Neural, Parallel, and Scientific Computations 23 (2015) 411-420

COMPARING TWO TYPES OF BASES FOR SOLVING ELLIPTIC BOUNDARY VALUE PROBLEMS

HAIYAN TIAN, HONGMEI CHI, AND YANZHAO CAO

Department of Mathematics, The University of Southern Mississippi,
Hattiesburg, Mississippi 39406 USA
Computer and Information Sciences, Florida A&M University, Tallahassee,
Florida 32307 USA
Department of Mathematics & Statistics, Auburn University
Auburn, Alabama 36830 USA

ABSTRACT. Delta-shaped basis functions are approximately compactly supported basis functions and they can effectively handle scattered data. In this paper, the performance of delta-shaped basis functions is compared with that of the compactly supported radial basis functions for solving elliptic boundary value problems in a collocation approach.

AMS (MOS) Subject Classification. 65N35.

1. INTRODUCTION

The traditional methods for partial differential equations (PDEs), such as finite difference and finite element methods, are mesh-based methods since they require domain discretization with a mesh. The mesh generation and refinement can be expensive, especially for domains of higher dimension and irregular shape. To overcome these difficulties different basis functions have been used for solving PDEs due to their meshfree feature and their flexibility in handling scattered data. The most commonly used basis is the radial basis functions (RBFs), the applications of which can be found in various areas in science and engineering during the past three decades.

Let R^d denote a space of dimension d. A few examples of RBFs are the multiquadrics $\varphi(r) = (r^2 + c^2)^{1/2}$, inverse multiquadrics $\varphi(r) = (r^2 + c^2)^{-1/2}$, thin plate splines $\varphi(r) = r^2 \log(r)$, and Gaussians $\varphi(r) = e^{-cr^2}$. Here $r = \|x\|$ for $x \in R^d$, $\|\cdot\|$ is the Euclidean norm, and $c > 0$ is a selected constant. These RBFs are globally supported and referred to as classical RBFs [5]. They have found many applications [10, 11, 17]. Despite the meshless feature and other advantages such as that Gaussians and multiquadrics possess spectral convergence properties, one major drawback in all globally supported RBFs (GS-RBFs) is that the GS-RBFs often generate dense

Received September 15, 2015

and highly ill-conditioned matrices for a complex problem with a large set of scattered data. For the computational advantage of a matrix that is sparse and less ill-conditioned, the compactly supported RBFs (CS-RBFs) [20, 21, 22] have been used for various problems [1, 2, 7, 13].

Although the Delta-shaped basis functions (DBFs) are approximately compactly supported, they have shown excellent performances [15, 16, 18, 19]. In this article, we would like to demontrate the effectiveness of the DBFs by comparing the results of DBFs and CS-RBFs in solving elliptic boundary value problems in the context of Kansa's method [8, 9].

We consider the elliptic boundary value problem,

$$L\,[u] = f(x),\ \ x \in \Omega, \tag{1.1}$$

$$u = g\,(x)\,,\ \ x \in \partial\Omega, \tag{1.2}$$

where $\Omega \subset R^2$ is a simply connected domain bounded by a simple closed curve $\partial\Omega$, L is a linear elliptic differential operator of the form,

$$L = \sum_{k_1,k_2=0}^{q} A_{k_1,k_2} \frac{\partial^{k_1+k_2}}{\partial x_1^{k_1} \partial x_2^{k_2}},\ \ A_{k_1,k_2} \text{ are constants,}$$

and f and g are continuous functions.

In [1, 2], the CS-RBFs are used under the dual reciprocity method (DRM) [12, 14] to handle the inhomogeneous term f. In their approach, a particular solution of (1.1) is represented by the particular solutions corresponding to the CS-RBFs. In this paper, the CS-RBFs are used in the context of Kansa's method for the problem (1.1)–(1.2). Instead of approximating the source term, we approximate the solution of (1.1)–(1.2) directly. The organization of the paper is as follows: In Section 2, we describe the Wendland CS-RBFs and the approximation of the solution of (1.1)–(1.2). In Section 3, we describe the characteristics of DBFs. In Section 4, numerical results by CS-RBFs and DBFs are presented. Better results are observed from using DBFs. Concluding remarks are given in Section 5.

2. THE CS-RBF APPROXIMATE SOLUTION

In [20], a class of CS-RBFs are constructed using the operator I and the univariate function ϕ_l which are defined respectively as

$$I\,(f)\,(r) = \int_r^{\infty} t f(t) dt,$$

and

$$\phi_l\,(r) = (1-r)_+^l = \begin{cases} (1-r)^l, \ \text{if } 0 \le r \le 1, \\ 0, \ \text{if } r > 1. \end{cases}.$$

The Wendland CS-RBFs $\varphi_{l,k} = I^k \phi_l$ with $l > 0$ have support in $[0, 1]$. Several optimal Wendland's CS-RBFs for $d = 2, 3$ are listed in Table 1. These functions $\varphi_{l,k}$ represent Wendland CS-RBFs with diferentiability $2k$ for some $l \in N$.

TABLE 1. Wendland's CS-RBFs with various degrees of smoothness

C^0	$\varphi_{2,0}(r) = (1-r)_+^2$
C^2	$\varphi_{3,1}(r) = (1-r)_+^4(4r+1)$
C^4	$\varphi_{4,2}(r) = (1-r)_+^6(35r^2+18r+3)$
C^6	$\varphi_{5,3}(r) = (1-r)_+^8(32r^3+25r^2+8r+1)$

The CS-RBFs can be scaled with a shape parameter α. Since the CS-RBF $\varphi_{l,k}(r)$ has a support of radius 1, the basis function $\varphi_{l,k}\left(\frac{r}{\alpha}\right)$ has a support of radius α. Similar to the trade-off principle between accuracy and ill-conditioning for GS-RBFs, there is a trade-off principle between computational efficiency and convergence for CS-RBFs as noted in [3, 4].

Here we use CS-RBFs in Kansa's collocation approach for the problem (1.1)–(1.2). We choose N_1 collocation points $\left\{x^{(i)}\right\}_{i=1}^{N_1}$ in Ω and N_b collocation points $\left\{x^{(N_1+i)}\right\}_{i=1}^{N_b}$ on $\partial\Omega$. In a one-level approach, the solution of (1.1)–(1.2) is approximated by the basis functions $\varphi_{l,k}$ of the same scaling factor α. In a multi-level approach, the solution is approximated by the basis functions of different scaling factors α_s, $s = 1, \cdots, S$, with $S > 1$. For each α_s, K_s center points $\xi^{(s,\,j)} \in \Omega$, $j = 1, \cdots, K_s$ are chosen. The approximate solution $\widetilde{u}(x)$ of (1.1)–(1.2) is written in the form of

$$\widetilde{u}(x) = \sum_{s=1}^{S}\sum_{j=1}^{K_s} c_{s,j}\varphi_{l,k}\left(\left\|x-\xi^{(s,j)}\right\|/\alpha_s\right). \tag{2.1}$$

When $S = 1$, (2.1) becomes a one-level case. We collocate at the interior and boundary collocation points. With $L_{(x)}$ denoting the operator on $\varphi_{l,k}(\|x-\xi\|/\alpha)$ viewed as a function of x, we obtain the following linear system of $c_{s,j}$,

$$\sum_{s=1}^{S}\sum_{j=1}^{K_s} c_{s,j}L_{(x)}\left[\varphi_{l,k}\left(\left\|x^{(i)}-\xi^{(s,j)}\right\|/\alpha_s\right)\right] = f\left(x^{(i)}\right),\ i=1,\ \cdots,\ N_1,$$

$$\sum_{s=1}^{S}\sum_{j=1}^{K_s} c_{s,j}\left[\varphi_{l,k}\left(\left\|x^{(N_1+i)}-\xi^{(s,j)}\right\|/\alpha_s\right)\right] = g\left(x^{(N_1+i)}\right),\ i=1,\ \cdots,\ N_b,$$

The total number of variables in the above system is $K = \sum_{s=1}^{S} K_s$. We require that the total number of collocation points $N = N_1 + N_b$ be larger than K. The system is solved by least squares method.

3. THE APPROXIMATELY SUPPORTED DELTA-SHAPED BASIS

The delta-shaped basis [15, 16, 19] for 1D is in the form of,

$$I_{M,\chi}(x,\xi) = \sum_{n=1}^{M} c_n(\xi)\,\phi_n(x)\,, \tag{3.1}$$

where $\phi_n(x)$ are solutions of some Sturm-Liouville problem. For example, one choice of $\phi_n(x)$ is

$$\phi_n(x) = \sin\left(n\pi\frac{x+1}{2}\right), \tag{3.2}$$

which are solutions of

$$-\phi''(x) = \lambda\phi, \quad -1 < x < 1,$$
$$\phi(-1) = \phi(1) = 0.$$

In this paper, the coefficients $c_n(\xi)$ are chosen to be

$$c_n(\xi) = \left[1 - \left(\frac{n}{M+1}\right)^2\right]^l \varphi_n(\xi), \tag{3.3}$$

with l being the regularizing parameter. In general, l is coupled with the shape parameter M. The DBFs are approximately compactly supported and are not identically equal to zero on any domain. By ploting $I_{M,l}(x,\xi)$ defined by (3.1), (3.2) and (3.3) for coupled pairs of (M,l), we list the approximate radius of support in Table 2.

TABLE 2. Approximate radius of support of Delta-shaped basis

(M,l)	$(10,4)$	$(20,6)$	$(30,9)$	$(40,12)$	$(50,14)$	$(80,16)$	$(100,18)$
R	0.473	0.319	0.286	0.268	0.243	0.170	0.150

The 2D delta-shaped basis functions are of the form,

$$I_{M,l}(x,y;\xi,\eta) = \sum_{m=1}^{M}\sum_{n=1}^{M} c_n(\xi)\,c_m(\eta)\,\varphi_n(x)\,\varphi_m(y)\,.$$

We note that the 1D basis $I_{M,\chi}(x,\xi)$ in (3.1) vanishes on the boundary of $[-1,1]$ and its 2D basis vanishes on the boundary of $[-1,1]^2$. Hence we ask that the domain of the function to be approximated be imbeded in the interval $[-0.5,0.5]$ for 1D or $\Omega_s = [-0.5,0.5]^2$ for 2D. There is no domain restriction by CS-RBFs. For comparison purpose, we assume the domain Ω satisfies $\overline{\Omega} \subseteq \Omega_s$. If a function is not originally defined on such a domain, proper translation and scaling can make it so.

The DBFs are infinitely differentiable. However, the CS-RBFs $\varphi_{l,k}(r/\alpha)$ possess continuous derivatives up to order $2k$. Thus, any DBFs are applicable to a $2p$-th order differential operator L. When using CS-RBF approach, we should choose CS-RBFs with at least the C^{2p} smoothness.

4. NUMERICAL EXAMPLES

Numerical examples are presented in this section. The mean square root error

$$E = \sqrt{\frac{1}{N_t}\sum_{t=1}^{N_t}\left[\widetilde{u}\left(x_t, y_t\right) - u\left(x_t, y_t\right)\right]^2}, \tag{4.1}$$

is used to measure the solution error. In (4.1), u and $\widetilde{u}$ are repectively the exact and approximate solutions, and N_t is the number of test points that are randomly distributed in the domain. We let $N_t = 200$. For all examples, the interior center and collocation points are randomly distributed in the domain.

Example 4.1. We consider the problem,

$$\Delta u = 1, \text{ for } (x, y) \in D = [0, 1] \times [0, 2],$$
$$u = 0, \text{ for } (x, y) \in \partial D,$$

for which the exact solution is

$$u\left(x, y\right) = \sum_{n,m=1}^{\infty} -\frac{4\left(\cos n\pi - 1\right)\left(\cos m\pi - 1\right)}{nm\pi^4\left[n^2 + m^2/4\right]} \sin\left(n\pi x\right) \sin\left(\frac{m\pi y}{2}\right). \tag{4.2}$$

We use $N_1 = 1317$ interior collocation points produced by the Matlab Poisson solver and $N_b = 300$ collocation points evenly distributed on $\partial\Omega$. The number of center points in Ω and on $\partial\Omega$ are 900 and 150 respectively. The exact solution is computed by using $n = m = 300$ in (4.2). The errors of the numerical solutions in the one-level approach by the basis functions $\varphi_{3,1}$, $\varphi_{4,2}$, and $\varphi_{5,3}$ with scaling factors $\alpha = 0.6$, 1.0, and 1.4 are provided in Table 3. The error by the finite element method is 4.3E-05 and the error by the DBFs $I_{30,9}$ with the approximate radius of support 0.286 is 7.0E-07 [18].

TABLE 3. Example 1 results by CS-RBFs of different scaling factors

	$\alpha = 0.6$	$\alpha = 1.0$	$\alpha = 1.4$
$\varphi_{3,1}$	5.3E-03	1.1E-03	8.5E-04
$\varphi_{4,2}$	2.0E-03	3.9E-04	1.8E-04
$\varphi_{5,3}$	1.1E-03	1.6E-04	6.3E-05

Example 4.2. We consider the problem,

$$\Delta u\left(x, y\right) - 10u\left(x, y\right) = f\left(x, y\right), \ (x, y) \in \Omega, \tag{4.3}$$

$$u\left(x, y\right) = g\left(x, y\right), \ (x, y) \in \partial\Omega, \tag{4.4}$$

where $\Omega = [-0.5, 0.5]^2$, $f\left(x, y\right) = -14 + 10x^2 + 10y^2$, and $g\left(x, y\right) = 1 - x^2 - y^2$. The exact solution is $u\left(x, y\right) = 1 - x^2 - y^2$. Numerical results by CS-RBFs and DBFs are provided in Tables 4–7. The number of collocation points is twice as that of center points, i.e., $N = 2K$. In the calculation, we let $N_b = 100$. For the two-level approach, $K_1 + K_2 = K$ with K_2 corresponding to the basis of the smaller scaling factor. According to Table 2, the approximate radii of support for the DBFs $I_{10,4}$ and $I_{20,6}$ are respectively 0.473 and 0.319. For comparison purpose, we first choose $\alpha_1 = 0.473$ and $\alpha_2 = 0.319$ for the CS-RBF $\varphi_{4,2}$ collocation method, and we show the results in Table 4. The results by $I_{10,4}$ and $I_{20,6}$ are displayed in Table 5. The DBF method shows much better acuracy. The results of CS-RBF method are improved when we use CS-RBFs $\varphi_{4,2}$ with much larger scaling factors (see Table 6). Although CS-RBF $\varphi_{5,3}$ is of higher differentiability, it only produces slightly better results than $\varphi_{4,2}$ (see Table 7). In the CS-RBF one-level collocation, the basis functions of larger scaling factors attain better accuracy compared with those with smaller scaling factors. We notice that the combination of CS-RBFs of two different scaling factors ($\alpha = 0.8$, $\alpha = 1.0$) in Tables 6–7 helps improve the solution. They, however, do not outperforme DBFs.

TABLE 4. Example 2 results of one-level and two-level CS-RBFs

K	150	300	450
E by $\varphi_{4,2}$, $\alpha = 0.473$	3.6E-01	3.0E-01	2.3E-01
E by $\varphi_{4,2}$, $\alpha = 0.319$	8.9E-01	5.4E-01	4.9E-01
(K_1, K_2)	$(50, 100)$	$(100, 200)$	$(100, 350)$
E by $\varphi_{4,2}$, $\alpha_1 = 0.473$, $\alpha_2 = 0.319$	5.1E-01	3.1E-01	2.5E-01

TABLE 5. Example 2 results of one-level and two-level DBFs

K	150	300	450
E by $I_{10,4}$	3.3E-04	5.1E-04	5.0E-04
E by $I_{20,6}$	2.0 E-02	1.4E-05	2.4E-07
(K_1, K_2)	$(50, 100)$	$(100, 200)$	$(100, 350)$
E by $I_{10,4}$&$I_{20,6}$	6.6E-04	1.6E-07	1.9E-07

TABLE 6. Example 2 results of one-level and two-level CS-RBFs of larger scaling factors

K	150	300	450
E by $\varphi_{4,2}$, $\alpha = 1.0$	2.9E-02	4.0E-02	1.3E-02
E by $\varphi_{4,2}$, $\alpha = 0.8$	7.5E-02	9.4E-02	4.3E-02
(K_1, K_2)	$(50, 100)$	$(100, 200)$	$(100, 350)$
E by $\varphi_{4,2}$, $\alpha_1 = 1.0$, $\alpha_2 = 0.8$	1.0E-02	5.1E-03	2.2E-03

TABLE 7. Example 2 results of one-level and two-level CS-RBFs of higher smoothness

K	150	300	450
E by $\varphi_{5,3}$, $\alpha = 1.0$	2.2E-02	2.6E-02	5.4E-03
E by $\varphi_{5,3}$, $\alpha = 0.8$	7.0E-02	7.5E-02	2.6E-02
(K_1, K_2)	$(50, 100)$	$(100, 200)$	$(100, 350)$
E by $\varphi_{5,3}$, $\alpha_1 = 1.0$, $\alpha_2 = 0.8$	1.3E-02	1.2E-03	4.6E-04

When choosing a value for the scaling factor of a CS-RBF, we need to be aware of the trade-off principle as pointed out in [3]. When the scaling factor α is too small, the error is large; when α is too large, the matrix is dense and the computation is not efficient. Here, in order to get acceptable results by CS-RBFs collocation method, the value of α has to be very large which defeats the purpose of getting a sparse matrix.

Example 4.3. Since Franke's function is often used as a benchmark problem [6, 19, 23], we let f and g in (4.3)–(4.4) be given such that the exact solution is the rescaled Franke's function on $\Omega = [-0.5, 0.5]^2$,

$$u(x,y) = \frac{3}{4}\exp\left(-\frac{(9x+2.5)^2+(9y+2.5)^2}{4}\right) + \frac{3}{4}\exp\left(-\frac{(9x+5.5)^2+(9y+5.5)^2}{49}\right)$$
$$+ \frac{1}{2}\exp\left(-\frac{(9x-2.5)^2+(9y+1.5)^2}{4}\right) - \frac{1}{5}\exp\left(-(9x+0.5)^2-(9y-2.5)^2\right).$$

The source function f is highly oscillative with a large amplitude over the domain Ω. We list in Table 8 the one-level, two-level, and three-level results with scaling factors $\alpha_1 = 1.0$, $\alpha_2 = 0.8$, and $\alpha_3 = 0.6$ by CS-RBFs $\varphi_{3,1}$, $\varphi_{4,2}$ and $\varphi_{5,3}$ respectively. The one-level and two-level results by DBFs $I_{10,4}$, $I_{30,9}$, $I_{40,12}$, and $I_{50,14}$ are given in Table 9. The numbers of interior collocation points are 100, 600, and 1200 respectively for one-level, two-level, and three-level collocation methods, and $N_b = 200$ for all cases. In Table 8, the CS-RBFs with higher order smoothness help improve the accuracy of the numerical solution. With the same number of center points, the three-level CS-RBFs collocation method achieves slightly better accuracy than its two-level method. For the two-level DBF method in Table 9, the approximate radii of support for the pair of $I_{10,4}$ and $I_{30,9}$ are 0.473 and 0.286, which are much smaller compared with $\alpha_1 = 1.0$ and $\alpha_2 = 0.8$ for the CS-RBFs. No useful results can be obtained when 0.473 and 0.286 are used for α_1 and α_2 in CS-RBFs. Note that when using the pair of $I_{10,4}$ and $I_{50,14}$, their approximate radii are only 0.473 and 0.243. Although a larger shape parameter M involves more terms to construct the delta-shaped basis, their effectiveness in capturing oscillative details pays off as shown in Table 9.

TABLE 8. Example 3 results of one-level, two-level, and three-level CS-RBFs

K	K_1	K_2	K_3	E by $\varphi_{3,1}$	E by $\varphi_{4,2}$	E by $\varphi_{5,3}$
100	100	0	0	2.8E-02	2.7E-01	2.0E-01
400	100	300	0	7.3E-02	5.0E-03	2.3E-03
900	100	800	0	1.2E-02	4.2E-04	9.1E-05
900	100	300	500	5.2E-03	2.9E-04	7.2E-05

TABLE 9. Example 3 results of one-level and two-level DBFs

K	K_1	K_2	E by $I_{10,4}\&I_{30,9}$	E by $I_{10,4}\&I_{40,12}$	E by $I_{10,4}\&I_{50,14}$
100	100	0	2.1E-01	2.1E-01	2.1E-01
400	100	300	4.1E-04	1.7E-04	7.8E-05
900	100	800	1.2E-05	3.0E-07	2.1E-08

5. CONCLUDING REMARKS

The globally supported radial basis functions often results in a dense and highly ill-conditioned matrix. The compactly supported RBFs and the approximately compactly supported DBFs have the computational advantage of a less dense and less ill-conditioned matrix. This paper compares DBFs with CS-RBFs and provides some results of these basis functions for solving elliptic boundary value problems. An optimal α of CS-RBFs should balance the accuracy and computational efficiency. However, in the CS-RBFs collocation for the elliptic problems, the values of α have to be very large in order for us to get reasonable results. For large α, the matrix is nearly dense. The DBFs collocation in general produces a much more acurate solution, which is especially true for situations that require a large number of basis functions and collocation points.

REFERENCES

[1] C. S. Chen, M. Ganesh, M. A. Golberg and A. H.-D. Cheng, Multilevel compact radial functions based computational schemes for some elliptic problems, *Comput. Math. Appl.*, 43: 359–378, 2002.

[2] C. S. Chen, M. A. Golberg and R. S. Schaback, Recent developments of the dual reciprocity method using compactly supported radial basis functions, *Transformation of Domain Effects to the Boundary*, ed. Y.F. Rashed, WIT Press, Boston, 183–225, 2003.

[3] G. E. Fasshauer, Solving differential equations with radial basis functions: multilevel methods and smoothing, *Adv. Comput. Math.*, 11: 139–159, 1999.

[4] G. E. Fasshauer, *Meshfree approximation methods with MATLAB*, *Interdisciplinary Mathematical Sciences* Vol 6, World Scientific, Singapore, 2007.

[5] M. S. Floater and A. Iske, Multistep scattered data interpolation using compactly supported radial basis functions, *J. Comput. Appl. Math.*, 73: 65–78, 1996.

[6] R. Franke, Scattered data interpolation: test of some methods, *Math. Comput.*, 38: 181–200, 1982.

[7] S. L. Ho, S. Yang, G. Ni and H. C. Wong, A fast global optimizer based on improved CS-RBF and stochastic optimal algorithm, *IEEE Trans. Magn.*, 42: 1175–1178, 2006.

[8] E. J. Kansa, Multiquadrics - A scattered data approximation scheme with applications to computational fluid-dynamics–I. Surface approximations and partial derivative estimates, *Comput. Math. Appl.*, 19: 127–145, 1990.

[9] 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: 147–161, 1990.

[10] J. Li, Application of radial basis meshless methods to direct and inverse biharmonic boundary value problems, *Comm. Numer. Methods Engrg.*, 21: 169–182, 2005.

[11] J. Li, S. Luo, Y. Qi and Y. Huang, Numerical solution of elliptic partial differential equation by growing radial basis function neural networks, *Proceedings of the International Joint Conference on Neural Networks*, 85–90, 2003.

[12] D. Nardini and C. A. Brebbia, A new approach to free vibration analysis using boundary elements, *Appl. Math. Modelling*, 7: 157–162, 1983.

[13] Y. Ohtake, A. Belyaev, and H. P. Seidel, 3D Scattered data approximation with adaptive compactly supported radial basis functions, *Proceedings of the Shape Modeling International*, 1–9, 2004.

[14] P. W. Partridge, C. A. Brebbia and L. C. Wrobel, *The dual reciprocity boundary element methods*, Computational Mechanics Publications, Southampton, UK, 1992.

[15] S. Reutskiy, A boundary method of Trefftz type for PDEs with scattered data, *Eng. Anal. Boundary Elem.*, 29: 713–724, 2005.

[16] S. Reutskiy, C. S. Chen and H. Y. Tian, A boundary meshless method using Chebyshev interpolation and trigonometric basis function for solving heat conduction problems, *Internat. J. Numer. Methods Engrg.*, 74: 1621–1644, 2008.

[17] C. Shu, H. Ding and K. S. Yeo, Solution of partial differential equations by a global radial basis function-based differential quadrature method, *Eng. Anal. Boundary Elem.*, 28: 1217–1226, 2004.

[18] H. Y. Tian, Computations of critical domains for quenching problems by delta-shaped basis functions, *Neural Parallel Sci. Comput.*, 15: 501–514, 2007.

[19] H. Y. Tian, S. Reutskiy and C. S. Chen, A basis function for approximation and the solutions of partial differential equations, *Numer. Methods Partial Differential Equations*, 24: 1018–1036, 2008.

[20] H. Wendland, Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree, *Adv. Comput. Math.*, 4: 389–396, 1995.

[21] H. Wendland, Error estimates for interpolation by compactly supported radial basis functions of minimal degree, *J. Approx. Theory*, 93: 258–272, 1998.

[22] Z. Wu, Compactly supported positive definite radial functions, *Adv. Comput. Math.*, 4: 283–292, 1995.

[23] X. Li, C. H. Ho and C. S. Chen, Computational test of approximation of functions and their derivatives by radial basis functions, *Neural Parallel Sci. Comput.*, 10: 25–46, 2002.