

The University of Southern Mississippi
The Aquila Digital Community

Faculty Publications

2-2018

Improved Kansa RBF Method for the Solution of Nonlinear Boundary Value Problems

Malgorzata A. Jankowska

Poznan University of Technology, malgorzata.jankowska@put.poznan.pl

Andreas Karageorghis

University of Cyprus, andreask@ucy.ac.cy

Ching-Shyang Chen

University of Southern Mississippi, CS.Chen@usm.edu

Follow this and additional works at: https://aquila.usm.edu/fac_pubs

 Part of the [Mathematics Commons](#)

Recommended Citation

Jankowska, M. A., Karageorghis, A., Chen, C. (2018). Improved Kansa RBF Method for the Solution of Nonlinear Boundary Value Problems. *Engineering Analysis with Boundary Elements*, 87, 173-183.
Available at: https://aquila.usm.edu/fac_pubs/14918

This Article is brought to you for free and open access by The Aquila Digital Community. It has been accepted for inclusion in Faculty Publications by an authorized administrator of The Aquila Digital Community. For more information, please contact Joshua.Cromwell@usm.edu.

IMPROVED KANSA RBF METHOD FOR THE SOLUTION OF NONLINEAR BOUNDARY VALUE PROBLEMS

MALGORZATA A. JANKOWSKA, ANDREAS KARAGEORGHIS, AND C. S. CHEN

ABSTRACT. We apply the Kansa–radial basis function (RBF) collocation method to two–dimensional nonlinear boundary value problems. In it, the solution is approximated by a linear combination of RBFs and the governing equation and boundary conditions are satisfied in a collocation sense at interior and boundary points, respectively. The nonlinear system of equations resulting from the Kansa–RBF discretization for the unknown coefficients in the RBF approximation is solved by directly applying a standard nonlinear solver. In a natural way, the value of the shape parameter in the RBFs employed in the approximation may be included in the unknowns to be determined. The numerical results of several examples are presented and analyzed.

1. INTRODUCTION

Radial basis function (RBF) methods have become popular in recent years in approximation theory as well as in the numerical solution of partial differential equations. The most widely used RBF method for the latter class of problems is the RBF collocation method due to Kansa [13], known as the *Kansa method*. The popularity of this method is due to its meshlessness which means that only a set of points is required in the discretization of the continuous problem. This renders the implementation of the method particularly easy, especially for problems in complex geometries and/or in three dimensions. A disadvantage of the method is the (unknown) optimal choice of the shape parameter which is found in most RBFs. Various techniques have been proposed for the determination of an appropriate value of the shape parameter, see e.g. [5, 14, 15, 18, 20, 24]. In addition, the RBF collocation methods discretization leads to highly ill-conditioned matrices and this has limited the accuracy to a certain level. Traditionally, RBF expansions have been augmented with linear combinations of low degree polynomial basis functions primarily for theoretical reasons. This approach, however, leads to little or no improvement in accuracy and has therefore been largely ignored in most applications. Recently, however, Yao *et. al* [30] discovered that the accuracy can be significantly improved if the RBF approximations are enriched with higher degree polynomial basis functions. Although, in general, high degree polynomials are numerically notoriously unstable, when coupled with RBFs, this instability is somehow tamed. From numerical observations, if the RBF expansion is augmented with low degree polynomial basis functions, the major contribution to the accuracy of the approximation is due to the RBFs. In contrast, when the expansion is augmented with higher degree polynomial basis functions, it

Date: November 8, 2017.

2010 Mathematics Subject Classification. Primary 65N35; Secondary 65H10.

Key words and phrases. radial basis functions, Kansa method, collocation.

is the polynomials that gradually take over as the major contributors to the accuracy and RBFs play the (minor) role of merely stabilizing the system. In this paper, we adopt this new modified RBF collocation approach of enriching the RBF expansions with polynomial basis functions to improve the accuracy. In [12] we presented some preliminary results for the solution of second order nonlinear boundary value problems using the Kansa method. In the method proposed in [12] the solution is expressed as a linear combination of only radial basis functions (see (2.2) below) without the inclusion of the polynomial basis as proposed in the current study. The latter *improved Kansa approximation* (see (2.5) below) yields considerably superior accuracy than the approach used in [12]. Furthermore, in this paper we also consider the case of using a predetermined shape parameter instead of determining it from the nonlinear solver.

In this work we shall consider the solution of boundary value problems for nonlinear partial differential equations. The RBF discretization of these problems invariably leads to systems of nonlinear equations which we shall solve using standard software. In particular, we shall be using the MATLAB[®] optimization toolbox routines `fsolve` and `lsqnonlin`. Since we shall be solving nonlinear problems it seems natural to include the (unknown) value of the shape parameter in the set of unknowns of the problem. Thus the solution of the nonlinear problem yields not only the coefficients in the RBF approximation to the solution but, also, an appropriate value of the shape parameter. As mentioned earlier, when the RBF expansions are augmented with high degree polynomial basis functions, the RBFs play a merely stabilizing (and minor) role with regard to accuracy and the shape parameter can be chosen more freely. This provides another alternative for choosing the shape parameter. Moreover, one of the main attractions of the proposed method is its simplicity and the ease with which it can be implemented.

The paper is organized as follows. In Section 2 we present the type of second order boundary value problems considered and describe the formulation of the Kansa method for their solution. The results of three numerical examples are presented in Section 3. In Section 4 we describe the formulation of the proposed method to fourth order boundary value problems and in Section 5 present the results of two further numerical examples. Finally, some concluding remarks are provided in Section 6.

2. THE KANSA METHOD

2.1. The problem. We consider the boundary value problem in \mathbb{R}^2

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

subject to the boundary condition

$$\mathcal{B}u = g \quad \text{on } \partial\Omega, \tag{2.1b}$$

where \mathcal{L} is a second order nonlinear elliptic operator and \mathcal{B} is a linear (or nonlinear) operator describing the boundary condition.

2.2. The method. In Kansa's method [13] we approximate the solution u of boundary value problem (2.1) by a linear combination of RBFs.

$$u_N(x, y) = \sum_{n=1}^N a_n \phi_n(x, y), \quad (x, y) \in \overline{\Omega}. \quad (2.2)$$

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

$$\phi_n(x, y) = \Phi(r_n), \quad \text{where } r_n^2 = (x - x_n)^2 + (y - y_n)^2. \quad (2.3)$$

Thus each RBF ϕ_n is associated with a point (x_n, y_n) . These points $\{(x_n, y_n)\}_{n=1}^N$ are usually referred to as *centers*. We shall assume that we have N_{int} interior centres $\{(x_n, y_n)\}_{n=1}^{N_{\text{int}}}$ and N_{bry} boundary centres $\{(x_n, y_n)\}_{n=N_{\text{int}}+1}^{N_{\text{int}}+N_{\text{bry}}}$. We take $N = N_{\text{int}} + N_{\text{bry}}$.

In [29], for scattered data interpolation problems, expansions of certain types of conditionally positive definite RBFs were augmented by low degree polynomial basis functions to ensure the invertibility of the resultant matrices. In the case of the Kansa method, Hon and Schaback [11] indicated that the corresponding resultant matrices may be singular in only very rare cases. More recently, Fasshauer [8] proposed a modification of the Kansa method, based on Hermite collocation, to ensure the invertibility of the resulting coefficient matrix. However, because of the complexity of the implementation of RBF Hermite collocation, this approach has apparently not been well received and, despite the invertibility issue, the Kansa method remains very popular. Furthermore, in terms of accuracy there is no evidence of an obvious benefit in adding these terms and, therefore, in most applications they have been ignored. Recently, in [30] it was shown that the addition of higher degree polynomial basis functions in the method of particular solutions (MPS) lead to a significant improvement in accuracy. As will be described in the sequel, the same approach is also equally effective for the Kansa method.

Let \mathbb{P}_p be the set of bivariate polynomials of degree up to p and $\{p_k\}_{k=1}^K$ be a basis of \mathbb{P}_p [11, 30]. It is known that the number of polynomial terms for degree p is $K = (p+1)(p+2)/2$. The polynomial basis is thus

$$p_k(x, y) = x^{i-j} y^j, \quad 0 \leq j \leq i, \quad 0 \leq i \leq p, \quad \text{for } k = 1, \dots, K. \quad (2.4)$$

In the modified Kansa method with an augmented polynomial basis, the approximation (2.2) of the solution of boundary value problem (2.1) is thus replaced by

$$u_N(x, y) = \sum_{n=1}^N a_n \phi_n(x, y) + \sum_{k=1}^K a_{N+k} p_k(x, y), \quad (x, y) \in \overline{\Omega}. \quad (2.5)$$

An example of an RBF is the *normalized multiquadric* basis function (MQ)

$$\phi_n(x, y) = \Phi(r_n) = \sqrt{(cr_n)^2 + 1}, \quad (2.6)$$

where c is the *shape parameter*. Such shape parameters are often present in RBFs and the determination of their optimal value remains a major challenge.

Alternatively, one may use polyharmonic splines (PS) given by

$$\phi_n(x, y) = \Phi(r_n) = \begin{cases} r_n^{2\ell-1}, & \text{in 3D,} \\ r_n^{2\ell} \log r_n, & \text{in 2D,} \end{cases} \quad \ell = 1, 2, 3, \dots \quad (2.7)$$

The RBFs r^{2n-1} are also known as the radial power RBFs [9] and may be used for problems in all (both even and odd) dimensions. Clearly, the advantage of PS and radial power RBFs is the absence of a shape parameter.

In addition to the centres we consider the *collocation points* $\{(x_m, y_m)\}_{m=1}^M \in \bar{\Omega}$. Of these, we have M_{int} interior collocation points $\{(x_m, y_m)\}_{m=1}^{M_{\text{int}}}$ and M_{bry} boundary collocation points $\{(x_m, y_m)\}_{m=M_{\text{int}}+1}^{M_{\text{int}}+M_{\text{bry}}}$. We take $M = M_{\text{int}} + M_{\text{bry}}$.

Note that the number of centres is normally taken to be less than the number of collocation points. The coefficients $\{a_n\}_{n=1}^{N+K}$ in equation (2.5) are determined from the collocation equations

$$\mathcal{L}u_N(x_m, y_m) = f(x_m, y_m), \quad m = 1, \dots, M_{\text{int}}, \quad (2.8a)$$

$$\mathcal{B}u_N(x_m, y_m) = g(x_m, y_m), \quad m = M_{\text{int}} + 1, \dots, M_{\text{int}} + M_{\text{bry}}. \quad (2.8b)$$

In addition to (2.8a)-(2.8b) we impose the standard insolvency conditions [9, Chapter 6], see also [3],

$$\sum_{n=1}^{N_{\text{int}}} a_n \{\mathcal{L}p_k\}(x_n, y_n) = 0, \quad k = 1, \dots, K, \quad (2.8c)$$

and

$$\sum_{n=N_{\text{int}}+1}^N a_n \{\mathcal{B}p_k\}(x_n, y_n) = 0, \quad k = 1, \dots, K. \quad (2.8d)$$

In fact we may combine (2.8c) and (2.8d) as

$$\sum_{n=1}^{N_{\text{int}}} a_n \{\mathcal{L}p_k\}(x_n, y_n) + \sum_{n=N_{\text{int}}+1}^N a_n \{\mathcal{B}p_k\}(x_n, y_n) = 0, \quad k = 1, \dots, K. \quad (2.8e)$$

We have $M + K$ equations in $N + K$ unknown coefficients $\mathbf{a} = [a_1, a_2, \dots, a_{N+K}]^T$ and we therefore take $M \geq N$.

In case the shape parameter is included in the unknowns then the number of unknowns becomes $N + K + 1$ consisting of \mathbf{a} and c and we need to take $M \geq N + 1$.

Since the operator \mathcal{L} is nonlinear, the system of $M + K$ equations (2.8a), (2.8b) and (2.8e) is nonlinear and can be written in the form

$$\mathbf{F}(\mathbf{a}, c) := \begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_{M+K} \end{bmatrix} = \begin{bmatrix} \mathcal{L}u_N(x_1, y_1) - f(x_1, y_1) \\ \vdots \\ \mathcal{L}u_N(x_{M_{\text{int}}}, y_{M_{\text{int}}}) - f(x_{M_{\text{int}}}, y_{M_{\text{int}}}) \\ \mathcal{B}u_N(x_{M_{\text{int}}+1}, y_{M_{\text{int}}+1}) - g(x_{M_{\text{int}}+1}, y_{M_{\text{int}}+1}) \\ \vdots \\ \mathcal{B}u_N(x_M, y_M) - g(x_M, y_M) \\ \sum_{n=1}^{N_{\text{int}}} a_n \{\mathcal{L}p_1\}(x_n, y_n) + \sum_{n=N_{\text{int}}+1}^N a_n \{\mathcal{B}p_1\}(x_n, y_n) \\ \vdots \\ \sum_{n=1}^{N_{\text{int}}} a_n \{\mathcal{L}p_K\}(x_n, y_n) + \sum_{n=N_{\text{int}}+1}^N a_n \{\mathcal{B}p_K\}(x_n, y_n) \end{bmatrix} = \mathbf{0}. \quad (2.9)$$

The nonlinear system (2.9) may be solved with the MATLAB[®] [21] optimization toolbox routine `fsolve`. The algorithm used is either a trust-region-dogleg/reflective or the Levenberg-Marquardt algorithm. The routine `fsolve` does not require the user to provide the gradient. When the system is not square, i.e. the number of equations is larger than the number of unknowns, which is the case in the current problem, the routine uses the Levenberg-Marquardt algorithm.

Alternatively, the solution of the nonlinear system (2.9) may be recast as a nonlinear least squares minimization problem by setting

$$S(\mathbf{a}, c) := \sum_{m=1}^{M+K} F_m^2, \quad (2.10)$$

where the $\{F_m\}_{m=1}^{M+K}$ are defined in (2.9). The functional $S(\mathbf{a}, c)$ is minimized using the MATLAB[®] [21] optimization toolbox routine `lsqnonlin` which solves nonlinear least squares problems using, like `fsolve`, either a subspace trust region method or the Levenberg-Marquardt algorithm. The routine `lsqnonlin` does not require the user to provide the gradient and, in addition, it offers the option of imposing lower and upper bounds on the elements of the vector of unknowns $\mathbf{x} = [\mathbf{a}, c]^T$ through the vectors `lb` and `ub`. We can thus easily impose the constraints on the values of the coefficients and in particular on the values of the shape parameter. It should be noted that both `lsqnonlin` and `fsolve` might give only local solutions. Note that when the shape parameter c is not taken to be one of the unknowns, $\mathbf{F}(\mathbf{a}, c)$ and $S(\mathbf{a}, c)$ become $\mathbf{F}(\mathbf{a})$ and $S(\mathbf{a})$, respectively. The MATLAB[®] nonlinear solvers perform a number of iterations that generally depends on a solver's stopping criteria including prescribed tolerances and other conditions. The routine stops when one of the criteria is met. The stopping criteria for `fsolve` and `lsqnonlin` may be summarized by the following user-prescribed quantities (i.e. the iterative process stops when these are reached):

- (i) the maximum number of iterations that the algorithm performs (`MaxIter`),

- (ii) the maximum number of evaluations of the objective function (`MaxFunEvals`),
- (iii) the termination tolerance for the vector of unknowns \mathbf{x} (`TolX`),
- (iv) the termination tolerance for the objective function value (`TolFun`).

On the other hand, an important optimality measure is the **first-order optimality** which is a measure that shows how close \mathbf{x} is to being optimal. It is a necessary condition but not a sufficient one. This measure must be zero at a minimum, but on the other hand a final \mathbf{x} is not necessarily a minimum. The definition of first-order optimality differs for different algorithms. For further details see e.g. [23].

In the current study we set `TolX=TolFun=10-14`, `MaxFunEvals=2000000` and controlled the convergence with appropriate values of `MaxIter`. In most cases the process stopped when `MaxIter` was exhausted but in some instances the tolerance `TolX` was reached.

After extensive experimentation with both `lsqnonlin` and `fsolve` we concluded that their performance for the problems considered in this study is very similar. The results presented in the first four examples were obtained with `fsolve`, while in Example 5, `lsqnonlin` is used.

In all subsequent examples, the number of iterations used is denoted by `niter`. The initial value of the shape parameter is denoted by c_0 and we took the initial value of \mathbf{a} to be $\mathbf{a}_0 = \mathbf{0}$.

3. NUMERICAL EXAMPLES FOR SECOND ORDER PROBLEMS

In the numerical examples considered in this section, we calculated the approximate solution $u_{\mathbf{N}}$ at L test points on a grid in $\bar{\Omega}$. In the case of Examples 1 and 2 the analytical solution is known. Hence, we calculated the maximum relative error E defined by

$$E = \frac{\|u - u_{\mathbf{N}}\|_{\infty, \bar{\Omega}}}{\|u\|_{\infty, \bar{\Omega}}} \quad (3.1)$$

and the root mean square error (RMSE) \mathcal{E} from

$$\mathcal{E} = \sqrt{\frac{1}{L} \sum_{\ell=1}^L [u(x_{\ell}, y_{\ell}) - u_{\mathbf{N}}(x_{\ell}, y_{\ell})]^2}. \quad (3.2)$$

In some cases, for comparison purposes, as well as E and \mathcal{E} , we calculated the maximum error e defined as

$$e = \|u - u_{\mathbf{N}}\|_{\infty, \bar{\Omega}}. \quad (3.3)$$

In the numerical examples presented in this work, we used MQ RBFs (2.6) and PS (2.7).

3.1. Example 1. We first consider an example from [26, 27] where the governing equation is

$$\mathcal{L}u = \Delta u - 4u^3 = 0 \quad \text{in } \Omega, \quad (3.4)$$

subject to Dirichlet boundary conditions corresponding to the exact solution

$$u(x, y) = \frac{1}{4 + x + y}.$$

The computational domain Ω is peanut shaped and its boundary $\partial\Omega$ is defined parametrically by

$$x = r(\vartheta) \cos \vartheta, \quad y = r(\vartheta) \sin \vartheta, \quad \text{where } r(\vartheta) = 0.3\sqrt{\cos 2\vartheta + \sqrt{1.1 - \sin^2 2\vartheta}}, \quad 0 \leq \vartheta \leq 2\pi. \quad (3.5)$$

A uniform distribution of the boundary collocation points is obtained as follows, see e.g [19]. We first calculate the length of the boundary curve from

$$S = \int_0^{2\pi} \sqrt{r(\vartheta)^2 + r'(\vartheta)^2} d\vartheta, \quad (3.6)$$

which may be evaluated using the MATLAB[®] routine `quadl` which evaluates an integral using adaptive Lobatto quadrature within a user-prescribed accuracy. Since there will be M_{bry} collocation points on the boundary, we take the length of each subsegment to be S/M_{bry} . The angles ϑ_k which give such a distribution are obtained by choosing $\vartheta_1 = 0$ and then solving, serially for $k = 1, \dots, M_{\vartheta} - 1$, the nonlinear equations

$$F(t) = \sqrt{(r(t) \cos t - r(\vartheta_k) \cos \vartheta_k)^2 + (r(t) \sin t - r(\vartheta_k) \sin \vartheta_k)^2} - \frac{S}{M_{\text{bry}}} = 0, \quad (3.7)$$

to yield $t = \vartheta_k, k = 2, \dots, M_{\text{bry}}$, respectively. The solution of the nonlinear equations may be carried out using the MATLAB[®] routine `fzero`. The angles $\vartheta_k, k = 1, \dots, M_{\text{bry}}$, define equally spaced points on the boundary curve and the boundary collocation points are now distributed as follows:

$$(x_{M_{\text{int}}+i}, y_{M_{\text{int}}+i}) = 0.3\sqrt{\cos 2\vartheta_i + \sqrt{1.1 - \sin^2 2\vartheta_i}} (\cos \vartheta_i, \sin \vartheta_i), \quad i = 1, \dots, M_{\text{bry}}.$$

The interior collocation points $(x_i, y_i), i = 1, \dots, M_{\text{int}}$, are taken to be the *Halton* points [9, Appendix A.1]. The corresponding boundary and interior centres are obtained in a similar way. Finally, we chose a set of $L = 300$ interior Halton points to be the set of test points. A typical distribution of collocation points is presented in Figure 1.

In Table 1 we present some typical numerical results for various \mathbf{p} using MQ RBFs. We observe that as we increase the number of auxiliary basis functions the accuracy improves considerably. The corresponding results obtained using the PS r^3 are presented in Table 2. While it is well-known that the convergence rate of MQ is superior to that of r^3 , from Tables 1 and 2, their accuracy is similar when higher degree polynomials are included. This is an indication that in the computational process the higher degree polynomials have prevailed over the RBFs in expansion (2.5).

\mathbf{p}	c_0	c	niter	CPU (secs)	E	\mathcal{E}
0	4	1.665	51	4.31	1.130(-3)	8.315(-5)
4	4	3.489	51	4.80	1.158(-6)	1.271(-7)
8	4	3.992	26	3.42	4.381(-11)	2.160(-12)
12	4	4.000	25	4.55	8.224(-15)	4.887(-16)

TABLE 1. Example 1: Results for various \mathbf{p} using MQ with $M_{\text{int}}=200$, $M = 300$, $N_{\text{int}} = 50$, $N = 70$.

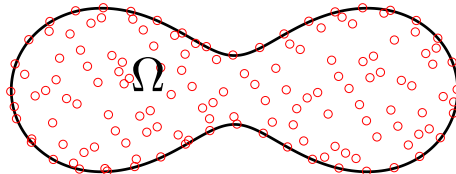


FIGURE 1. Example 1. Computational domain and typical distribution of collocation points.

p	niter	CPU (secs)	E	\mathcal{E}
0	15	0.40	4.064(-2)	5.382(-3)
4	7	0.28	2.468(-6)	1.842(-7)
8	17	0.61	3.138(-10)	2.296(-11)
12	26	1.04	3.133(-15)	2.320(-16)

TABLE 2. Example 1: Results for various p using the PS r^3 and $M_{\text{int}} = 200$, $M = 300$, $N_{\text{int}} = 50$, $N = 70$.

3.2. **Example 2.** We next consider an example from [7, 10] where

$$\mathcal{L}u = -\varepsilon^2 \Delta u - u + u^3 \quad \text{in } \Omega,$$

subject to the homogeneous Dirichlet boundary condition

$$\mathcal{B}u = u = 0 \quad \text{on } \partial\Omega,$$

where $\Omega = (0, 1) \times (0, 1)$, the exact solution is given by

$$u(x, y) = (1 + e^{-1/\varepsilon} - e^{-x/\varepsilon} - e^{(x-1)/\varepsilon}) (1 + e^{-1/\varepsilon} - e^{-y/\varepsilon} - e^{(y-1)/\varepsilon}), \quad (3.8)$$

and ε is a known constant. Clearly, the function f in (2.1a) is obtained by calculating $\mathcal{L}u$ for u given in (3.8).

We chose M_{bry} uniformly distributed collocation points on the boundary. We also chose M_{int} interior collocation points using the Halton points. Similarly, we chose N_{bry} boundary centres and N_{int} interior centres. Finally, we chose a set of $L = 50$ interior Halton points to be the set of test points. A typical distribution of collocation points is presented in Figure 2.

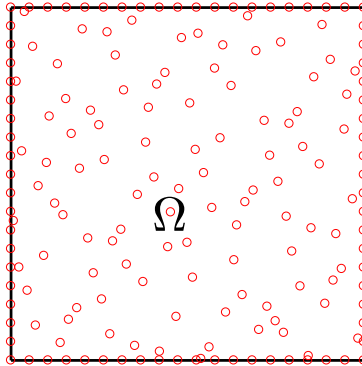


FIGURE 2. Example 2. Computational domain and typical distribution of collocation points.

3.2.1. *Case $\varepsilon = 1$.* In this case the solution is quite flat and convergence is rapid. In Table 3 we present typical results obtained using MQ RBFs with various \mathbf{p} . As the number of auxiliary polynomial basis functions increases, the convergence becomes more rapid.

\mathbf{p}	c_0	c	niter	CPU (secs)	E	\mathcal{E}
0	4	3.098	51	11.85	2.018(-1)	2.427(-3)
4	4	3.209	51	12.96	4.436(-4)	3.321(-6)
8	4	3.784	51	15.63	1.330(-8)	8.000(-11)
12	4	3.944	26	10.11	1.844(-14)	1.239(-16)

TABLE 3. Example 2: Results for $\varepsilon = 1$ with $M_{\text{int}}=200$, $M=276$, $N_{\text{int}} = 100$, $N = 140$.

3.2.2. *Case $\varepsilon = 0.25$.* In this case the solution is not as flat and convergence is no as rapid as for $\varepsilon = 1$. In Table 4 we present typical results obtained using MQ RBFs with various \mathbf{p} . Again, as the number of auxiliary polynomial basis functions increases the convergence becomes more rapid. A larger value of c_0 is taken and more iterations are required to obtain satisfactory accuracy.

\mathbf{p}	c_0	c	niter	CPU (secs)	E	\mathcal{E}
0	100	48.863	101	24.32	1.429(-1)	3.392(-2)
4	100	141.612	101	26.97	8.564(-3)	1.837(-3)
8	100	98.024	39	12.59	1.232(-4)	3.265(-5)
12	100	135.346	101	41.18	2.263(-6)	3.596(-7)

TABLE 4. Example 2: Results for $\varepsilon = 0.25$ with $M_{\text{int}}=200$, $M=276$, $N_{\text{int}} = 100$, $N = 140$.

3.2.3. *Case $\varepsilon = 0.1$.* In this more difficult case, we need to take more degrees of freedom as well as many more iterations as convergence is much slower. In Table 5 we present results obtained using MQ RBFs with various \mathbf{p} and different numbers of degrees of freedom. From this table, as expected because of the increased number of degrees of freedom and iterations, we observe that the solution of the problem is considerably computationally more expensive. As the number of auxiliary polynomial basis functions increases the convergence becomes more rapid.

M_{int}	M	N_{int}	N	\mathbf{p}	c_0	c	niter	CPU (secs)	E	\mathcal{E}
300	376	150	190	8	100	174.581	201	151.72	1.633	6.365(-1)
				10	100	161.987	201	188.44	2.133(-2)	5.128(-3)
				12	100	185.140	201	181.36	6.027(-3)	1.717(-3)
300	376	150	190	8	100	492.396	601	441.69	1.615	6.198(-1)
				10	100	413.713	601	463.22	1.275(-2)	2.610(-3)
				12	100	67.810	601	510.68	1.666(-3)	5.256(-4)
400	496	200	256	8	100	187.601	201	420.82	1.361(-2)	4.402(-3)
				10	100	130.277	201	444.85	1.274(-2)	5.143(-3)
				12	100	192.610	201	506.45	1.248(-2)	2.964(-3)
400	496	200	256	8	100	561.853	601	1439.18	9.984(-3)	1.903(-3)
				10	100	326.881	601	1419.78	7.297(-3)	1.279(-3)
				12	100	119.708	601	1402.47	2.399(-3)	4.618(-4)

TABLE 5. Example 2: Results for $\varepsilon = 0.1$.

For this difficult case, we consider an alternative approach by fixing the value of the shape parameter and not including it as one of the unknowns in the nonlinear solver. We first examined the effect of the number of iterations on the accuracy. As can be observed from Table 6, the accuracy remains practically the same after 20 iterations and, hence, we shall limit the number of iterations to no more than 50 (for this example).

Next, we study the effect of the value of the MQ shape parameter on the accuracy when a high degree polynomial basis is used in expansion (2.5). As may be observed from Table 7, the accuracy does not appear to depend much on the value of the MQ shape parameter. This provides greater

niter	CPU (secs)	E	\mathcal{E}
10	2.01	5.851(-3)	6.213(-4)
20	3.86	5.273(-3)	3.935(-4)
50	9.37	4.910(-3)	3.372(-4)
100	17.99	4.876(-3)	3.364(-4)
200	35.41	4.816(-3)	3.328(-4)
300	53.03	4.760(-3)	3.296(-4)

TABLE 6. Example 2: Results of the number of iterations and the accuracy for $\varepsilon = 0.1$ with $c = 100$, $\mathbf{p} = 12$, $M_{\text{int}}=400$, $M=496$, $N_{\text{int}} = 200$, $N = 256$.

c	E	\mathcal{E}
50	6.918(-3)	4.056(-4)
100	4.910(-3)	3.372(-4)
150	3.770(-3)	3.112(-4)
200	3.488(-3)	3.139(-4)
250	2.853(-3)	2.762(-4)
300	5.340(-3)	7.023(-4)

TABLE 7. Example 2: The shape parameter versus the accuracy for $\varepsilon = 0.1$ with $\mathbf{p} = 12$, $\mathbf{M}_{\text{int}}=400$, $\mathbf{M}=496$, $\mathbf{N}_{\text{int}} = 200$, $\mathbf{N} = 256$.

freedom in choosing the value of the shape parameter without affecting the accuracy. The benefit of coupling RBFs and polynomial basis functions in (2.5) is thus mutual. On the one hand, the presence of RBFs helps to contain the notorious ill-conditioning problems occurring when high degree polynomials are used, while, on the other hand, the inclusion of the polynomials facilitates the task of selecting an appropriate value of the shape parameter in the RBFs. Overall, in terms of accuracy, the results obtained by fixing the shape parameter are similar to the ones obtained in Table 5. The only difference is clearly the computational cost. The inclusion of the value of the shape parameter as one of the unknowns renders an already difficult nonlinear problem even more difficult which requires much more CPU time. For the easier cases $\varepsilon = 1$ and 0.25 , since the process converges very fast, there is no significant difference in terms of either accuracy or efficiency between the two approaches.

3.3. Example 3. Next, we consider an elastoplastic torsion problem in the case of a prismatic rod [2, 17, 22] with a square cross section. The governing equation for Prandtl's stress function ψ is proposed in [17] and its formulation in non-dimensional coordinates is

$$\mathcal{L}\psi = 0 \quad \text{in } \Omega, \quad (3.9)$$

where

$$\mathcal{L}\psi = \Delta\psi + \frac{1}{1 + \kappa \tilde{\tau}^n} \left\{ 2\tilde{\theta} + \kappa n \tilde{\tau}^{n-2} \left[\left(\frac{\partial\psi}{\partial x} \right)^2 \frac{\partial^2\psi}{\partial x^2} + 2 \frac{\partial\psi}{\partial x} \frac{\partial\psi}{\partial y} \frac{\partial^2\psi}{\partial x\partial y} + \left(\frac{\partial\psi}{\partial y} \right)^2 \frac{\partial^2\psi}{\partial y^2} \right] \right\}$$

and where $\tilde{\theta}$ is the non-dimensional angle of twist, n , κ are the non-dimensional material parameters and $\tilde{\tau} = \tilde{\tau}(x, y)$ is the non-dimensional resultant shear stress given by

$$\tilde{\tau} = \left[\left(\frac{\partial\psi}{\partial x} \right)^2 + \left(\frac{\partial\psi}{\partial y} \right)^2 \right]^{1/2}.$$

In the torsion problem considered, the lateral surface of the bar is stress-free. Hence, the boundary curve is a line on which the stress function is constant. For a simply connected cross section, we may take $\psi = 0$ on $\partial\Omega$. The computational domain Ω is shown in Figure 3, where a is a characteristic dimension of the cross section (we subsequently take $a = 1$). Note that because of

symmetry we may consider only one repeated element of it (e.g. the region $ABCD$ in Figure 3). In such a case, the governing equation (3.9) is subject to the boundary conditions

$$\psi = 0 \quad \text{on } BC \text{ and } CD, \quad (3.10)$$

$$\frac{\partial \psi}{\partial y} = 0 \quad \text{on } AB \quad \text{and} \quad \frac{\partial \psi}{\partial x} = 0 \quad \text{on } AD. \quad (3.11)$$

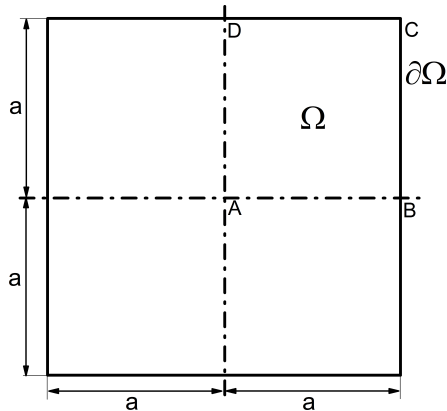


FIGURE 3. Example 3. The square cross section of the prismatic bar.

Having determined the stress function ψ , we can compute the non-dimensional torsional moment defined as

$$\widetilde{M}_T = 2 \iint_{\Omega} \psi \, dx \, dy. \quad (3.12)$$

We shall present the numerical results obtained with the proposed algorithm for $n = 3.83$, $\kappa = 0.023552997$ and $0.1 \leq \tilde{\theta} \leq 5$. The torsional moment is then computed for each $\tilde{\theta}$ considered.

We chose an $M \times M$ grid of interior collocation points in Ω together with $4M$ boundary collocation points and, similarly, an $N \times N$ grid of interior centre points in Ω and $4N$ boundary centre points. In this way the total number of collocation points is equal to $\mathbf{M} = \mathbf{M}_{\text{int}} + \mathbf{M}_{\text{bry}}$, where $\mathbf{M}_{\text{int}} = M^2$ and $\mathbf{M}_{\text{bry}} = 4M$. Similarly, we have $\mathbf{N} = \mathbf{N}_{\text{int}} + \mathbf{N}_{\text{bry}}$ centres, where $\mathbf{N}_{\text{int}} = N^2$ and $\mathbf{N}_{\text{bry}} = 4N$. For the set of $L = L_{\text{int}} + L_{\text{bry}}$ test points we took $L_{\text{int}} = 224$ interior Halton points and $L_{\text{bry}} = 76$ uniformly distributed boundary points and we chose $c_0 = 3$.

The approximate value of the integral given in (3.12) was obtained with the MATLAB[®] build-in routine `trapz` which approximates integrals using the trapezoidal rule.

After computing the solution ψ of problem (3.9)–(3.11), we calculated the maximum error e (3.3) and the RMSE \mathcal{E} (3.2) at a total number L_{bry} of test points located on the boundary. The boundary value problem considered was also solved with the Finite Element Method (FEM) package COMSOL Multiphysics[®] software [6]. Thus, we could measure the difference between the approximate solutions obtained with the proposed method and the FEM at a total number L

of test points in $\bar{\Omega}$. We used the formulæ (3.3) and (3.2) and denote by d and \mathcal{D} the maximum difference and the root mean square difference, respectively (cf. e and \mathcal{E} in (3.3) and (3.2), respectively). The reference solution u in (3.2)–(3.3) is the approximate solution obtained with the FEM. Finally, we computed the relative difference $D_{\widetilde{M}_T} = |\widetilde{M}_T^{(K)} - \widetilde{M}_T^{(F)}| / |\widetilde{M}_T^{(F)}|$, where $\widetilde{M}_T^{(K)}$ and $\widetilde{M}_T^{(F)}$ are the torsion moments obtained with the nonlinear Kansa method and the FEM, respectively.

We subsequently performed two kinds of tests. First, we took a sequence of twist angles $\widetilde{\theta}$ and computed the appropriate values of the stress function ψ and corresponding torsional moments \widetilde{M}_T . We applied the proposed Kansa method for $M = 15$, $N = 14$, $\mathbf{M} = 285$, $\mathbf{N} = 252$, $\mathbf{p} = 0, 2, 4, \dots, 20$, and $\mathbf{niter} = 50$. The results obtained for various values of \mathbf{p} are presented in Table 8 and for $\mathbf{p} = 20$ in Table 9. In Figure 4 we can see the torsional moment \widetilde{M}_T corresponding to given values of the twist angle $\widetilde{\theta}$ obtained with both the nonlinear Kansa method for $\mathbf{K} = 0$ and the FEM. The corresponding plots obtained with $\mathbf{p} = 0, 2, 4, \dots, 20$, are indistinguishable from Figure 4. Furthermore, the distributions of the stress ψ in the prismatic rod for the twist angles $\widetilde{\theta} = 0.5, 1, 2, 3, 4, 5$, obtained using the proposed Kansa method for $\mathbf{p} = 12$ are presented in Figure 5.

Secondly, we examined how increasing the maximum number of iterations \mathbf{niter} and the number of degrees of freedom of the proposed Kansa method for $\mathbf{p} = 12$ influences the accuracy of the solution. The results corresponding to the selected values of the theta angle $\widetilde{\theta} = 1, 3, 5$ are given in Table 10.

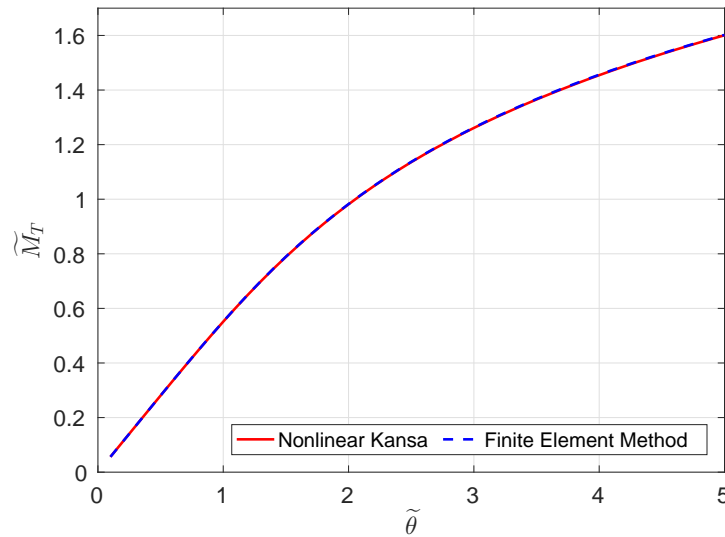


FIGURE 4. Example 3. The torsional moment \widetilde{M}_T vs. the twist angle $\widetilde{\theta}$ obtained with the nonlinear Kansa method and the FEM.

(a) $\tilde{\theta} = 1$

p	c	$\widetilde{M}_T^{(K)}$	$\widetilde{M}_T^{(F)}$	$D_{\widetilde{M}_T}$	e	\mathcal{E}	d	\mathcal{D}
0	2.653	0.552666	0.551872	1.438(-3)	3.646(-3)	7.777(-4)	3.646(-3)	7.073(-4)
4	2.938	0.552638	0.551872	1.387(-3)	2.029(-3)	4.811(-4)	2.029(-3)	5.818(-4)
8	2.996	0.550787	0.551872	1.967(-3)	5.028(-4)	1.152(-4)	2.449(-3)	8.261(-4)
12	3.000	0.551762	0.551872	2.009(-4)	2.196(-4)	4.871(-5)	2.196(-4)	7.644(-5)
16	3.000	0.551842	0.551872	5.493(-5)	1.778(-4)	3.903(-5)	1.778(-4)	2.388(-5)
20	3.000	0.551860	0.551872	2.243(-5)	1.062(-4)	2.188(-5)	1.062(-4)	1.155(-5)

(b) $\tilde{\theta} = 3$

p	c	$\widetilde{M}_T^{(K)}$	$\widetilde{M}_T^{(F)}$	$D_{\widetilde{M}_T}$	e	\mathcal{E}	d	\mathcal{D}
0	2.941	1.263626	1.262388	9.810(-4)	1.155(-2)	2.504(-3)	1.155(-2)	1.795(-3)
4	2.955	1.262359	1.262388	2.269(-5)	9.501(-3)	2.058(-3)	9.501(-3)	1.147(-3)
8	3.011	1.261203	1.262388	9.388(-4)	7.517(-3)	1.594(-3)	7.517(-3)	1.329(-3)
12	3.016	1.261580	1.262388	6.404(-4)	1.257(-2)	2.523(-3)	1.257(-2)	1.761(-3)
16	3.019	1.261912	1.262388	3.767(-4)	8.785(-3)	1.740(-3)	8.785(-3)	1.198(-3)
20	3.034	1.263486	1.262388	8.700(-4)	6.643(-3)	1.683(-3)	6.643(-3)	1.151(-3)

(c) $\tilde{\theta} = 5$

p	c	$\widetilde{M}_T^{(K)}$	$\widetilde{M}_T^{(F)}$	$D_{\widetilde{M}_T}$	e	\mathcal{E}	d	\mathcal{D}
0	2.999	1.602900	1.601834	6.655(-4)	2.620(-2)	4.984(-3)	2.620(-2)	2.829(-3)
4	3.041	1.601989	1.601834	9.643(-5)	2.522(-2)	4.849(-3)	2.522(-2)	2.688(-3)
8	3.010	1.602746	1.601834	5.695(-4)	3.019(-2)	5.668(-3)	3.019(-2)	3.270(-3)
12	3.005	1.602640	1.601834	5.029(-4)	3.174(-2)	6.249(-3)	3.174(-2)	3.568(-3)
16	3.007	1.602738	1.601834	5.644(-4)	3.207(-2)	6.331(-3)	3.207(-2)	3.620(-3)
20	3.006	1.600608	1.601834	7.653(-4)	2.760(-2)	5.310(-3)	2.760(-2)	3.074(-3)

TABLE 8. Example 3: The relative difference $D_{\widetilde{M}_T}$, the error measure for the satisfaction of the boundary conditions and the difference between the solution obtained for selected values of the twist angle $\tilde{\theta}$ with the nonlinear Kansa method and the FEM vs. the degree p of the auxiliary polynomial.

$\tilde{\theta}$	c	$\widetilde{M}_T^{(K)}$	$\widetilde{M}_T^{(F)}$	$d_{rel}\widetilde{M}_T$	e	\mathcal{E}	d	\mathcal{D}
0.1	3.000	0.056230	0.056231	1.530(-5)	5.190(-6)	1.266(-6)	5.190(-6)	7.153(-7)
0.5	3.000	0.280732	0.280738	2.344(-5)	5.975(-5)	1.252(-5)	5.975(-5)	6.527(-6)
1.0	3.000	0.551860	0.551872	2.243(-5)	1.062(-4)	2.188(-5)	1.062(-4)	1.155(-5)
2.0	3.011	0.982323	0.982374	5.146(-5)	3.108(-4)	8.032(-5)	5.549(-4)	1.895(-4)
3.0	3.034	1.263486	1.262388	8.700(-4)	6.643(-3)	1.683(-3)	6.643(-3)	1.151(-3)
4.0	3.011	1.457466	1.456156	8.994(-4)	1.778(-2)	3.778(-3)	1.778(-2)	2.336(-3)
5.0	3.006	1.600608	1.601834	7.653(-4)	2.760(-2)	5.310(-3)	2.760(-2)	3.074(-3)

TABLE 9. Example 3: The torsional moment \widetilde{M}_T , the shape parameter c , the error measure for the satisfaction of the boundary conditions and the difference between the solution obtained with the nonlinear Kansa method and the FEM for selected values of the twist angle $\tilde{\theta}$.

(a) $\tilde{\theta} = 1$									
M	N	M	N	niter	c	e	\mathcal{E}	d	\mathcal{D}
9	8	117	96	10	3.001	7.087(-4)	1.181(-4)	7.087(-4)	7.963(-5)
				30	3.001	6.903(-4)	1.142(-4)	6.903(-4)	7.632(-5)
				50	3.001	6.815(-4)	1.125(-4)	6.815(-4)	7.340(-5)
				100	3.001	6.750(-4)	1.113(-4)	6.750(-4)	7.189(-5)
17	16	357	320	10	3.000	3.540(-4)	1.018(-4)	3.540(-4)	5.961(-5)
				30	3.001	1.647(-4)	4.630(-5)	1.865(-4)	6.154(-5)
				50	3.001	1.514(-4)	4.270(-5)	1.568(-4)	5.928(-5)
				100	3.001	1.497(-4)	4.220(-5)	1.497(-4)	5.742(-5)
(b) $\tilde{\theta} = 3$									
M	N	M	N	niter	c	e	\mathcal{E}	d	\mathcal{D}
9	8	117	96	10	3.021	3.572(-3)	1.009(-3)	9.299(-3)	4.816(-3)
				30	3.136	5.522(-3)	1.234(-3)	2.810(-2)	9.977(-3)
				50	3.218	5.822(-3)	1.249(-3)	3.085(-2)	1.072(-2)
				100	3.349	1.838(-2)	3.624(-3)	5.636(-2)	1.743(-2)
17	16	357	320	10	3.005	1.785(-2)	3.655(-3)	1.785(-2)	2.059(-3)
				30	3.009	1.428(-2)	2.946(-3)	1.428(-2)	1.787(-3)
				50	3.011	1.318(-2)	2.717(-3)	1.318(-2)	1.768(-3)
				100	3.013	1.201(-2)	2.458(-3)	1.201(-2)	1.784(-3)
(c) $\tilde{\theta} = 5$									
M	N	M	N	niter	c	e	\mathcal{E}	d	\mathcal{D}
9	8	117	96	10	3.090	7.003(-2)	1.449(-2)	7.003(-2)	9.609(-3)
				30	3.130	4.583(-2)	1.178(-2)	4.583(-2)	6.948(-3)
				50	3.154	4.925(-2)	1.242(-2)	4.390(-2)	8.091(-3)
				100	3.185	5.255(-2)	1.246(-2)	4.174(-2)	9.064(-3)
17	16	357	320	10	3.001	3.722(-2)	7.513(-3)	3.722(-2)	4.478(-3)
				30	3.004	3.220(-2)	6.464(-3)	3.220(-2)	3.927(-3)
				50	3.005	3.215(-2)	6.443(-3)	3.215(-2)	3.900(-3)
				100	3.006	3.213(-2)	6.427(-3)	3.213(-2)	3.887(-3)

TABLE 10. Example 3: The error measure for the satisfaction of the boundary conditions and the difference between the solution obtained with the nonlinear Kansa method and the FEM for selected values of the twist angle $\tilde{\theta}$ vs. the maximum number of iterations **niter** and degrees of freedom M, N .

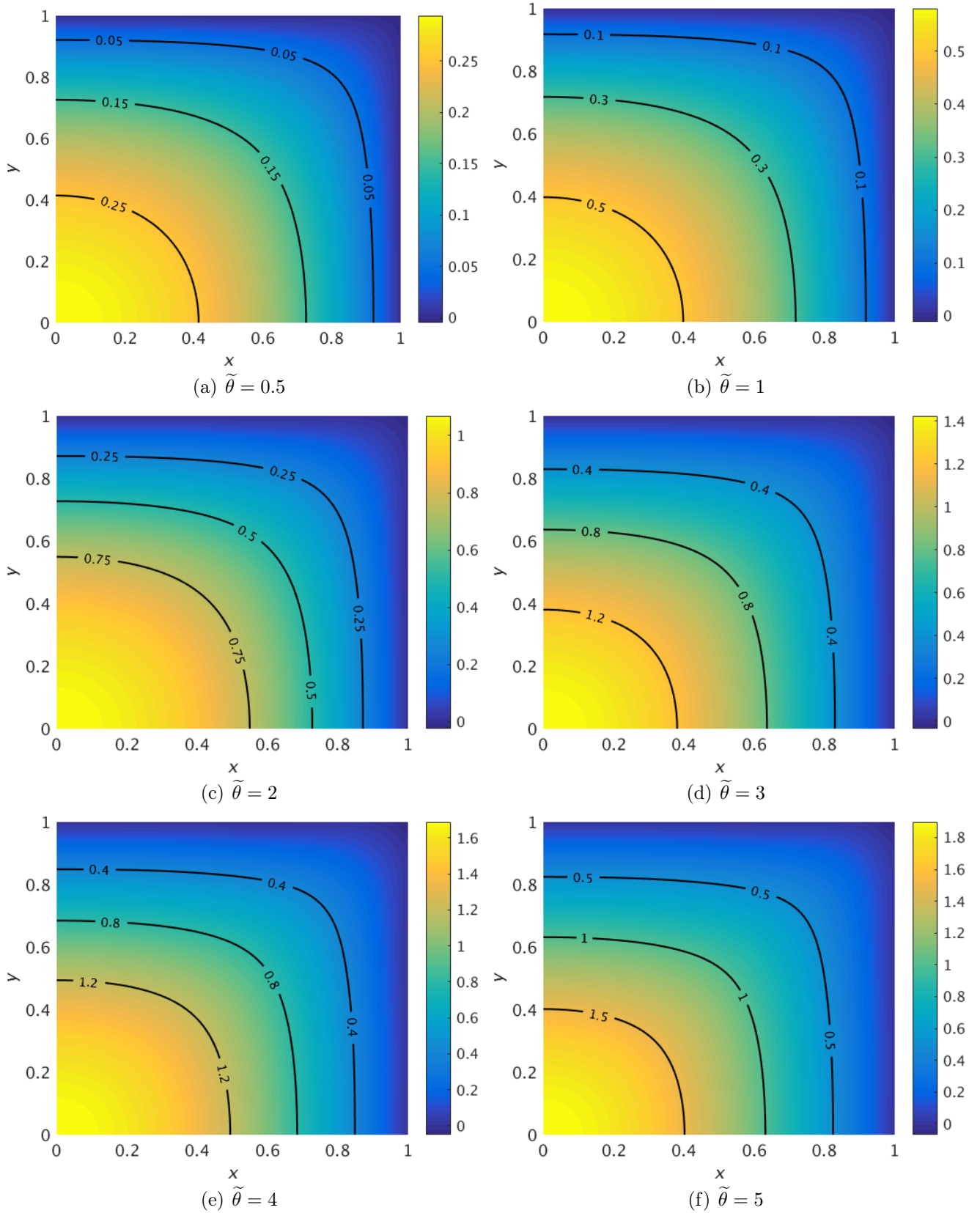


FIGURE 5. Example 3: The distribution of the stress ψ in the prismatic rod for selected values of the twist angle $\tilde{\theta}$.

4. FOURTH ORDER PROBLEMS

4.1. **The problem.** We now consider the boundary value problem in \mathbb{R}^2

$$\mathcal{L}u = f \quad \text{in } \Omega, \quad (4.1a)$$

subject to the boundary conditions

$$\mathcal{B}_1 u = g_1 \quad \text{and} \quad \mathcal{B}_2 u = g_2 \quad \text{on } \partial\Omega, \quad (4.1b)$$

where \mathcal{L} is a fourth order nonlinear elliptic operator and $\mathcal{B}_1, \mathcal{B}_2$ are linear (or nonlinear) operators describing the boundary conditions.

4.2. **The method.** We approximate the solution u of boundary value problem (4.1) by (2.5) and consider the same collocation points $\{(x_m, y_m)\}_{m=1}^M \in \bar{\Omega}$ as in the second order case, of which M_{int} are interior collocation points and M_{bry} are boundary collocation points. We shall also consider the interior centres $\{(x_n, y_n)\}_{n=1}^{N_{\text{int}}}$ and the boundary centres $\{(x_n, y_n)\}_{n=N_{\text{int}}+1}^{N_{\text{int}}+N_{\text{bry}}}$. Now, however, we take $\mathbf{N} = N_{\text{int}} + 2N_{\text{bry}}$.

The coefficients $\{a_n\}_{n=1}^{N+K}$ are determined from the collocation equations

$$\mathcal{L}u_{\mathbf{N}}(x_m, y_m) = f(x_m, y_m), \quad m = 1, \dots, M_{\text{int}}, \quad (4.2a)$$

$$\mathcal{B}_1 u_{\mathbf{N}}(x_m, y_m) = g_1(x_m, y_m), \quad m = M_{\text{int}} + 1, \dots, M_{\text{int}} + M_{\text{bry}}, \quad (4.2b)$$

$$\mathcal{B}_2 u_{\mathbf{N}}(x_m, y_m) = g_2(x_m, y_m), \quad m = M_{\text{int}} + 1, \dots, M_{\text{int}} + M_{\text{bry}}. \quad (4.2c)$$

In addition to (4.2a)-(4.2c) we impose the standard insolvency conditions

$$\sum_{n=1}^{N_{\text{int}}} a_n \{\mathcal{L} p_k\}(x_n, y_n) = 0, \quad k = 1, \dots, K, \quad (4.2d)$$

$$\sum_{n=N_{\text{int}}+1}^{N_{\text{int}}+N_{\text{bry}}} a_n \{\mathcal{B}_1 p_k\}(x_n, y_n) = 0, \quad k = 1, \dots, K, \quad (4.2e)$$

and

$$\sum_{n=N_{\text{int}}+N_{\text{bry}}+1}^{N_{\text{int}}+2N_{\text{bry}}} a_n \{\mathcal{B}_2 p_k\}(x_n, y_n) = 0, \quad k = 1, \dots, K. \quad (4.2f)$$

As before, we may combine (4.2d), (4.2e) and (4.2f) as

$$\begin{aligned} & \sum_{n=1}^{N_{\text{int}}} a_n \{\mathcal{L} p_k\}(x_n, y_n) + \sum_{n=N_{\text{int}}+1}^{N_{\text{int}}+N_{\text{bry}}} a_n \{\mathcal{B}_1 p_k\}(x_n, y_n) \\ & + \sum_{n=N_{\text{int}}+N_{\text{bry}}+1}^{N_{\text{int}}+2N_{\text{bry}}} a_n \{\mathcal{B}_2 p_k\}(x_n, y_n) = 0, \quad k = 1, \dots, K. \end{aligned} \quad (4.2g)$$

In order to ensure we have at least as many equations as unknowns we thus need to take $M_{\text{int}} + 2M_{\text{bry}} + K \geq N_{\text{int}} + 2N_{\text{bry}} + K$

In case the shape parameter is included in the unknowns then the number of unknowns becomes $N_{\text{int}} + 2N_{\text{bry}} + K + 1$ consisting of \mathbf{a} and c .

Since the operator \mathcal{L} is nonlinear, the system of $M_{\text{int}} + 2M_{\text{bry}} + K$ equations (4.2a), (4.2b), (4.2c) and (4.2g) is nonlinear and can be written in the form

$$\mathbf{F}(\mathbf{a}, c) := \begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_{M_{\text{int}}+2M_{\text{bry}}+K} \end{bmatrix}$$

$$= \begin{bmatrix} \mathcal{L}u_{\text{N}}(x_1, y_1) - f(x_1, y_1) \\ \vdots \\ \mathcal{L}u_{\text{N}}(x_{M_{\text{int}}}, y_{M_{\text{int}}}) - f(x_{M_{\text{int}}}, y_{M_{\text{int}}}) \\ \mathcal{B}_1 u_{\text{N}}(x_{M_{\text{int}}+1}, y_{M_{\text{int}}+1}) - g_1(x_{M_{\text{int}}+1}, y_{M_{\text{int}}+1}) \\ \vdots \\ \mathcal{B}_1 u_{\text{N}}(x_{M_{\text{int}}+M_{\text{bry}}}, y_{M_{\text{int}}+M_{\text{bry}}}) - g_1(x_{M_{\text{int}}+M_{\text{bry}}}, y_{M_{\text{int}}+M_{\text{bry}}}) \\ \mathcal{B}_2 u_{\text{N}}(x_{M_{\text{int}}+1}, y_{M_{\text{int}}+1}) - g_2(x_{M_{\text{int}}+1}, y_{M_{\text{int}}+1}) \\ \vdots \\ \mathcal{B}_2 u_{\text{N}}(x_{M_{\text{int}}+M_{\text{bry}}}, y_{M_{\text{int}}+M_{\text{bry}}}) - g_2(x_{M_{\text{int}}+M_{\text{bry}}}, y_{M_{\text{int}}+M_{\text{bry}}}) \\ \sum_{n=1}^{N_{\text{int}}} a_n \{\mathcal{L} p_1\}(x_n, y_n) + \sum_{n=N_{\text{int}}+1}^{N_{\text{int}}+N_{\text{bry}}} a_n \{\mathcal{B}_1 p_1\}(x_n, y_n) + \sum_{n=N_{\text{int}}+N_{\text{bry}}+1}^{N_{\text{int}}+2N_{\text{bry}}} a_n \{\mathcal{B}_2 p_1\}(x_n, y_n) \\ \vdots \\ \sum_{n=1}^{N_{\text{int}}} a_n \{\mathcal{L} p_K\}(x_n, y_n) + \sum_{n=N_{\text{int}}+1}^{N_{\text{int}}+N_{\text{bry}}} a_n \{\mathcal{B}_1 p_K\}(x_n, y_n) + \sum_{n=N_{\text{int}}+N_{\text{bry}}+1}^{N_{\text{int}}+2N_{\text{bry}}} a_n \{\mathcal{B}_2 p_K\}(x_n, y_n). \end{bmatrix} = \mathbf{0}. \quad (4.3)$$

Note that in equations (4.2f), (4.2g) and (4.3) we have used the convention $\{(x_n, y_n)\}_{n=N_{\text{int}}+N_{\text{bry}}+1}^{N_{\text{int}}+2N_{\text{bry}}} = \{(x_n, y_n)\}_{n=N_{\text{int}}+1}^{N_{\text{int}}+N_{\text{bry}}}$.

5. NUMERICAL EXAMPLES FOR FOURTH ORDER PROBLEMS

5.1. **Example 4.** We next consider the boundary value problem

$$\mathcal{L}u = \Delta^2 u - 96u^5 = 0 \quad \text{in } \Omega, \quad (5.1)$$

subject to boundary conditions for u and $\partial u/\partial n$ corresponding to the exact solution

$$u(x, y) = \frac{1}{4 + x + y}.$$

The computational domain Ω is peanut shaped and its boundary $\partial\Omega$ is defined by (3.5). The discretization of the domain is identical to the one described in Example 2. In the implementation of the method we shall need the outward unit normal vector to the boundary. This is defined by

$$\mathbf{n} = \frac{1}{\sqrt{r^2(\vartheta) + r'^2(\vartheta)}} \left(r'(\vartheta) \sin \vartheta + r(\vartheta) \cos \vartheta, r(\vartheta) \sin \vartheta - r'(\vartheta) \cos \vartheta \right)$$

where r is given in (3.5).

In Table 11 we present some typical numerical results for various \mathbf{p} using MQ RBFs. We observe that as we increase the number of auxiliary basis functions the accuracy improves considerably. The corresponding results obtained using the PS r^5 are presented in Table 12.

\mathbf{p}	c_0	c	niter	CPU (secs)	E	\mathcal{E}
0	4.2	1.245	51	12.01	5.678(-3)	1.166(-3)
4	4.0	3.067	51	14.79	1.059(-5)	1.286(-6)
8	4.0	3.855	51	19.07	6.800(-9)	4.623(-10)
12	4.0	4.000	20	10.76	3.419(-12)	2.111(-13)

TABLE 11. Example 4: Results for various numbers of degrees of freedom using MQ RBFs with $M_{\text{int}} = 200$, $M = 300$ $N_{\text{int}} = 50$, $N_{\text{int}} + N_{\text{bry}} = 70$.

\mathbf{p}	niter	CPU (secs)	E	\mathcal{E}
0	14	5.42	1.044(-1)	1.551(-2)
4	33	13.94	1.235(-5)	1.442(-6)
8	21	11.81	4.546(-7)	4.061(-8)
12	33	25.86	1.893(-10)	1.131(-11)

TABLE 12. Example 4: Results for various numbers of degrees of freedom using the PS r^5 with $M_{\text{int}} = 200$, $M = 300$ $N_{\text{int}} = 50$, $N_{\text{int}} + N_{\text{bry}} = 70$.

5.2. **Example 5.** We finally consider the stream–function formulation on the Navier–Stokes equations for laminar viscous flow in a wavy channel [16, 25, 28]. From [16] the governing equation for the stream–function ψ is

$$\mathcal{L}\psi = \Delta^2\psi - \text{Re} \left(\frac{\partial\psi}{\partial y} \frac{\partial\Delta\psi}{\partial x} - \frac{\partial\psi}{\partial x} \frac{\partial\Delta\psi}{\partial y} \right) = 0 \quad \text{in } \Omega, \quad (5.2a)$$

where Re is the Reynolds number, subject to the boundary conditions

$$\psi = 0 \quad \text{and} \quad \frac{\partial^2\psi}{\partial y^2} = 0 \quad \text{on } AB, \quad (5.2b)$$

$$\psi = 1, \quad \frac{\partial\psi}{\partial n} = 0 \quad \text{on } CD, \quad (5.2c)$$

$$\psi = \frac{3(H-E)^2 y - y^3}{2(H-E)^3} \quad \text{and} \quad \frac{\partial\psi}{\partial x} = 0 \quad \text{on } BC \text{ and } DA. \quad (5.2d)$$

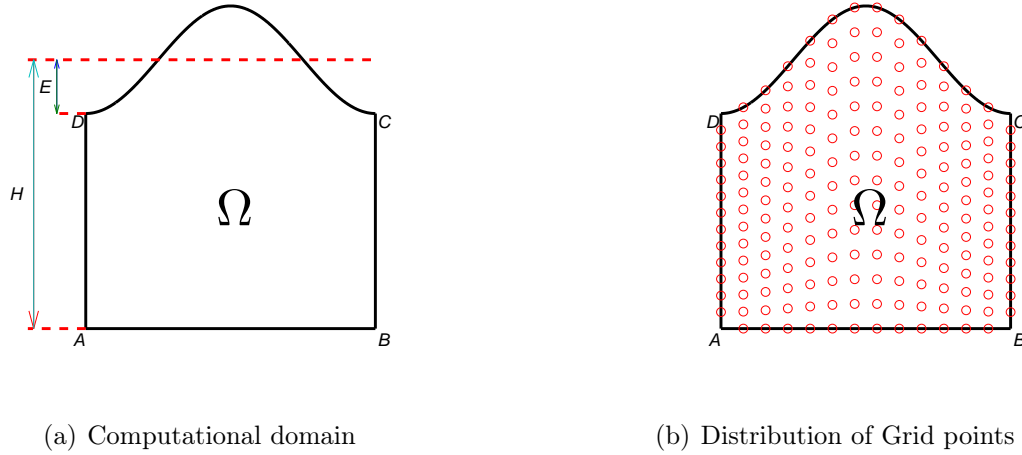


FIGURE 6. Example 5. The profiles of computational domain and typical distribution of collocation points.

The computational domain Ω is shown in Figure 6(a). Note that the length $AB = 1$ and the wavy surface CD is given by $y = H - E \cos(2\pi x)$. Therefore, in (5.2c)

$$\frac{\partial\psi}{\partial n} = \frac{1}{\sqrt{1 + 4\pi^2 E^2 \sin^2(2\pi x)}} \left(\frac{\partial\psi}{\partial y} - 2\pi E \sin(2\pi x) \frac{\partial\psi}{\partial x} \right).$$

We took $M = M_{\text{int}} + M_{\text{bry}}$ collocation points, where M_{int} is the number of interior points and $M_{\text{bry}} = M_{\text{bryc}} + 3M_{\text{brys}}$ is the number of boundary points. We denote by M_{bryc} the number of collocation points located on a curved boundary CD and by M_{brys} the number of collocation points located on each other side AB , BC and DA , respectively. The centres are distributed in

the similar way and we thus have a total number of $\mathbf{N} = \mathbf{N}_{\text{int}} + \mathbf{N}_{\text{bry}}$ centres, where \mathbf{N}_{int} is the number of interior centres and $\mathbf{N}_{\text{bry}} = \mathbf{N}_{\text{bryc}} + 3\mathbf{N}_{\text{brys}}$ is the number of boundary centres. A typical distribution of collocation points is presented in Figure 6(b). For the set of $L = L_{\text{int}} + L_{\text{bry}}$ test points we take $L_{\text{int}} = 224$ interior Halton points and $L_{\text{bry}} = 76$ uniformly distributed boundary points.

The solution of the problem (5.2) was obtained with the proposed Kansa method, where we used the logarithmic PS $r^2 \log r$. We took $\mathbf{M} = 281$ collocation points, $\mathbf{N} = 280$ centres, $\mathbf{p} = 20$ and `niter` = 200. The results obtained were compared with the corresponding results obtained using the FEM package COMSOL Multiphysics® and the RMSE \mathcal{E} calculated over a set of points in $\bar{\Omega}$ was found to be of $O(10^{-3})$. In Figure 7 we present the contour plots for the stream function for $\text{Re} = 20, 40, 60, 80$ and $H = 1, E = 0.2$. These are in good agreement with the corresponding figures presented in [16] obtained using a combination of the method of fundamental solutions and RBFs and a Picard iterative method and the figures obtained using the FEM package COMSOL Multiphysics®.

6. CONCLUSIONS

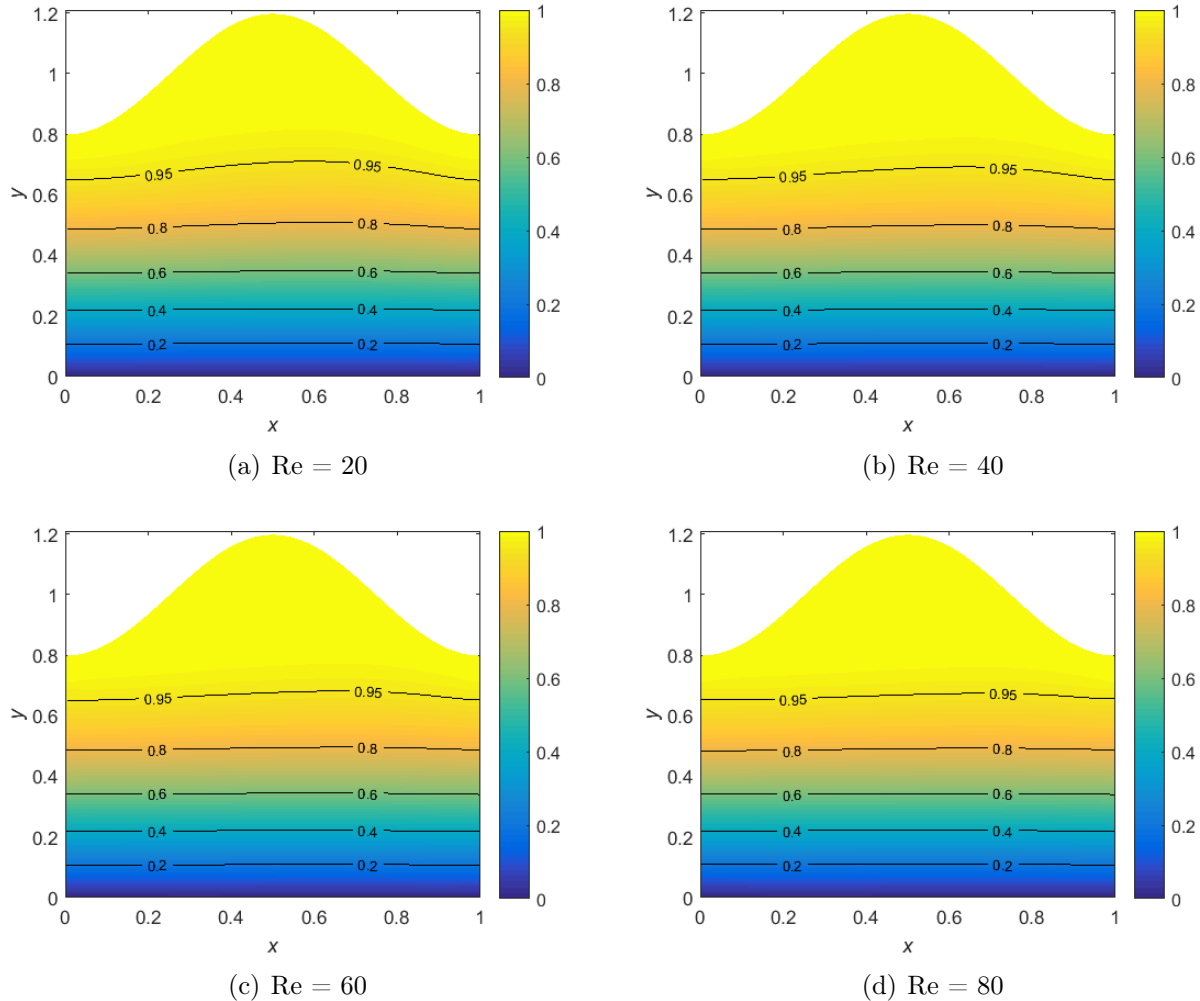
A Kansa–RBF collocation method has been applied for the solution of boundary value problems for second and fourth order nonlinear partial differential equations. The linear combination of RBFs in the approximation of the solution was enriched by linear combinations of polynomial basis functions (and the corresponding insolvency conditions). The Kansa–RBF discretization of these problems leads to systems of nonlinear equations which are solved using either the nonlinear MATLAB® routine `fsolve` or the nonlinear least–squares minimization MATLAB® routine `lsqnonlin`. The main advantage of the proposed technique is the ease with which it can be implemented while the results of several numerical experiments reveal that it leads to accurate numerical results with a relatively small number of iterations. An attempt to resolve the issue of obtaining an appropriate value of the shape parameter, which remains a challenging problem in the application of Kansa–RBF collocation methods, is made by taking the unknown value of the shape parameter to be part of the unknowns in `fsolve` or `lsqnonlin`. The extension of the proposed technique to three–dimensional problems will be the subject of future research.

ACKNOWLEDGEMENTS

The work of M. A. Jankowska was supported in part by the PLGrid Infrastructure.

REFERENCES

- [1] M. D. Buhmann, *Radial Basis Functions: Theory and Implementations*, Cambridge Monographs on Applied and Computational Mathematics, vol. 12, Cambridge University Press, Cambridge, 2003.
- [2] J. Chakrabarty, *Theory of Plasticity*, McGraw-Hill Companies, New York, 1987.
- [3] C. S. Chen, Y. C. Hon, and R. A. Schaback, *Scientific computing with radial basis functions*, Technical Report, Department of Mathematics, University of Southern Mississippi, Hattiesburg, MS 39406, USA, preprint, 2005.

FIGURE 7. Example 5: Streamlines for selected values of Re .

- [4] W. Chen, Z.-J. Fu, and C. S. Chen, *Recent Advances in Radial Basis Function Collocation Methods*, Springer Briefs in Applied Sciences and Technology, Springer, Heidelberg, 2014.
- [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] COMSOL Multiphysics® v. 5.2. www.comsol.com, COMSOL AB, Stockholm, Sweden.
- [7] G. E. Fasshauer, *Newton iteration with multiquadrics for the solution of nonlinear PDEs*, Comput. Math. Appl. **43** (2002), 423–438.
- [8] G. E. Fasshauer, *Solving Partial Differential Equations by Collocation with Radial Basis Functions: Surface Fitting and Multiresolution Methods*, Vanderbilt University Press, (1997), 131–138.
- [9] G. E. Fasshauer, *Meshfree Approximation Methods with MATLAB*, Interdisciplinary Mathematical Sciences, vol. 6, World Scientific Publishing Co. Pte. Ltd., Hackensack, NJ, 2007.

- [10] G. E. Fasshauer, E. C. Gartland and J. W. Jerome, *Newton iteration for partial differential equations and the approximation of the identity*, Numer. Algor. **25** (2000), 181–195.
- [11] Y. C. Hon and R. Schaback, *On unsymmetric collocation by radial basis functions*, Appl. Math. Comput. **119** (2001), 177–186.
- [12] M. Jankowska, A. Karageorghis and C. S. Chen, *Kansa RBF method for nonlinear problems*, Int. J. Comp. Meth. and Exp. Meas., to appear.
- [13] 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), 147–161.
- [14] 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.
- [15] A. Karageorghis, C. S. Chen, and X.-Y. Liu, *Kansa-RBF algorithms for elliptic problems in axisymmetric domains*, SIAM J. Sci. Comput., **38** (2016), A435–A470.
- [16] J. A. Kołodziej and J. K. Grabski, *Application of the method of fundamental solutions and the radial basis functions for viscous laminar flow in wavy channel*, Eng. Anal. Bound. Elem. **57** (2015), 58–65.
- [17] J. A. Kołodziej, M. A. Jankowska and M. Mierzwiak, *Meshless methods for the inverse problem related to the determination of elastoplastic properties from the torsional experiment*, Int. J. Solids Struct. **50** (2013), 4217–4225.
- [18] 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.
- [19] M. Li, C. S. Chen, and A. Karageorghis, *The MFS for the solution of harmonic boundary value problems with non-harmonic boundary conditions*, Comput. Math. Appl. **66** (2013), 2400–2424.
- [20] X.-Y. Liu, A. Karageorghis, and C. S. Chen, *A Kansa-radial basis function method for elliptic boundary value problems in annular domains*, J. Sci. Comput. **65** (2015), 1240–1269.
- [21] The MathWorks, Inc., 3 Apple Hill Dr., Natick, MA, *Matlab*.
- [22] A. Mendelson, *Plasticity: Theory and Application*, McMillan Company, New York, 1968.
- [23] J. Nocedal and S. J. Wright, *Numerical Optimization*, Second Edition, Springer Series in Operations Research, Springer Verlag, New York, 2006.
- [24] 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.
- [25] I. J. Sobey, *On flow through furrowed channels. Part 1. Calculated flow patterns*, J. Fluid Mech. **96** (1980), 1–26.
- [26] C.-C. Tsai, *Homotopy method of fundamental solutions for solving certain nonlinear partial differential equations*, Eng. Anal. Bound. Elem. **36** (2012), 1226–1234.
- [27] C.-C. Tsai, C.-S. Liu and W.-C. Yeih, *Fictitious time integration method of fundamental solutions with Chebyshev polynomials for solving Poisson-type nonlinear PDEs*, CMES Comput. Model. Eng. Sci. **56** (2010), 131–151.
- [28] C. C. Wang and C. K. Chen, *Forced convection in a wavy-wall channel*, Int. J. Heat Mass Transf. **45** (2002), 2587–2595.
- [29] Y. L. Wu and C. Shu, *Development of RBF-DQ method for derivative approximation and its application to simulate natural convection in concentric annuli*, Comput. Mech. **29** (2002), 477–485.
- [30] G. Yao, C. S. Chen and H. Zheng, *A modified method of approximate particular solutions for solving linear and nonlinear PDEs*, Numer. Methods Partial Differential Equations **33** (2017), 1839–1858.

INSTITUTE OF APPLIED MECHANICS, FACULTY OF MECHANICAL ENGINEERING AND MANAGEMENT, POZNAN
UNIVERSITY OF TECHNOLOGY, JANA PAWLA II 24, 60-965 POZNAN, POLAND

E-mail address: malgorzata.jankowska@put.poznan.pl

DEPARTMENT OF MATHEMATICS AND STATISTICS, UNIVERSITY OF CYPRUS/ ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ,
P.O.Box 20537, 1678 NICOSIA/ΛΕΥΚΩΣΙΑ, CYPRUS/ΚΥΠΡΟΣ

E-mail address: andreak@ucy.ac.cy

INTERNATIONAL CENTER FOR SIMULATION SOFTWARE IN ENGINEERING AND SCIENCES, COLLEGE OF ME-
CHANICS AND MATERIALS, HOHAI UNIVERSITY, NANJING 211100, CHINA & DEPARTMENT OF MATHEMATICS,
UNIVERSITY OF SOUTHERN MISSISSIPPI, HATTIESBURG, MS 39402, USA

E-mail address: cschen.math@gmail.com