An adaptive LOOCV-based refinement scheme for RBF collocation methods over irregular domains

An adaptive LOOCV-based refinement scheme for RBF collocation methods over irregular domains

Roberto Cavoretto[a,b], Alessandra De Rossi[a,b]

[a] *Department of Mathematics "Giuseppe Peano", University of Torino, via Carlo Alberto 10, 10123 Torino, Italy*
[b] *Member of the INdAM Research group GNCS*

Abstract

In this paper we enhance the adaptive scheme presented in [2] for solving elliptic boundary value problems via RBF collocation methods. More precisely, this study concerns a leave one out cross validation technique applied as an error estimate and used in the adaptive refinement process. The modified algorithm we propose here allows us to get numerical convergence also when L-shape or irregular domains are considered. Moreover, a comparison between unsymmetric and symmetric RBF collocation schemes is performed.

Keywords: meshless approximation, adaptive algorithms, refinement schemes, collocation methods, partial differential equations
2010 MSC: 65D15, 65M70

1. Introduction

Radial basis function (RBF) methods have been receiving great attention and recognition from scientists and engineers for their power and efficacy in solving interpolation or approximation problems, also in high dimensions, and partial differential equations (PDEs). In particular, over the last years RBF collocation methods have constantly being studied and applied to solve a large number of science and engineering problems (see e.g. [1, 3, 11, 20]).

In this paper we propose an adaptive refinement scheme for solving elliptic PDE problems via RBF collocation methods over 2D irregular domains. The adaptive algorithm is characterized by the use of a leave one out cross validation (LOOCV) technique, which is here applied as an error indicator to be used in combination with a new strategy of adaptive refinement. The entire process is tested by using both unsymmetric Kansa's method [12] and the symmetric Hermite-based approach [4] in order to solve some benchmark Poisson and modified Helmholtz equations over L-shape or irregular domains. Note that the above-mentioned LOOCV technique refers to the algorithm – or more precisely to the single formula – originally proposed in [17], and later used not as an error indicator but to choose an optimal value of the RBF shape parameter (see e.g. [5]). This article extends our previous work in [2], enabling the current algorithm to achieve numerical convergence also when irregular domains or other types of PDE problems are considered. In doing so, it came naturally to make also a comparison between the unsymmetric and symmetric collocation methods that are used within our adaptive scheme.

The paper is organized as follows. In Section 2 we recall unsymmetric and symmetric collocation methods for solving elliptic PDEs. In Section 3 we present the adaptive LOOCV-based refinement algorithm, also describing our error indicator and a refinement strategy. In Section 4 we show numerical results in order to illustrate the performance of our adaptive scheme. Section 5 contains conclusions.

Email addresses: roberto.cavoretto@unito.it (Roberto Cavoretto), alessandra.derossi@unito.it (Alessandra De Rossi)

2. RBF collocation methods

Given a domain $\Omega \subset \mathbb{R}^d$ and a linear elliptic partial differential operator $\mathcal{L}$, we define an elliptic PDE problem with Dirichlet boundary conditions (BCs) of the form

$$\begin{aligned} \mathcal{L}u(\boldsymbol{x}) &= f(\boldsymbol{x}), \qquad && \boldsymbol{x} \in \Omega, \\ u(\boldsymbol{x}) &= g(\boldsymbol{x}), \qquad && \boldsymbol{x} \in \partial\Omega. \end{aligned} \tag{1}$$

To solve the problem (1) through RBF collocation methods, we make use of both Kansa's unsymmetric method [12] and Hermite-based symmetric approach [4].

2.1. Kansa's unsymmetric collocation method

For Kansa's approach we express the approximate solution $\hat{u}$ as a linear combination of basis functions as usually occurs in RBF interpolation [5], i.e.

$$\hat{u}(\boldsymbol{x}) = \sum_{j=1}^{N} c_j \phi_\varepsilon(||\boldsymbol{x} - \boldsymbol{z}_j||_2), \tag{2}$$

where c_j denotes unknown real coefficients, $||\cdot||_2$ is the Euclidean norm, and $\phi_\varepsilon : [0,\infty) \to \mathbb{R}$ is some RBF depending on a *shape parameter* $\varepsilon > 0$ such that

$$\phi_\varepsilon(||\boldsymbol{x} - \boldsymbol{z}||_2) = \phi(\varepsilon||\boldsymbol{x} - \boldsymbol{z}||_2), \qquad \forall\, \boldsymbol{x}, \boldsymbol{z} \in \Omega.$$

As an example, globally supported RBFs that are commonly used for solving PDEs are listed below along with their smoothness degrees (see e.g. [5, 19]):

$$\begin{aligned} &\phi_\varepsilon(r) = (1 + \varepsilon^2 r^2)^{-1/2}, && \text{Inverse MultiQuadric } C^\infty \text{ (IMQ)}, \\ &\phi_\varepsilon(r) = \exp(-\varepsilon r)(\varepsilon^3 r^3 + 6\varepsilon^2 r^2 + 15\varepsilon r + 15), && \text{Matérn } C^6 \text{ (M6)}. \end{aligned}$$

In (2) we distinguish between a set $Z_N = \{\boldsymbol{z}_1, \dots, \boldsymbol{z}_N\}$ of *centers* and a set $X_N = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\} \subset \Omega$ of *collocation points*. Even if formally different, they can also coincide. So, in the following, we will assume the scenario with $Z_N = X_N$. Moreover, for our purposes the set X_N is split into the sets X_{N_I} of interior points and X_{N_B} of boundary points, so that $X_N = X_{N_I} \cup X_{N_B}$, with the subscript N_I and N_B denoting the number of interior and boundary points, respectively.

When matching the PDE and the BCs in (1) at the set X_N, we get the collocation system of linear equations

$$A\boldsymbol{c} = \boldsymbol{v}, \tag{3}$$

where A is the collocation matrix defined as

$$A = \begin{bmatrix} \hat{A}_\mathcal{L} \\ \hat{A} \end{bmatrix}, \tag{4}$$

whose two blocks in (4) are given by

$$\begin{aligned} (\hat{A}_\mathcal{L})_{ij} &= \mathcal{L}\phi_\varepsilon(||\boldsymbol{x}_i - \boldsymbol{z}_j||_2), \qquad && \boldsymbol{x}_i \in X_{N_I},\ \boldsymbol{z}_j \in Z_N, \\ \hat{A}_{ij} &= \phi_\varepsilon(||\boldsymbol{x}_i - \boldsymbol{z}_j||_2), \qquad && \boldsymbol{x}_i \in X_{N_B},\ \boldsymbol{z}_j \in Z_N, \end{aligned}$$

and $\boldsymbol{v}$ is the vector of entries

$$v_i = \begin{cases} f(\boldsymbol{x}_i), & \boldsymbol{x}_i \in X_{N_I}, \\ g(\boldsymbol{x}_i), & \boldsymbol{x}_i \in X_{N_B}. \end{cases} \tag{5}$$

2.2. Hermite-based symmetric collocation method

The symmetric collocation method is based on the generalized Hermite interpolation method (see e.g. [5]). The approximate solution $\hat{u}$ is expressed in the following form

$$\hat{u}(\boldsymbol{x}) = \sum_{j=1}^{N_I} c_j \mathcal{L}^{\boldsymbol{z}} \phi_\varepsilon(||\boldsymbol{x} - \boldsymbol{z}_j||_2) + \sum_{j=N_I+1}^{N} c_j \phi_\varepsilon(||\boldsymbol{x} - \boldsymbol{z}_j||_2), \tag{6}$$

where $\mathcal{L}^{\boldsymbol{z}}$ is the differential operator used in (1), but acting on ϕ_ε viewed as a function of the second argument, i.e. $\mathcal{L}\phi_\varepsilon$ is equal to $\mathcal{L}^{\boldsymbol{z}}\phi_\varepsilon$ up to a possible difference in sign. Notice that in the second sum on the right the notation that involves an index $j = N_I + 1, \ldots, N$ refers to the N_B boundary indexes.

After enforcing the collocation conditions at the set X_N, we obtain a linear system of the form (3), where the vector $\boldsymbol{v}$ is defined as in (5) and the collocation matrix A is given by

$$A = \begin{bmatrix} \hat{A}_{\mathcal{L}\mathcal{L}^{\boldsymbol{z}}} & \hat{A}_{\mathcal{L}} \\ \hat{A}_{\mathcal{L}^{\boldsymbol{z}}} & \hat{A} \end{bmatrix}. \tag{7}$$

The four blocks in (7) are generated as follows:

$$\begin{aligned}
(\hat{A}_{\mathcal{L}\mathcal{L}^{\boldsymbol{z}}})_{ij} &= \mathcal{L}\phi_\varepsilon(||\boldsymbol{x}_i - \boldsymbol{z}_j||_2), && \boldsymbol{x}_i \in X_{N_I}, \ \boldsymbol{z}_j \in Z_{N_I}, \\
(\hat{A}_{\mathcal{L}})_{ij} &= \mathcal{L}\phi_\varepsilon(||\boldsymbol{x}_i - \boldsymbol{z}_j||_2), && \boldsymbol{x}_i \in X_{N_I}, \ \boldsymbol{z}_j \in Z_{N_B}, \\
(\hat{A}_{\mathcal{L}^{\boldsymbol{z}}})_{ij} &= \mathcal{L}^{\boldsymbol{z}}\phi_\varepsilon(||\boldsymbol{x}_i - \boldsymbol{z}_j||_2), && \boldsymbol{x}_i \in X_{N_B}, \ \boldsymbol{z}_j \in Z_{N_I}, \\
\hat{A}_{ij} &= \phi_\varepsilon(||\boldsymbol{x}_i - \boldsymbol{z}_j||_2), && \boldsymbol{x}_i \in X_{N_B}, \ \boldsymbol{z}_j \in Z_{N_B}.
\end{aligned}$$

Thus, if the RBF centers coincide with collocation points, i.e. $Z_N = X_N$, the matrix (7) is symmetric and the collocation method is certainly well-posed. Further, we have the following convergence result [5, 19]:

Theorem 2.1. *Let $\Omega \subseteq \mathbb{R}^d$ be a polygonal and open region. Let $\mathcal{L} \neq 0$ be a second-order linear elliptic differential operator with coefficients in $C^{2(k-2)}(\bar{\Omega})$ that either vanish on $\bar{\Omega}$ or have no zero there. Suppose that $\Phi \in C^{2k}(\mathbb{R}^d)$ is a strictly positive definite function. Suppose further that the boundary value problem* (1) *has a unique solution $u \in \mathcal{N}_\Phi(\Omega)$ for given $f \in C(\Omega)$ and $g \in C(\partial\Omega)$. Let $\hat{u}$ be the approximate collocation solution of the form* (6) *based on $\Phi = \phi(||\cdot||)$. Then*

$$||u - \hat{u}||_{L_\infty(\Omega)} \leq C h^{k-2} ||u||_{\mathcal{N}_\Phi(\Omega)}$$

for all sufficiently small h, where $h = \sup_{\boldsymbol{x}\in\Omega} \min_{\boldsymbol{x}_i \in X_N} ||\boldsymbol{x} - \boldsymbol{x}_i||_2$ denotes the fill distance, namely the larger of the distances in the interior and on the boundary of Ω, respectively.

2.3. Comparison between unsymmetric and symmetric collocation

An issue due to the unsymmetric method is that the matrix (4) may be singular for certain rare configurations of the centers. In fact, Kansa's method might not be well-posed for arbitrary center locations [10]. However, the huge popularity of this collocation method associated with a theoretical/convergence analysis has encouraged the use of Kansa's approach despite its potential for failure (see [5, 6]). It is, indeed, possible to find sufficient conditions on the centers as in [15] so that invertibility of the matrix (4) is ensured.

On the other hand, the symmetric method guarantees invertibility of the collocation matrix (7), since it depends on a different use of the basis functions deriving from the Hermite interpolation approach. This choice results in a convergence result (cf. Theorem 2.1) and related analysis given in [8, 9].

Numerical comparisons between unsymmetric and symmetric methods have been discussed in [5, 13]. However, while the work [13] states that the symmetric approach is more accurate than the unsymmetric one, in [5] a little superiority of the unsymmetric method is shown. On the basis of these opposing results we have thus decided to analyze numerically the behavior of both collocation methods when applying in an adaptive scheme (see Section 4).

3. Adaptive LOOCV-based scheme

3.1. Error computation

The idea of LOOCV for RBF collocation methods can be depicted as follows. At first, the data are split into two distinct sets: a *training set* $\{v(\boldsymbol{x}_1),\ldots, v(\boldsymbol{x}_{k-1}), v(\boldsymbol{x}_{k+1}),\ldots, v(\boldsymbol{x}_N)\}$, and a *validation set* that is merely made of the single value $v(\boldsymbol{x}_k)$, i.e. the one left out when generating the training set [5].

Thus, given an index $k \in \{1,\ldots,N\}$ and a fixed value of ε, we consider a partial RBF approximant denoted by $\hat{u}^{[k]}$ and obtained from (2) (or (6)) by excluding the point $\boldsymbol{x}_k$. The related coefficients $c_j^{[k]}$ are determined by collocating the training data, i.e.

$$\begin{array}{rlr} \mathcal{L}\hat{u}^{[k]}(\boldsymbol{x}_i) = v(\boldsymbol{x}_i), & \quad \boldsymbol{x}_i \in X_{N_I}, & \\ & & \text{for } i = 1,\ldots,k-1,k+1,\ldots,N. \\ \hat{u}^{[k]}(\boldsymbol{x}_i) = v(\boldsymbol{x}_i), & \quad \boldsymbol{x}_i \in X_{N_B}, & \end{array}$$

To measure the quality of this attempt, we compute the absolute error

$$e_k = |v(\boldsymbol{x}_k) - \hat{u}^{[k]}(\boldsymbol{x}_k)|, \tag{8}$$

at the one validation point $\boldsymbol{x}_k$ not used to find the approximant $\hat{u}^{[k]}$. Now, evaluating the error in (8), for all $k = 1,\ldots,N$, we get a vector $\boldsymbol{e} = (e_1,\ldots,e_N)^T$, which can be used as an error indicator to identify the regions that need to be refined in the neighborhood of the point $\boldsymbol{x}_k$. However, the error (8) can be computed more efficiently without solving N collocation problems, each of size $(N-1)\times(N-1)$. In fact, in [17] the error computation can be simplified to a single formula. The rule (8) can thus be rewritten as

$$e_k = \left|\frac{c_k}{A^{-1}_{kk}}\right|, \qquad k = 1,\ldots,N, \tag{9}$$

where c_k is the k-th coefficient of the full approximate solution (2) and A^{-1}_{kk} is the k-th diagonal element of the inverse of the corresponding $N \times N$ collocation matrix A in (4).

3.2. Computational procedure: error indicator and refinement

`Stage 1`. We define an initial set $X_{N^{(1)}} \equiv X_N = \{\boldsymbol{x}_1^{(1)},\ldots,\boldsymbol{x}_{N^{(1)}}^{(1)}\}$ of collocation points, associated with an initial set $Z_{N^{(1)}} \equiv Z_N = \{\boldsymbol{z}_1^{(1)},\ldots,\boldsymbol{z}_{N^{(1)}}^{(1)}\}$ of centers, where $Z_N = X_N$. It is then split into two subsets, i.e. the set $X_{N_I^{(j)}} = \{\boldsymbol{x}_1^{(j)},\ldots,\boldsymbol{x}_{N_I^{(j)}}^{(j)}\}$ of interior points, and the set $X_{N_B^{(j)}} = \{\boldsymbol{x}_1^{(j)},\ldots,\boldsymbol{x}_{N_B^{(j)}}^{(j)}\}$ of boundary points, where $j = 1,2,\ldots$ denotes the algorithm iteration and the subscript $_{N^{(j)}}$ defines the number of points.

`Stage 2`. Fixed a tolerance $\tau > 0$, from (9) we can iteratively define an error indicator via LOOCV as

$$e_k^{(j)} = \left|\frac{c_k}{A^{-1}_{kk}}\right|, \qquad k = 1,\ldots,N^{(j)}, \qquad j = 1,2,\ldots. \tag{10}$$

`Stage 3`. If the error indicator (10) is such that $e_k^{(j)} > \tau$, then a refinement is applied in the neighborhood of $\boldsymbol{x}_k$. In doing so, we first compute the so-called *separation distance*

$$q_{X_{N^{(j)}}} = \frac{1}{2}\min_{u\neq v}||\boldsymbol{x}_u^{(j)} - \boldsymbol{x}_v^{(j)}||_2, \qquad \boldsymbol{x}_u^{(j)}, \boldsymbol{x}_v^{(j)} \in X_{N^{(j)}}, \qquad j = 1,2,\ldots, \tag{11}$$

and then sum up or subtract the quantity in (11) to one or both coordinates of the point $\boldsymbol{x}_k = (\boldsymbol{x}_{k,1}, \boldsymbol{x}_{k,2})$. More precisely, the refinement technique used here consists in the addition of the following eight points around $\boldsymbol{x}_k$, whose coordinates are $(\boldsymbol{x}_{k,1}, \boldsymbol{x}_{k,2} + q_{X_{N^{(j)}}})$, $(\boldsymbol{x}_{k,1} - q_{X_{N^{(j)}}}, \boldsymbol{x}_{k,2} + q_{X_{N^{(j)}}})$, $(\boldsymbol{x}_{k,1} - q_{X_{N^{(j)}}}, \boldsymbol{x}_{k,2})$, $(\boldsymbol{x}_{k,1} - q_{X_{N^{(j)}}}, \boldsymbol{x}_{k,2} - q_{X_{N^{(j)}}})$, $(\boldsymbol{x}_{k,1}, \boldsymbol{x}_{k,2} - q_{X_{N^{(j)}}})$, $(\boldsymbol{x}_{k,1} + q_{X_{N^{(j)}}}, \boldsymbol{x}_{k,2} - q_{X_{N^{(j)}}})$, $(\boldsymbol{x}_{k,1} + q_{X_{N^{(j)}}}, \boldsymbol{x}_{k,2})$, and

$(\boldsymbol{x}_{k,1}+q_{X_{N^{(j)}}}, \boldsymbol{x}_{k,2}+q_{X_{N^{(j)}}})$. The choice of using an eight-point approach has been done because it enables us to cover uniformly the domain areas that need to be refined without adding too few or too many points, also when an irregular domain is considered.

`Stage 4`. When all components of error terms in (10) are less than or equal to τ, the refinement procedure stops and the final set of collocation points is returned.

3.3. Computational cost and complexity analysis

The computational cost of our algorithm depends primarily on the iterated use of the LOOCV-based error indicator in (10). This requires that the collocation matrix of the form (4) (or (7)) is inverted at each iteration. As a result, the computational cost is $\mathcal{O}((N^{(j)})^3)$, where $j = 1, 2 \dots$ denotes the algorithm iteration. Once the refinement process was completed, the algorithm stops and the approximate solution (2) (or (6)) is computed on a set of evaluation points. This phase needs a matrix-vector multiplication with a vector that has the same size of the final set of centers.

4. Numerical results

In this section we consider two examples to validate our adaptive refinement scheme by solving Poisson and modified Helmholtz problems on L-shape and irregular domains. The codes are implemented in Matlab and run on a laptop with an Intel(R) Core(TM) i7-6500U CPU 2.50 GHz processor with 8GB RAM.

In order to assess the accuracy of the adaptive method, we compute the *root mean square error* (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{M}\sum_{i=1}^{M} |u(\boldsymbol{\xi}_i) - \hat{u}(\boldsymbol{\xi}_i)|^2},$$

where $\boldsymbol{\xi}_i$, $i = 1, \dots, M$, are gridded evaluation points and $M = 51 \times 51$. Moreover, to analyze the stability of the method, we evaluate the *condition number* (CN) of the collocation matrix A in (4) (or (7)) by using the Matlab command `cond`. Finally, as a measure of computational efficiency we report the CPU time expressed in seconds.

Example 1. Consider the Laplace operator $\mathcal{L} = -\Delta$ in (1), whose analytical solution is given by

$$u_1(x_1, x_2) = \frac{1}{25(4x_1-2)^2 + 25(4x_2-2)^2 + 1},$$

defined on a L-shape domain Ω contained in $[0, 1]^2$ (see [14]). In Table 1 we show the algorithm performance obtained via M6 as RBF for various values of ε, also including the total number N_{fin} of collocation points required to achieve the final results. Specifically, we remark that the choice of the shape parameters given in Table 1 turns out to be a good trade-off to get both accuracy and stability in the numerical method. Nevertheless, in this example the use of different values of ε has been done primarily to highlight as the choice of "good" shape parameters does not produce significant changes in the results. In Table 2 we then compare the numerical results by applying Kansa and Hermite-based methods with IMQ. These tests (and others omitted for shortness) reveal greater performances of Kansa's approach. Moreover, as an example, in Figure 2 we depict a couple of graphs with some final configurations of collocation points. Note that the adaptive scheme is able to detect and increase the points in the area where u_1 has a pick (see Figure 1, left).

Finally, in order to demonstrate the usefulness of our adaptive scheme, in Table 3 we report the results obtained by applying the non-adaptive Kansa and Hermite-based methods on uniform/gridded point sets. From a comparison with Table 2, we can observe that the non-adaptive schemes can achieve a similar level of accuracy only when using a quite larger number of collocation points, namely about 1300 points for Kansa's approach and 1900 points for Hermite's one. However, in this specific case our study also points out as CNs for Kansa and times for Hermite are lower than the ones obtained via our adaptive scheme whereas CNs for Hermite and times for Kansa are comparable.

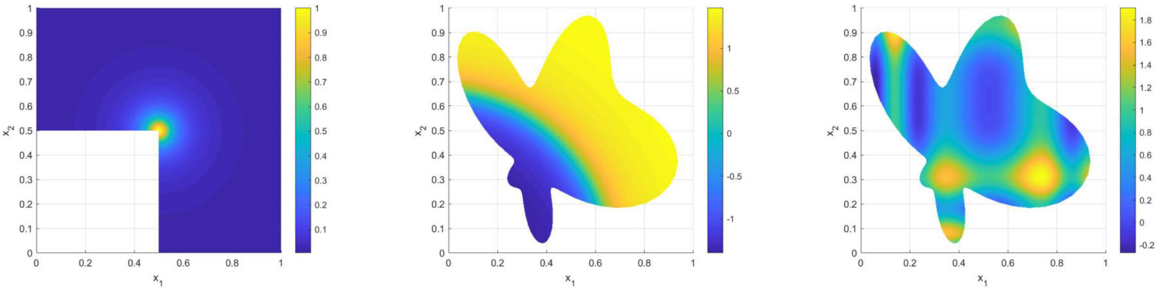

Figure 1: Exact solutions viewed from the top with colorbar: u_1 (left), u_2 (center) and u_3 (right).

ε	N_{fin}	RMSE	CN	time
5	722	1.73e−4	1.73e+13	0.8
6	737	1.04e−4	1.26e+14	1.1
7	737	1.05e−4	5.13e+13	1.1

Table 1: Results obtained for Example 1 via Kansa's method with M6 by varying ε for $\tau = 10$. The initial number of discretization points is $N = 469$, with $N_I = 385$ (gridded) and $N_B = 84$.

Method	N_{fin}	RMSE	CN	time
Kansa	774	9.30e−5	4.38e+11	0.7
Hermite	1614	1.34e−4	3.26e+10	3.9

Table 2: Results obtained for Example 1 with IMQ for $\varepsilon = 12$ and $\tau = 10$. The initial number of discretization points is $N = 469$, with $N_I = 385$ (gridded) and $N_B = 84$.

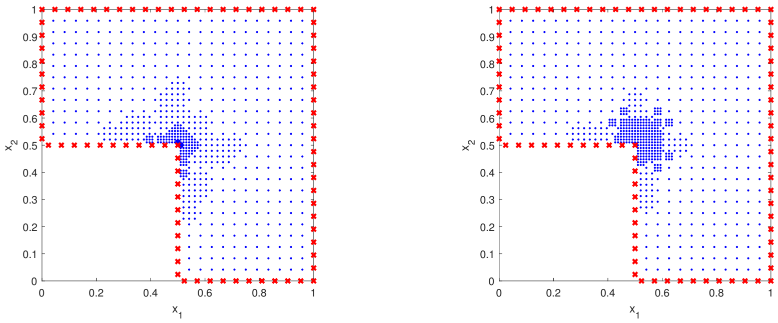

Figure 2: Final discretization points via Kansa's method for u_1 computed by starting with gridded points as in Tables 1–2. These tests have been carried out by using M6 with $\varepsilon = 6$ (left) and IMQ with $\varepsilon = 12$ (right).

Example 2. Consider the modified Helmholtz operator $\mathcal{L} = (-\Delta + \nu^2 \mathcal{I})$ in (1), where $\mathcal{I}$ is the identity

Method	N	RMSE	CN	time
Kansa	720	1.03e−3	7.26e+5	0.5
	1261	1.84e−4	1.91e+7	0.7
	1344	7.66e−5	9.39e+7	0.8
Hermite	1612	1.84e−4	8.09e+10	1.5
	1952	1.32e−4	1.04e+11	1.9
	2268	9.69e−5	2.95e+11	2.4

Table 3: Results obtained for Example 1 by applying non-adaptive methods computed on gridded points with IMQ for $\varepsilon = 12$ and $\tau = 10$.

operator. The exact solutions of such problems (cf. [14, 16]) are

$$u_2(x_1, x_2) = \arctan(20(\sqrt{(x_1+0.05)^2+(x_2+0.05)^2} - 0.7)),$$
$$u_3(x_1, x_2) = \sin(2(4x_1-2.1)^2)\cos((4x_1-2.3)^2) + \sin^2((4x_2-2.5)^2),$$

defined on an irregular domain such as the Ameoba like shape domain $\Omega \subset [0,1]^2$ and bounded by the parametric curve [18]

$$r(\theta) = e^{\sin(\theta)}\sin^2(2\theta) + e^{\cos(\theta)}\cos^2(2\theta).$$

In this example the numerical results are illustrated in Table 4 while the corresponding plots are shown in Figure 3. From both graphs we can see as the method finds the steep variation of u_2 (see Figure 1, center) and is sensitive to the several oscillations of u_3 (see Figure 1, right).

Problem	τ	N_{fin}	RMSE	CN	time
u_2	5	1209	2.39e−4	3.14e+14	3.6
u_3	15	1132	5.75e−4	1.76e+15	3.2

Table 4: Results obtained for Example 2 via Kansa's method with M6 for $\nu = 3$ and $\varepsilon = 10$. The initial number of discretization points is $N = 361$, with $N_I = 261$ (gridded) and $N_B = 100$.

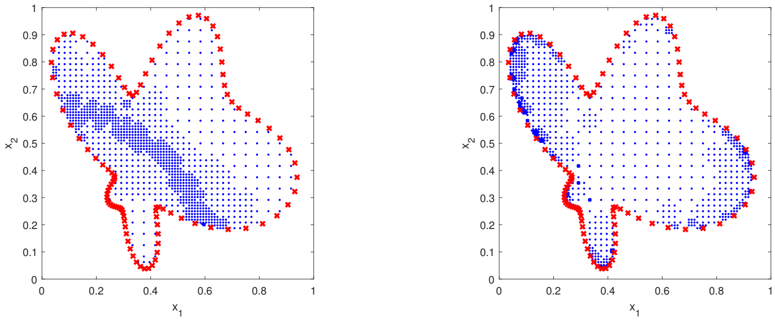

Figure 3: Final discretization points for u_2 (left) and u_3 (right) of Table 4.

So far, in our previous tests, we have assumed that $Z_N = X_N$, i.e. the interior and boundary centers coincide with the corresponding collocation points. Now, here, we consider a case in which $Z_{N_I} = X_{N_I}$ while the set X_{N_B} of collocation points is on the boundary of the domain and the set Z_{N_B} of centers is taken outside the domain as suggested in [7]. In Table 5 we show the results obtained with this different scenario whereas in Figure 4 (left to right) we report the graphical representation of the final collocation points for u_2 and u_3, respectively.

Problem	τ	N_{fin}	RMSE	CN	time
u_2	5	1299	2.56e−4	4.20e+14	3.7
u_3	15	1156	4.92e−4	3.50e+14	3.2

Table 5: Results obtained for Example 2 via Kansa's method with M6 for $\nu = 3$ and $\varepsilon = 10$. The initial number of discretization points is $N = 361$, with $N_I = 261$ (gridded) and $N_B = 100$. Here we assume $Z_{N_I} = X_{N_I}$ and $Z_{N_B} \neq X_{N_B}$.

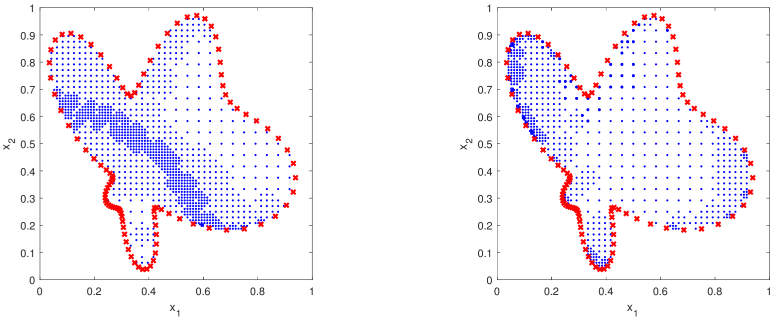

Figure 4: Final discretization points for u_2 (left) and u_3 (right) of Table 5.

In conclusion, though in this work we only considered globally supported RBFs, we can remark that the use of compactly supported RBFs is also possible. However, since the most accurate results were obtained with quite large supports, the use of compactly supported functions does not provide particular benefits w.r.t. globally supported ones. For this reason and for the sake of brevity, we do not consider this case in the present paper.

Remark 4.1. *In our experiments the iterative algorithm starts by considering uniform (or gridded) points. However, the use of different point distributions can influence even significantly the approximation results. In fact, from further tests (not reported in the paper) we observed in general a loss of accuracy and an increase of condition numbers in the numerical method when using non-uniform points such as random or quasi-random points.*

5. Conclusions

In this work we presented a variation of the adaptive scheme discussed in [2], which showed some issues in terms of algorithm convergence when one considered irregular (e.g. L-shape) domains. This target was achieved by using a LOOCV-based error indicator combined with an eight-point refinement procedure. The resulting algorithm was tested to solve some benchmark Poisson-type and modified Helmholtz problems. This allowed us to also make a comparison between unsymmetric and symmetric collocation, analyzing numerically the behavior of both collocation methods within our adaptive scheme. From this study we observed a greater sensitivity of the unsymmetric method, when the adaptive strategy is applied, and accordingly better performance in terms of accuracy and algorithm convergence than the symmetric approach.

Acknowledgments

The authors acknowledge support from the Department of Mathematics “Giuseppe Peano” of the University of Torino via Project 2018 “Mathematics for applications”. This work was partially supported by INdAM–GNCS Project 2019 “Kernel-based approximation, multiresolution and subdivision methods and related applications”. This research has been accomplished within RITA (Research ITalian network on Approximation).

References

[1] R. Cavoretto, A. De Rossi, Adaptive meshless refinement schemes for RBF-PUM collocation, Appl. Math. Lett. 90 (2019) 131–138.

[2] R. Cavoretto, A. De Rossi, A two-stage adaptive scheme based on RBF collocation for solving elliptic PDEs, submitted (2019).

[3] C.S. Chen, A. Karageorghis, F. Dou, A novel RBF collocation method using fictitious centres, Appl. Math. Lett. 101 (2020) 106069.

[4] G.E. Fasshauer, Solving partial differential equations by collocation with radial basis functions, in: Surface Fitting and Multiresolution Methods A. Le Méhauté, C. Rabut, L.L. Schumaker, eds., Vanderbilt Univ. Press, Nashville, TN, 1997, pp. 131–138.

[5] G.E. Fasshauer, Meshfree Approximation Methods with Matlab, Interdisciplinary Mathematical Sciences, vol. 6, World Scientific Publishing Co., Singapore, 2007.

[6] G.E. Fasshauer, M.J. McCourt, Kernel-based Approximation Methods using Matlab, Interdisciplinary Mathematical Sciences, Vol. 19, World Scientific Publishing Co., Singapore, 2015.

[7] A.I. Fedoseyev, M.J. Friedman, E.J. Kansa, Improved multiquadric method for elliptic partial differential equations via PDE collocation on the boundary, Comput. Math. Appl. 43 (2002) 439–455.

[8] C. Franke, R. Schaback, Convergence order estimates of meshless collocation methods using radial basis functions, Adv. Comput. Math. 8 (1998) 381–399.

[9] C. Franke, R. Schaback, Solving partial differential equations by collocation using radial basis functions, Appl. Math. Comput. 93(1998), 73–82.

[10] Y.C. Hon, R. Schaback, On unsymmetric collocation by radial basis functions, Appl. Math. Comput. 119 (2001) 177–186.

[11] S. Kaennakham, N. Chuathong, An automatic node-adaptive scheme applied with a RBF-collocation meshless method, Appl. Math. Comput. 348 (2019) 102–125.

[12] 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.

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

[14] E. Larsson, V. Shcherbakov, A. Heryudono, A least squares radial basis function partition of unity method for solving PDEs, SIAM J. Sci. Comput. 39 (2017) A2538–A2563.

[15] L. Ling, R. Opfer, R. Schaback, Results on meshless collocation techniques, Eng. Anal. Bound. Elem. 30 (2006) 247–253.

[16] W. F. Mitchell, A collection of 2D elliptic problems for testing adaptive grid refinement algorithms, Appl. Math. Comput. 220 (2013) 350–364.

[17] 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.

[18] M. Uddin, H. Ali, M. Taufiq, On the approximation of a nonlinear biological population model using localized radial basis function method, Math. Comput. Appl. (2019) 24, 54.

[19] H. Wendland, Scattered Data Approximation, Cambridge Monogr. Appl. Comput. Math., vol. 17, Cambridge Univ. Press, Cambridge, 2005.

[20] Y. Zhang, An accurate and stable RBF method for solving partial differential equations, Appl. Math. Lett. 97 (2019) 93–98.