrm{n}_y)(x_{m_1,1}, y_{m_1,1}) = -\left(\cos(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N}),\ \sin(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N})\right). \tag{A.9}$$

We next state and prove the following lemmata:

Lemma 1. *For any radial function, each submatrix* (A_{n_1,n_2}) *in* (2.12) *possesses a circulant structure.*

Proof. We first consider the submatrices (A_{n_1,n_2}), $n_1 = 2, \ldots, N-1$, $n_2 = 1, \ldots, N$, which are defined by (2.13a). From (A.6), $\Delta\phi_{m_2,n_2}$ is an RBF, i.e. it is a function of r_{m_2,n_2} (cf. (A.1)). Moreover,

$$r^2_{m_2,n_2}(x_{m_1,n_1}, y_{m_1,n_1}) = (\delta_{m_1,m_2}x)^2 + (\delta_{m_1,m_2}y)^2 = \mathsf{r}^2_{n_1} + \mathsf{r}^2_{n_2} - 2\mathsf{r}_{n_1}\mathsf{r}_{n_2}\cos\left(\vartheta_{m_1} - \vartheta_{m_2} + \frac{2\pi(\alpha_{n_1} - \alpha_{n_2})}{N}\right),$$

shows that for fixed n_1, n_2, the quantity $r_{m_2,n_2}(x_{m_1,n_1}, y_{m_1,n_1})$ only depends on $(\vartheta_{m_1} - \vartheta_{m_2})$, i.e. $(m_1 - m_2)$ and therefore the submatrices are circulant.
The proof that the submatrices (A_{n_1,n_2}), $n_1 = 1, N$, $n_2 = 1, \ldots, N$, defined by (2.13b)-(2.13c), are circulant in the case of Dirichlet boundary conditions is identical since their elements involve the RBFs ϕ_{m_2,n_2}, which only depend on r_{m_2,n_2}.
In the case of a Neumann boundary condition on $\partial\Omega_1$, however, we have that the submatrices $(A_{1,n})$, $n = 1, \ldots, N$, are defined by (2.15), or

$$(A_{1,n_2})_{m_1,m_2} = \frac{\partial\phi_{m_2,n_2}}{\partial x}(x_{m_1,1}, y_{m_1,1})\mathrm{n}_x(x_{m_1,1}, y_{m_1,1}) + \frac{\partial\phi_{m_2,n_2}}{\partial y}(x_{m_1,1}, y_{m_1,1})\mathrm{n}_y(x_{m_1,1}, y_{m_1,1}),$$

and from (A.2) and (A.9) we obtain

$$= -\frac{\Phi'(r_{m_2n_2})(x_{m_1,1}, y_{m_1,1})}{r_{m_2n_2}(x_{m_1,1}, y_{m_1,1}))}\left[(x_{m_1,1} - x_{m_2,n_2})\cos(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N}) + (y_{m_1,1} - y_{m_2,n_2})\sin(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N})\right].$$

Ignoring the radial part, we examine

$$\begin{aligned}
&(x_{m_1,1} - x_{m_2,n_2})\cos(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N}) + (y_{m_1,1} - y_{m_2,n_2})\sin(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N}) \\
&= \left(\mathsf{r}_1\cos(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N}) - \mathsf{r}_{n_2}\cos(\vartheta_{m_2} + \frac{2\pi\alpha_{n_2}}{N})\right)\cos(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N}) \\
&+ \left(\mathsf{r}_1\sin(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N}) - \mathsf{r}_{n_2}\sin(\vartheta_{m_2} + \frac{2\pi\alpha_{n_2}}{N})\right)\sin(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N}) \\
&= \mathsf{r}_1 - \mathsf{r}_{n_2}\cos(\vartheta_{m_2} - \vartheta_{m_1} + \frac{2\pi(\alpha_{n_2} - \alpha_1)}{N}),
\end{aligned} \tag{A.10}$$

which, again, only depends on $(m_2 - m_1)$. Hence, this part is also radial and the corresponding submatrices are circulant. □

Following identical arguments we also arrive at the corresponding result for the biharmonic case.
Lemma 2. *For any radial function, each submatrix* (A_{n_1,n_2}) *in* (2.12) *corresponding to collocation scheme* (3.2)*, for both the first and the second biharmonic problem, possesses a circulant structure.*

Finally, we consider the corresponding result for the Cauchy-Navier equations of elasticity.

Lemma 3. *For any radial function, each submatrix* $\left(\tilde{A}_{n_1,n_2}\right)$ *in* (4.9) *possesses a* 2×2 *block circulant structure.*

Proof. In order to prove the lemma we need to show that, see, e.g. [22],

$$R_{\vartheta_{m_1}}(A_{n_1,n_2})_{m_1,m_2}R_{\vartheta_{m_2}} = R_{\vartheta_{m_1+m}}(A_{n_1,n_2})_{m_1+m,m_2+m}R_{\vartheta_{m_2+m}}, \quad m_1, m_2 = 1, \ldots, M,\ n_1, n_2 = 1, \ldots, N, \tag{A.11}$$

provided $m_1+m, m_2+m \leq M$. In case $m_1+m > M$, m_1+m is replaced by m_1+m-M and in case $m_2+m > M$, m_2+m is replaced by m_2+m-M.
Since $R^2_{\vartheta_k} = I_2$ proving (A.11) is equivalent to proving

$$R_{\vartheta_{m_1+m}} R_{\vartheta_{m_1}} \left(A_{n_1,n_2}\right)_{m_1,m_2} R_{\vartheta_{m_2}} R_{\vartheta_{m_1+m}} = \left(A_{n_1,n_2}\right)_{m_1+m,m_2+m}, \quad m_1, m_2 = 1, \ldots, M, \; n_1, n_2 = 1, \ldots, N. \tag{A.12}$$

However,

$$R_{\vartheta_{m_1+m}} R_{\vartheta_{m_1}} = \begin{pmatrix} \cos\vartheta_{m_1+m} & \sin\vartheta_{m_1+m} \\ \sin\vartheta_{m_1+m} & -\cos\vartheta_{m_1+m} \end{pmatrix} \begin{pmatrix} \cos\vartheta_{m_1} & \sin\vartheta_{m_1} \\ \sin\vartheta_{m_1} & -\cos\vartheta_{m_1} \end{pmatrix} = \begin{pmatrix} \cos\vartheta_m & -\sin\vartheta_m \\ \sin\vartheta_m & \cos\vartheta_m \end{pmatrix} = W_{\vartheta_m},$$

and

$$R_{\vartheta_{m_1}} R_{\vartheta_{m_1+m}} = \begin{pmatrix} \cos\vartheta_{m_1} & \sin\vartheta_{m_1} \\ \sin\vartheta_{m_1} & -\cos\vartheta_{m_1} \end{pmatrix} \begin{pmatrix} \cos\vartheta_{m_1+m} & \sin\vartheta_{m_1+m} \\ \sin\vartheta_{m_1+m} & -\cos\vartheta_{m_1+m} \end{pmatrix} = \begin{pmatrix} \cos\vartheta_m & \sin\vartheta_m \\ -\sin\vartheta_m & \cos\vartheta_m \end{pmatrix} = W^{-1}_{\vartheta_m}.$$

Hence proving (A.12) is equivalent to proving that

$$W_{\vartheta_m} \left(A_{n_1,n_2}\right)_{m_1,m_2} W^{-1}_{\vartheta_m} = \left(A_{n_1,n_2}\right)_{m_1+m,m_2+m}. \tag{A.13}$$

From (4.4a), for $m_1, m_2 = 1, \ldots, M, \; n_1 = 2, \ldots, N-1, \; n_2 = 1, \ldots, N$, we can write

$$\left(A_{n_1,n_2}\right)_{m_1,m_2} =$$

$$\mu \Delta \phi_{m_2,n_2}(x_{m_1,n_1}, y_{m_1,n_1}) I_2 + \frac{\mu}{1-2\nu} \begin{pmatrix} \dfrac{\partial^2 \phi_{m_2,n_2}}{\partial x^2}(x_{m_1,n_1}, y_{m_1,n_1}) & \dfrac{\partial^2 \phi_{m_2,n_2}}{\partial x \partial y}(x_{m_1,n_1}, y_{m_1,n_1}) \\ \dfrac{\partial^2 \phi_{m_2,n_2}}{\partial x \partial y}(x_{m_1,n_1}, y_{m_1,n_1}) & \dfrac{\partial^2 \phi_{m_2,n_2}}{\partial y^2}(x_{m_1,n_1}, y_{m_1,n_1}) \end{pmatrix}. \tag{A.14}$$

We first consider the first term in (A.14), namely $\mu \Delta \phi_{m_2,n_2}(x_{m_1,n_1}, y_{m_1,n_1}) I_2$.
We clearly have that

$$\mu \Delta \phi_{m_2,n_2}(x_{m_1,n_1}, y_{m_1,n_1}) W_{\vartheta_m} I_2 W^{-1}_{\vartheta_m} = \mu \Delta \phi_{m_2,n_2}(x_{m_1,n_1}, y_{m_1,n_1}) I_2 = \mu \Delta \phi_{m_2+m,n_2}(x_{m_1+m,n_1}, y_{m_1+m,n_1}) I_2,$$

since the Laplacian of $\phi_{m_2,n_2}(x,y)$ is radial from (A.6) and

$$\begin{aligned} (\delta_{m_1,m_2} x)^2 + (\delta_{m_1,m_2} y)^2 &= \mathsf{r}^2_{n_1} + \mathsf{r}^2_{n_2} - 2\mathsf{r}_{n_1} \mathsf{r}_{n_2} \cos\left(\vartheta_{m_1} - \vartheta_{m_2} + \frac{2\pi(\alpha_{n_1} - \alpha_{n_2})}{N}\right) \\ &= (\delta_{m_1+m,m_2+m} x)^2 + (\delta_{m_1+m,m_2+m} y)^2. \end{aligned} \tag{A.15}$$

Therefore, the first term in (A.14) satisfies (A.13).
For the second term in (A.14), from (A.3)-(A.5) and ignoring the radial parts we only need show that

$$\begin{aligned} &\begin{pmatrix} \cos\vartheta_m & -\sin\vartheta_m \\ \sin\vartheta_m & \cos\vartheta_m \end{pmatrix} \begin{pmatrix} (\delta_{m_1,m_2} x)^2 & (\delta_{m_1,m_2} x)(\delta_{m_1,m_2} y) \\ (\delta_{m_1,m_2} x)(\delta_{m_1,m_2} y) & (\delta_{m_1,m_2} y)^2 \end{pmatrix} \begin{pmatrix} \cos\vartheta_m & \sin\vartheta_m \\ -\sin\vartheta_m & \cos\vartheta_m \end{pmatrix} \\ &= \begin{pmatrix} (\delta_{m_1+m,m_2+m} x)^2 & (\delta_{m_1+m,m_2+m} x)(\delta_{m_1+m,m_2+m} y) \\ (\delta_{m_1+m,m_2+m} x)(\delta_{m_1+m,m_2+m} y) & (\delta_{m_1+m,m_2+m} y)^2 \end{pmatrix}. \end{aligned} \tag{A.16}$$

By performing the multiplications on the left hand side of (A.16) we obtain that it is equal to

$$\begin{pmatrix} (\cos\vartheta_m \delta_{m_1,m_2} x - \sin\vartheta_m \delta_{m_1,m_2} y)^2 & (\cos\vartheta_m \delta_{m_1,m_2} x - \sin\vartheta_m \delta_{m_1,m_2} y)(\sin\vartheta_m \delta_{m_1,m_2} x + \cos\vartheta_m \delta_{m_1,m_2} y) \\ (\cos\vartheta_m \delta_{m_1,m_2} x - \sin\vartheta_m \delta_{m_1,m_2} y)(\sin\vartheta_m \delta_{m_1,m_2} x + \cos\vartheta_m \delta_{m_1,m_2} y) & (\sin\vartheta_m \delta_{m_1,m_2} x + \cos\vartheta_m \delta_{m_1,m_2} y)^2 \end{pmatrix}.$$

Moreover, it can be easily shown that

$$\cos\vartheta_m \delta_{m_1,m_2} x - \sin\vartheta_m \delta_{m_1,m_2} y = \mathsf{r}_{n_1} \cos(\vartheta_{m_1+m} + \frac{2\pi\alpha_{n_1}}{N}) - \mathsf{r}_{n_2} \cos(\vartheta_{m_2+m} + \frac{2\pi\alpha_{n_2}}{N})$$

and

$$\sin\vartheta_m \delta_{m_1,m_2} x + \cos\vartheta_m \delta_{m_1,m_2} y = \mathsf{r}_{n_1} \sin(\vartheta_{m_1+m} + \frac{2\pi\alpha_{n_1}}{N}) - \mathsf{r}_{n_2} \sin(\vartheta_{m_2+m} + \frac{2\pi\alpha_{n_2}}{N}),$$

from which (A.16) follows. Hence, the second term in (A.14) also satisfies (A.13).

We next need to prove that

$$W_{\vartheta_m}\left(A_{1,n_2}\right)_{m_1,m_2} W_{\vartheta_m}^{-1} = \left(A_{1,n_2}\right)_{m_1+m,m_2+m} \quad \text{and} \quad W_{\vartheta_m}\left(A_{N,n_2}\right)_{m_1,m_2} W_{\vartheta_m}^{-1} = \left(A_{N,n_2}\right)_{m_1+m,m_2+m}. \tag{A.17}$$

Since from (4.4b) we can write

$$\left(A_{1,n_2}\right)_{m_1,m_2} = \phi_{m_2,n_2}(x_{m_1,1}, y_{m_1,1}) I_2 \quad \text{and} \quad \left(A_{N,n_2}\right)_{m_1,m_2} = \phi_{m_2,n_2}(x_{m_1,N}, y_{m_1,N}) I_2,$$

from (A.15), following a similar argument to the one used for the first term of (A.14), (A.17) follows.

Finally, we show that if instead of the displacements u_1, u_2 we have that the tractions t_1, t_2 are prescribed on, say, the boundary $\partial\Omega_1$, cf. Section 4.2.1, then

$$W_{\vartheta_m}\left(A_{1,n_2}\right)_{m_1,m_2} W_{\vartheta_m}^{-1} = \left(A_{1,n_2}\right)_{m_1+m,m_2+m} \tag{A.18}$$

is still true.
We write

$$\left(A_{1,n}\right)_{m_1,m_2} = \begin{pmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{pmatrix}, \tag{A.19}$$

where $T_{ij}, i, j = 1, 2$, are the appropriate quantities in (4.6).
We can easily show that

$$W_{\vartheta_m} \begin{pmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{pmatrix} W_{\vartheta_m}^{-1}$$

$$= \begin{pmatrix} \cos^2\vartheta_m T_{11} - \cos\vartheta_m \sin\vartheta_m (T_{21}+T_{12}) + \sin^2\vartheta_m T_{22} & \cos^2\vartheta_m T_{12} + \cos\vartheta_m \sin\vartheta_m (T_{11}-T_{22}) - \sin^2\vartheta_m T_{21} \\ \cos^2\vartheta_m T_{21} + \cos\vartheta_m \sin\vartheta_m (T_{11}-T_{22}) - \sin^2\vartheta_m T_{12} & \sin^2\vartheta_m T_{11} + \cos\vartheta_m \sin\vartheta_m (T_{21}+T_{12}) + \cos^2\vartheta_m T_{22} \end{pmatrix}.$$

We shall first show that the first term in (4.6) is radial. Using the notation of (A.8) with $n_1 = 1$, we have from (A.2) that

$$\begin{aligned} &\frac{\partial\phi_{m_2,n}}{\partial x}(x_{m_1,1}, y_{m_1,1})\mathrm{n}_x(x_{m_1,1}, y_{m_1,1}) + \frac{\partial\phi_{m_2,n}}{\partial y}(x_{m_1,1}, y_{m_1,1})\mathrm{n}_y(x_{m_1,1}, y_{m_1,1}) \\ &\quad = \frac{\Phi'(r_{m_2 n})}{r_{m_2 n}}\left[\delta_{m_1,m_2}x\, \mathrm{n}_x(x_{m_1,1}, y_{m_1,1}) + \delta_{m_1,m_2}y\, \mathrm{n}_y(x_{m_1,1}, y_{m_1,1})\right], \end{aligned}$$

and ignoring the radial factor, we have, using (A.10) that

$$\begin{aligned} &\delta_{m_1,m_2}x\, \mathrm{n}_x(x_{m_1,1}, y_{m_1,1}) + \delta_{m_1,m_2}y\, \mathrm{n}_y(x_{m_1,1}, y_{m_1,1}) \\ &\quad = \mathsf{r}_1 - \mathsf{r}_{n_2}\cos(\vartheta_{m_2} - \vartheta_{m_1} + \frac{2\pi(\alpha_{n_2}-\alpha_1)}{N}), \end{aligned}$$

which only depends on $(m_2 - m_1)$, hence it is radial.
We next consider the second term in (4.6). Dropping the multiplying constants and the obviously radial parts, and with the appropriate notation, we consider the quantities

$$\tilde{T}_{11}(m_1, m_2) = \delta_{m_1,m_2}x\, \mathrm{n}_x(x_{m_1,1}, y_{m_1,1}) = \mathsf{r}_1\cos^2(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N}) - \mathsf{r}_{n_2}\cos(\vartheta_{m_2} + \frac{2\pi\alpha_{n_2}}{N})\cos(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N}),$$

$$\begin{aligned} \tilde{T}_{12}(m_1, m_2) &= 2\nu\delta_{m_1,m_2}y\, \mathrm{n}_x(x_{m_1,1}, y_{m_1,1}) + (1-2\nu)\delta_{m_1,m_2}x\, \mathrm{n}_y(x_{m_1,1}, y_{m_1,1}) \\ &= \mathsf{r}_1\sin(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N})\cos(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N}) \\ &\quad -\mathsf{r}_{n_2}\left(2\nu\cos(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N})\sin(\vartheta_{m_2} + \frac{2\pi\alpha_{n_2}}{N}) + (1-2\nu)\sin(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N})\cos(\vartheta_{m_2} + \frac{2\pi\alpha_{n_2}}{N})\right) \\ &= \mathsf{r}_1\sin(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N})\cos(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N}) \\ &\quad -\mathsf{r}_{n_2}\left(2\nu\sin(\vartheta_{m_2} - \vartheta_{m_1} + \frac{2\pi(\alpha_{n_2}-\alpha_1)}{N}) + \sin(\vartheta_{m_1} + \frac{2\pi\alpha_1}{N})\cos(\vartheta_{m_2} + \frac{2\pi\alpha_{n_2}}{N})\right), \end{aligned}$$

$$\begin{aligned}\tilde{T}_{21}(m_1,m_2) &= (1-2\nu)\delta_{m_1,m_2} y\,\mathrm{n}_x(x_{m_1,1},y_{m_1,1}) + 2\nu\delta_{m_1,m_2} x\,\mathrm{n}_y(x_{m_1,1},y_{m_1,1})\\ &= \mathsf{r}_1 \sin(\vartheta_{m_1}+\frac{2\pi\alpha_1}{N})\cos(\vartheta_{m_1}+\frac{2\pi\alpha_1}{N})\\ -\mathsf{r}_{n_2}&\left((1-2\nu)\cos(\vartheta_{m_1}+\frac{2\pi\alpha_1}{N})\sin(\vartheta_{m_2}+\frac{2\pi\alpha_{n_2}}{N}) + 2\nu\sin(\vartheta_{m_1}+\frac{2\pi\alpha_1}{N})\cos(\vartheta_{m_2}+\frac{2\pi\alpha_{n_2}}{N})\right)\\ &= \mathsf{r}_1 \sin(\vartheta_{m_1}+\frac{2\pi\alpha_1}{N})\cos(\vartheta_{m_1}+\frac{2\pi\alpha_1}{N})\\ -\mathsf{r}_{n_2}&\left(2\nu\sin(\vartheta_{m_1}-\vartheta_{m_2}+\frac{2\pi(\alpha_1-\alpha_{n_2})}{N}) + \cos(\vartheta_{m_1}+\frac{2\pi\alpha_1}{N})\sin(\vartheta_{m_2}+\frac{2\pi\alpha_{n_2}}{N})\right),\\ \tilde{T}_{22}(m_1,m_2) &= \delta_{m_1,m_2} y\,\mathrm{n}_y(x_{m_1,1},y_{m_1,1}) = \mathsf{r}_1\sin^2(\vartheta_{m_1}+\frac{2\pi\alpha_1}{N}) - \mathsf{r}_{n_2}\sin(\vartheta_{m_2}+\frac{2\pi\alpha_{n_2}}{N})\sin(\vartheta_{m_1}+\frac{2\pi\alpha_1}{N}).\end{aligned}$$

By using (A.9), we first consider the term

$$\begin{aligned}&\cos^2\vartheta_m\tilde{T}_{11} - \cos\vartheta_m\sin\vartheta_m(\tilde{T}_{21}+\tilde{T}_{12}) + \sin^2\vartheta_m\tilde{T}_{22}\\ &= \cos^2\vartheta_m\delta_{m_1,m_2}x\,\mathrm{n}_x - \cos\vartheta_m\sin\vartheta_m\,(\delta_{m_1,m_2}y\,\mathrm{n}_x + \delta_{m_1,m_2}x\,\mathrm{n}_y) + \sin^2\vartheta_m\delta_{m_1,m_2}y\,\mathrm{n}_y\\ &= \mathsf{r}_1\cos^2(\vartheta_{m_1}+\vartheta_m+\frac{2\pi\alpha_1}{N}) - \mathsf{r}_{n_2}\cos(\vartheta_{m_2}+\vartheta_m+\frac{2\pi\alpha_{n_2}}{N})\cos(\vartheta_{m_1}+\vartheta_m+\frac{2\pi\alpha_1}{N})\\ &= \tilde{T}_{11}(m_1+m,m_2+m).\end{aligned}$$

We next consider

$$\begin{aligned}&\cos^2\vartheta_m\tilde{T}_{12} + \cos\vartheta_m\sin\vartheta_m(\tilde{T}_{11}-\tilde{T}_{22}) - \sin^2\vartheta_m\tilde{T}_{21}\\ &= \cos^2\vartheta_m\,(2\nu\delta_{m_1,m_2}y\,\mathrm{n}_x + (1-2\nu)\delta_{m_1,m_2}x\,\mathrm{n}_y) + \cos\vartheta_m\sin\vartheta_m(\delta_{m_1,m_2}x\,\mathrm{n}_x - \delta_{m_1,m_2}y\,\mathrm{n}_y)\\ &\quad - \sin^2\vartheta_m\,((1-2\nu)\delta_{m_1,m_2}y\,\mathrm{n}_x + 2\nu\delta_{m_1,m_2}x\,\mathrm{n}_y)\\ &= -2\nu\mathsf{r}_{n_2}\sin(\vartheta_{m_2}-\vartheta_{m_1}+\frac{2\pi(\alpha_{n_2}-\alpha_1)}{N}) + \mathsf{r}_1\sin(\vartheta_{m_1}+\vartheta_m+\frac{2\pi\alpha_1}{N})\cos(\vartheta_{m_1}+\vartheta_m+\frac{2\pi\alpha_1}{N})\\ &\quad -\mathsf{r}_{n_2}\sin(\vartheta_{m_1}+\vartheta_m+\frac{2\pi\alpha_1}{N})\cos(\vartheta_{m_2}+\vartheta_m+\frac{2\pi\alpha_{n_2}}{N}) = \tilde{T}_{12}(m_1+m,m_2+m).\end{aligned}$$

Similarly, we have that

$$\begin{aligned}&\cos^2\vartheta_m\tilde{T}_{21} + \cos\vartheta_m\sin\vartheta_m(\tilde{T}_{11}-\tilde{T}_{22}) - \sin^2\vartheta_m\tilde{T}_{12}\\ &= \cos^2\vartheta_m\,((1-2\nu)\delta_{m_1,m_2}y\,\mathrm{n}_x + 2\nu\delta_{m_1,m_2}x\,\mathrm{n}_y) + \cos\vartheta_m\sin\vartheta_m(\delta_{m_1,m_2}x\,\mathrm{n}_x - \delta_{m_1,m_2}y\,\mathrm{n}_y)\\ &\quad - \sin^2\vartheta_m\,(2\nu\delta_{m_1,m_2}y\,\mathrm{n}_x + (1-2\nu)\delta_{m_1,m_2}x\,\mathrm{n}_y)\\ &= -2\nu\mathsf{r}_{n_2}\sin(\vartheta_{m_1}-\vartheta_{m_2}+\frac{2\pi(\alpha_1-\alpha_{n_2})}{N}) + \mathsf{r}_1\sin(\vartheta_{m_1}+\vartheta_m+\frac{2\pi\alpha_1}{N})\cos(\vartheta_{m_1}+\vartheta_m+\frac{2\pi\alpha_1}{N})\\ &\quad -\mathsf{r}_{n_2}\cos(\vartheta_{m_1}+\vartheta_m+\frac{2\pi\alpha_1}{N})\sin(\vartheta_{m_2}+\vartheta_m+\frac{2\pi\alpha_{n_2}}{N}) = \tilde{T}_{21}(m_1+m,m_2+m).\end{aligned}$$

Finally,

$$\begin{aligned}&\sin^2\vartheta_m\tilde{T}_{11} + \cos\vartheta_m\sin\vartheta_m(\tilde{T}_{21}+\tilde{T}_{12}) + \cos^2\vartheta_m\tilde{T}_{22}\\ &= \sin^2\vartheta_m\delta_{m_1,m_2}x\,\mathrm{n}_x + \cos\vartheta_m\sin\vartheta_m\,(\delta_{m_1,m_2}y\,\mathrm{n}_x + \delta_{m_1,m_2}x\,\mathrm{n}_y) + \cos^2\vartheta_m\delta_{m_1,m_2}y\,\mathrm{n}_y.\\ &= \mathsf{r}_1\sin^2(\vartheta_{m_1}+\vartheta_m+\frac{2\pi\alpha_1}{N}) - \mathsf{r}_{n_2}\sin(\vartheta_{m_2}+\vartheta_m+\frac{2\pi\alpha_{n_2}}{N})\sin(\vartheta_{m_1}+\vartheta_m+\frac{2\pi\alpha_1}{N})\\ &= \tilde{T}_{22}(m_1+m,m_2+m).\end{aligned}$$

Therefore (A.18) is satisfied and the proof is complete. □

References

[1] B. Bialecki, G. Fairweather, and A. Karageorghis, *Matrix decomposition algorithms for elliptic boundary value problems: a survey*, Numer. Algorithms **56** (2011), 253–295.

[2] M. D. Buhmann, *A new class of radial basis functions with compact support*, Math. Comp. **70** (2001), 307–318.

[3] ———, *Radial Basis Functions: Theory and Implementations*, Cambridge Monographs on Applied and Computational Mathematics, vol. 12, Cambridge University Press, Cambridge, 2003.

[4] C. S. Chen, C. M. Fan and P. H. Wen, *The method of particular solutions for solving certain partial differential equations*, Numer. Methods Partial Differential Equations **28** (2012), 506–522.

[5] A. H.-D. Cheng, *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. **36** (2012), 220–239.

[6] P. J. Davis, *Circulant Matrices, Second edition*, AMS Chelsea Publishing, Providence, Rhode Island, 1994.

[7] G. E. Fasshauer, *Meshfree Approximation Methods with MATLAB*, Interdisciplinary Mathematical Sciences, vol. 6, World Scientific Publishing Co. Pte. Ltd., Hackensack, NJ, 2007.

[8] ——— *Solving partial differential equations by collocation with radial basis functions*, Surface Fitting and Multiresolution Methods, (A. Le Mehaute, C. Rabut, and L. L. Schumaker, eds), Vanderbilt University Press, Nashville, TN, 1997, pp. 131–138.

[9] G. E. Fasshauer and J. G. Zhang, *On choosing optimal shape parameters for RBF approximation*, Numer. Algorithms **45** (2007), 345–368.

[10] B. Fornberg, E. Larsson and N. Flyer, *Stable computations with Gaussian radial basis functions*, SIAM J. Sci. Comput. **33** (2011), no. 2, 869–892.

[11] B. Fornberg and G. Wright, *Stable computation of multiquadric interpolants for all values of the shape parameter*, Comput. Math. Appl. **48** (2004), no. 5-6, 853–867.

[12] F. Hartmann, *Elastostatics*, Progress in Boundary Element Methods. Vol. 1 (C. A. Brebbia, ed.), Pentech Press, London, 1981, pp. 84–167.

[13] A. R. H. Heryudono and T. A. Driscoll, *Radial basis function interpolation on irregular domain through conformal transplantation*, J. Sci. Comput. **44** (2010), 286–300.

[14] E. J. Kansa, *Multiquadrics—a scattered data approximation scheme with applications to computational fluid-dynamics. II. Solutions to parabolic, hyperbolic and elliptic partial differential equations*, Comput. Math. Appl. **19** (1990), no. 8-9, 147–161.

[15] E.J. Kansa and Y.C. Hon, *Circumventing the ill-conditioning problem with multiquadric radial basis functions: applications to elliptic partial differential equations*, Comput. Math. Appl. **39** (2000), no. 7-8, 123–137.

[16] A. Karageorghis, *Efficient Kansa-type MFS algorithm for elliptic problems*, Numer. Algorithms **54** (2010), 261–278.

[17] ———, *Efficient MFS algorithms for inhomogeneous polyharmonic problems*, J. Sci. Comput. **46** (2011), 519–541.

[18] ———, *The method of fundamental solutions for elliptic problems in circular domains with mixed boundary conditions*, Numer. Algorithms **68** (2015), 185–211.

[19] A. Karageorghis and C. S. Chen, *A Kansa-RBF method for Poisson problems in annular domains*, Boundary elements and other mesh reduction methods XXXVII (C. A. Brebbia and A. H-D. Cheng, eds.), WIT Press, Southampton, 2014, pp. 77–83.

[20] A. Karageorghis, C. S. Chen, and Y.-S. Smyrlis, *A matrix decomposition RBF algorithm: approximation of functions and their derivatives*, Appl. Numer. Math. **57** (2007), 304–319.

[21] ———, *Matrix decomposition RBF algorithm for solving 3D elliptic problems*, Eng. Anal. Bound. Elem. **33** (2009), no. 12, 1368–1373.

[22] A. Karageorghis and L. Marin, *Efficient MFS algorithms for problems in thermoelasticity*, J. Sci. Comput. **56** (2013), 96–121.

[23] A. Karageorghis and Y.-S. Smyrlis, *Matrix decomposition MFS algorithms for elasticity and thermo-elasticity problems in axisymmetric domains*, J. Comput. Appl. Math. **206** (2007), 774–795.

[24] A. Karageorghis, Y.-S. Smyrlis, and T. Tsangaris, *A matrix decomposition MFS algorithm for certain linear elasticity problems*, Numer. Algorithms **43** (2006), 123–149.

[25] E. Larsson and B. Fornberg, *A numerical study of some radial basis function based solution methods for elliptic PDEs*, Comput. Math. Appl. **46** (2003), 891–902.

[26] C. K. Lee, X. Liu, and S. C. Fan, *Local multiquadric approximation for solving boundary value problems*, Comput. Mech. **30** (2003), 396–409.

[27] M. Li, W. Chen, and C. S. Chen, *The localized RBFs collocation methods for solving high dimensional PDEs*, Eng. Anal. Bound. Elem. **37** (2013), 1300–1304.

[28] N. Mai-Duy and T. Tran-Cong, *Indirect RBFN method with thin plate splines for numerical solution of differential equations*, CMES Comput. Model. Eng. Sci. **4** (2003), 85–102.

[29] B. Matérn, *Spatial Variation*, second ed., Lecture Notes in Statistics, vol. 36, Springer-Verlag, Berlin, 1986.

[30] C. T. Mouat, *Fast Algorithms and Preconditioning Techniques for Fitting Radial Basis Functions*, Ph.D. thesis, Mathematics Department, University of Canterbury, New Zealand, 2001.

[31] The MathWorks, Inc., 3 Apple Hill Dr., Natick, MA, *Matlab*.

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

[33] R. Schaback, *Error estimates and condition numbers for radial basis function interpolation*, **3** (1995), 251–264.

[34] ______, *Multivariate interpolation and approximation by translates of a basis function*, Approximation Theory VIII, Volume I (C. K. Chiu and L. L. Schumaker, eds.), World Scientific Publishing Co., Inc, Singapore, 1995, pp. 491–514.

[35] `http://www.math.usm.edu/cschen/JOMP/Example1.m`

School of Mathematics, Taiyuan University of Technology, China
E-mail address: `lucyyanxiao@163.com`

Department of Mathematics and Statistics, University of Cyprus/ Πανεπιστημιο Κυπρου, P.O.Box 20537, 1678 Nicosia/Λευκωσια, Cyprus/Κυπρος
E-mail address: `andreask@ucy.ac.cy`

School of Mathematics, Taiyuan University of Technology, China & Department of Mathematics, University of Southern Mississippi, Hattiesburg, MS 39406, USA
E-mail address: `cschen.math@gmail.com`