1-1-2017

Polynomial Particular Solutions for Solving Elliptic Partial Differential Equations

Thir R. Dangal
University of Southern Mississippi, thir.dangal@usm.edu

Ching-Shyang Chen
University of Southern Mississippi, CS.Chen@usm.edu

Ji Lin
Hohai University, linji861103@126.com

Recommended Citation

Dangal, T. R., Chen, C., Lin, J. (2017). Polynomial Particular Solutions for Solving Elliptic Partial Differential Equations. *Computers & Mathematics with Applications, 73*(1), 60-70.
Available at: https://aquila.usm.edu/fac_pubs/15309

Polynomial Particular Solutions for Solving Elliptic Partial Differential Equations

Thir Dangal *, C.S. Chen*, Ji Lin †, ‡

Abstract

In the past, polynomial particular solutions have been obtained for certain types of partial differential operators without convection terms. In this paper, a closed-form particular solution for more general partial differential operators with constant coefficients has been derived for polynomial basis functions. The newly derived particular solution is further coupled with the method of particular solutions (MPS) for numerically solving a large class of elliptic partial differential equations. In contrast to the use of Chebyshev polynomial basis functions, the proposed approach is more flexible in selecting the collocation points inside the domain. The polynomial basis functions are well-known for yielding ill-conditioned systems when their order becomes large. The multiple scale technique is applied to circumvent the difficulty of ill-conditioning problem. Five numerical examples are presented to demonstrate the effectiveness of the proposed algorithm.

Keywords: method of approximate particular solutions, polynomial basis function, multiple scale technique, particular solution, radial basis functions

1 Introduction

The derivation of particular solutions has played a key role for solving various types of differential equations. In general, for a given differential equation, if the particular

*Department of Mathematics, University of Southern Mississippi, Hattiesburg, MS 39406, USA

†State Key Laboratory of Hydrology-Water Resources and Hydraulic Engineering, International Center for Simulation Software in Engineering and Sciences, College of Mechanics and Materials, Hohai University, Nanjing, 211100, China

‡Corresponding author: Ji Lin, Email: linji861103@126.com

solution and homogenous solution can be obtained, the problem is considered to be solved [2]. However, it is a challenge to obtain a particular solution and the homogeneous solution is not always available. It is well-known that the particular solution of a given differential equation is not unique and there are numerous ways to find a particular solution [1, 4, 6, 8, 11, 13] for various differential operators and basis functions.

Consider the following partial differential equation:

$$Lu_p(x,y) = f(x,y)$$

where L is a given linear differential operator with constant coefficients and $f(x,y)$ is a given function. For a general function $f(x,y)$, the closed-form particular solution $u_p(x,y)$ is difficult, if not impossible, to obtain. Consequently, the approximate particular solution is often needed. Over the past two decades, many numerical methods have been proposed for the approximation of the particular solution [3, 4, 5, 7, 8]. In recent years, radial basis functions (RBFs) have been successfully employed for the construction of the approximate particular solutions. Due to the rapid development in this area, the method of particular solutions has been established [3, 4, 14] in the context of RBFs and has been applied for solving a large class of partial differential equations in science and engineering. Despite the success of the use of RBFs, there are still challenges such as the determination of the shape parameter of RBFs and the difficulty in deriving the closed-form particular solutions [6, 11, 12, 15]. As a result, Chebyshev polynomial functions have been adopted as an alternative to alleviate some of these difficulties [4, 5, 13]. These approaches have been proven to be highly accurate. However, the solution procedure is quite tedious and the closed-form particular solutions are only available for some specific differential operators. One of the disadvantages of using Chebyshev polynomials as the basis functions is the requirement that the forcing term of the differential equation should be smoothly extendable to the exterior of the domain for the case of non-rectangular domains. As such, the collocation points can be selected at the specific Gauss-Lobatto points.

In this paper, the closed-form particular solution using the standard polynomial basis function of order s $\{x^{i-j}y^j\}, 0 \leq i \leq s, 0 \leq j \leq i$, under a general linear differential operator has been derived. Coupling with the MPS using the newly derived particular solution, a large class of partial differential equations have been simulated. One of the clear advantages of the proposed approach using the standard polynomial basis over the Chebyshev polynomial basis is that the collocation points can be distributed arbitrarily inside the computational domain without the need for fictitious collocation points outside the domain. Hence, the applicability of the proposed method is wider. Furthermore, the proposed method can be easily coupled with the MPS which allows us to solve more general types of partial differential equations.

It is known that the polynomial basis functions are notoriously unstable when the order of the polynomial basis becomes higher. As a result, the polynomial basis functions are not ideal for a global approach since the resultant matrix is extremely ill-conditioned when the order of the polynomial basis is getting higher. Hence, our derived closed-form particular solution is useless without proper treatment of the matrix resulting from our formulation. There are various types of pre-conditioners in the literature. In this paper, we adopt the so-called multiple scale technique [9, 10] which is a pre-conditioning technique to reduce the condition number of the resultant matrix of the MPS. As we shall see in our presented numerical results, the multiple scale technique is very effective for the reduction of the condition number of our formulated matrix system and thus allows our proposed algorithm to successfully solve various kinds of boundary value problems.

The paper is organized as follows. In Section 2, we derive the closed-form particular solution for the general differential equation using polynomial basis functions. In Section 3, we give a brief review of the MPS in the context of polynomial basis functions. In Section 4, the multiple scale technique is re-introduced to reduce the condition number of the resultant matrix through the MPS and ensure the proposed method is effective. In Section 5, we present the results of five numerical examples to demonstrate the effectiveness of the proposed algorithm. Finally, some conclusions and ideas for future work are outlined in Section 5.

2 Particular solution of polynomial basis

In this section, we consider a polynomial basis and find the particular solutions of the basis functions for general partial differential operators. For simplicity, let us consider the 2D case. It is well-known that a polynomial basis of degree $\leq s$ can be written as follows:

$$\mathbf{P}_s^2 = \{x^{i-j}y^j : \ 0 \leq j \leq i, 0 \leq i \leq s\}. \tag{1}$$

Note that $w = (s+1)(s+2)/2$ is the number of polynomial basis functions in $\mathbf{P}_s^2$. The superscript and subscript of $\mathbf{P}$ in equation (1) are the dimension of the considered problem and the order of the polynomial basis functions, respectively.

To illustrate the core idea of the proposed method, we shall give a simple example on how to derive the particular solution explicitly for a general partial differential equation with constant coefficients. Let us consider the following differential equation:

$$(L - 3I)u_p = x^2y^2, \tag{2}$$

where

$$L = \left(\Delta + \frac{\partial}{\partial x} + \frac{\partial}{\partial y}\right), \tag{3}$$

and I is an identity operator. Equation (2) can be rewritten as follows:

$$\left(I - \frac{L}{3}\right)(-3u_p) = x^2y^2. \tag{4}$$

Since L is a differential operator, any polynomial of finite order can be annihilated by L in finite times. In other words, we observe that $L^{m+n+1}(x^my^n) = 0$. Hence, the following identity always hold:

$$\left(I - \frac{L^5}{3^5}\right)x^2y^2 = x^2y^2.$$

By simple algebraic factorization, it follows that:

$$\left(I - \frac{1}{3}L\right)\left(I + \frac{L}{3} + \frac{L^2}{3^2} + \frac{L^3}{3^3} + \frac{L^4}{3^4}\right)x^2y^2 = x^2y^2. \tag{5}$$

Then, comparing the left hand side of (4) and (5), we have

$$-3u_p = \left(I + \frac{L}{3} + \frac{L^2}{3^2} + \frac{L^3}{3^3} + \frac{L^4}{3^4}\right)x^2y^2,$$

or

$$u_p = \frac{-1}{3}\left(I + \frac{L}{3} + \frac{L^2}{3^2} + \frac{L^3}{3^3} + \frac{L^4}{3^4}\right)x^2y^2. \tag{6}$$

The particular solution u_p in (6) is actually computable by simply taking the derivatives and summing them up. Hence, the particular solution of (2) can be computed explicitly as follows:

$$u_p = -\frac{1}{3}x^2y^2 - \frac{2}{9}x^2y - \frac{2}{9}xy^2 - \frac{8}{27}x^2 - \frac{8}{27}y^2 - \frac{8}{27}xy - \frac{4}{9}x - \frac{4}{9}y - \frac{56}{81}.$$

Based on the above observation, we have the following theorem.

Theorem 1. Consider a general form second order linear partial differential equation in two variables with constant coefficients:

$$a_1\frac{\partial^2 u_p}{\partial x^2}+a_2\frac{\partial^2 u_p}{\partial x\partial y}+a_3\frac{\partial^2 u_p}{\partial y^2}+a_4\frac{\partial u_p}{\partial x}+a_5\frac{\partial u_p}{\partial y}+a_6 u_p=x^m y^n, \tag{7}$$

where $\{a_i\}_{i=1}^6$ are real constants, $a_6 \neq 0$ and m and n are positive integers. Then the polynomial particular solution of (7) is given by

$$u_p=\frac{1}{a_6}\sum_{k=0}^{\mathcal{N}}\left(\frac{-1}{a_6}\right)^k \mathfrak{L}^k(x^m y^n), \tag{8}$$

where $\mathcal{N}=m+n$ and

$$\mathfrak{L}=a_1\frac{\partial^2}{\partial x^2}+a_2\frac{\partial^2}{\partial x\partial y}+a_3\frac{\partial^2}{\partial y^2}+a_4\frac{\partial}{\partial x}+a_5\frac{\partial}{\partial y}.$$

Proof. Equation (7) can be written as

$$(\mathfrak{L}+a_6 I)\,u_p=x^m y^n, \tag{9}$$

which implies

$$\left(I+\frac{\mathfrak{L}}{a_6}\right)(a_6 u_p)=x^m y^n. \tag{10}$$

Since $\mathfrak{L}$ is a differential operator containing various partial derivatives, it is clear that $\mathfrak{L}^{m+n+1}(x^m y^n)=0$. Hence, the following identity always holds:

$$\left(I+\left(\frac{\mathfrak{L}}{a_6}\right)^{\mathcal{N}+1}\right)x^m y^n \;=\; x^m y^n, \tag{11}$$

where $\mathcal{N}=m+n$. By direct algebraic factorization, we have

$$I+\left(\frac{\mathfrak{L}}{a_6}\right)^{\mathcal{N}+1}=\left(I+\frac{\mathfrak{L}}{a_6}\right)\sum_{k=0}^{\mathcal{N}}\left(\frac{-1}{a_6}\right)^k\mathfrak{L}^k. \tag{12}$$

From (11) and (12), we have

$$\left(I+\frac{\mathfrak{L}}{a_6}\right)\sum_{k=0}^{\mathcal{N}}\left(\frac{-1}{a_6}\right)^k\mathfrak{L}^k(x^m y^n)=x^m y^n. \tag{13}$$

Comparing (10) and (13), it follows that

$$a_6 u_p=\sum_{k=0}^{\mathcal{N}}\left(\frac{-1}{a_6}\right)^k\mathfrak{L}^k(x^m y^n).$$

Consequently, the particular solution u_p for the above general differential operator is given by

$$u_p = \frac{1}{a_6} \sum_{k=0}^{\mathcal{N}} \left(\frac{-1}{a_6} \right)^k \mathcal{L}^k (x^m y^n). \tag{14}$$

□

The following algorithm is presented for finding the particular solution of the basis function $x^m y^n$ for the above operator.

Algorithm 1

Step 1:	`Let` $p(x,y) = x^m y^n$`,` m, n`: nonnegative integers.`
Step 2:	Let partsol = 0, coef = 1,
	and $\mathcal{L} = a_1 \frac{\partial^2}{\partial x^2} + a_2 \frac{\partial^2}{\partial x \partial y} + a_3 \frac{\partial^2}{\partial y^2} + a_4 \frac{\partial}{\partial x} + a_5 \frac{\partial}{\partial y}$.
	`for` $k = 1, 2, ..., m+n$
	`term` $= \mathcal{L}p$
	`coef = -coef/`a_6
	`partsol = partsol + coef * term`
	$p =$ `term`
	`end`
Step 3:	`The required particular solution of` $x^m y^n$ `for`
	`the operator` $\mathcal{L}$ `is given by` $\frac{1}{a_6}$`(partsol +` $x^m y^n$`).`

3 The method of particular solutions (MPS)

Once the particular solution of the associated differential operator is available, the method of particular solutions (MPS) can be employed to solve the boundary value problem related to the differential operator. In this section, we will give a brief review of the MPS. Let $f(x,y)$ and $g(x,y)$ be given functions. Consider the following boundary value problem

$$\mathcal{L}u(x,y) = f(x,y), (x,y) \in \Omega, \tag{15}$$

$$\mathcal{B}u(x,y) = g(x,y), (x,y) \in \Gamma, \tag{16}$$

where $\mathcal{L}$ is a linear elliptic partial differential operator, $\mathcal{B}$ is a boundary differential operator, and Ω is a closed and bounded domain with boundary Γ.

To discretize the given partial differential equation, we employ the MPS using a polynomial basis. In the MPS, we assume that the solution of (15)–(16) can be represented by:

$$u(x,y) \simeq \hat{u}(x,y) = \sum_{i=0}^{s}\sum_{j=0}^{i} a_{ij} u_p^{ij}(x,y), \tag{17}$$

where

$$\mathcal{L}u_p^{ij}(x,y) = x^{i-j}y^j, \ 0 \le j \le i, 0 \le i \le s. \tag{18}$$

Let $\{(x_i,y_i)\}_{i=1}^{n_i}$ be the set of interior points in the domain Ω and $\{(x_i,y_i)\}_{i=n_i+1}^{n}$ be the boundary points on Γ, and $n = n_i + n_b$. Applying (17) to (15), we obtain

$$\sum_{i=0}^{s}\sum_{j=0}^{i} a_{ij}\mathcal{L}u_p^{ij}(x_k,y_k) = f(x_k,y_k), \ k = 1,2,...,n_i. \tag{19}$$

From (18), the above equation becomes

$$\sum_{i=0}^{s}\sum_{j=0}^{i} a_{ij}x_k^{i-j}y_k^j = f(x_k,y_k), \ k = 1,2,...,n_i. \tag{20}$$

In the MPS, the governing differential equation (15) has been transformed to a simple data interpolation problem as shown in (20). Imposing (17) to satisfy the boundary condition (16), we obtain

$$\sum_{i=0}^{s}\sum_{j=0}^{i} a_{ij}\mathcal{B}u_p^{ij}(x_k,y_k) = g(x_k,y_k), \ k = n_i+1, n_i+2,...,n. \tag{21}$$

To ensure that the system of equations (20) – (21) is solvable, the total number of collocation points n has to be larger than $(s+1)(s+2)/2$. The method of least squares will be adopted to solve the above system. Once the undetermined coefficients

$$\{a_{ij}\} = \{a_{00}, a_{10}, a_{11}, a_{20}, a_{21}, a_{22}, \cdots, a_{ss}\}$$

are determined, the approximate solution $\hat{u}$ can be obtained from (17).

4 Multiple Scale Technique

High order polynomials are notorious for numerical interpolation due to the severe ill-conditioning of the resulting matrix. The MPS using polynomials as basis functions has experienced the same difficulty and a special treatment of the resultant matrix system (20) – (21) is required. To alleviate this difficulty, a multiple scale technique [9, 10] is applied to reduce the condition number of the resulting matrix.

Let $w = (s+1)(s+2)/2$. Equations (20) – (21) can be written in the matrix form

$$\mathbf{A}\mathbf{c} = \mathbf{b}, \tag{22}$$

where

$$\mathbf{A} = \begin{bmatrix} \left[x^{i-j}y^j\right]_{n_i \times w} \\ \\ \left[\mathcal{B}u_p^{ij}\right]_{n_b \times w} \end{bmatrix}, \mathbf{c} = \begin{bmatrix} a_{00} \\ a_{10} \\ \vdots \\ a_{ss} \end{bmatrix}, \mathbf{b} = \begin{bmatrix} f(x_1, y_1) \\ \vdots \\ f(x_{n_i}, y_{n_i}) \\ g(x_{n_{i+1}}, y_{ni+1}) \\ \vdots \\ g(x_n, y_n) \end{bmatrix}.$$

Let

$$\mathbf{A} = [A_1 \; A_2 \; \cdots \; A_w] \quad \text{and} \quad R_k = \|A_k\|_2, \; k = 1, 2, \cdots, w,$$

where A_k is the k^{th} column of matrix $\mathbf{A}$. In the multiple scale technique, the linear system (22) is equivalent to

$$\widetilde{\mathbf{A}}\,\widetilde{\mathbf{c}} = \mathbf{b}, \tag{23}$$

where

$$\widetilde{\mathbf{A}} = \left[\frac{A_1}{R_1} \; \frac{A_2}{R_2} \; \cdots \; \frac{A_w}{R_w}\right]$$

and

$$\begin{aligned} \widetilde{\mathbf{c}} &= \left[\widetilde{a}_{00} \; \widetilde{a}_{10} \; \cdots \; \widetilde{a}_{ss}\right]^T \\ &= \left[a_{00}R_1 \; a_{10}R_2 \; \cdots \; a_{ss}R_w\right]^T. \end{aligned} \tag{24}$$

Note that $\widetilde{A}$ is now better conditioned due to the reduction of round-off errors. Once (23) is solved, the $\{a_{ij}\}$ in $\mathbf{c}$ can be recovered from $\widetilde{\mathbf{c}}$ in (24); i.e.,

$$a_{00} = \frac{\widetilde{a}_{00}}{R_1}, \; a_{10} = \frac{\widetilde{a}_{10}}{R_2}, \; \cdots \;, \; a_{ss} = \frac{\widetilde{a}_{ss}}{R_w}.$$

We refer readers to references [9, 10] and the references cited therein for further details.

5 Numerical Results

To validate our proposed MPS algorithm using a polynomial basis, five numerical examples in 2D are given. For the numerical implementation, we have considered both regular and irregular domains. The parametric equations of the first three irregular boundaries $\partial\Omega$ are defined as follows:

$$\partial\Omega = \{(x, y)|x = r(\vartheta)\cos(\sigma(\vartheta)), y = r(\vartheta)\sin(\sigma(\vartheta)), 0 \leq \vartheta < 2\pi\},$$

where

- $$r(\vartheta) = \mathrm{e}^{\sin\vartheta}\sin^2(2\vartheta) + \mathrm{e}^{\cos\vartheta}\cos^2(2\vartheta) \tag{25}$$

  is the amoeba-like boundary.

- $$r(\vartheta) = \left(\cos(4\vartheta) + \sqrt{\frac{18}{5} - \sin^2(4\vartheta)}\right)^{1/3} \tag{26}$$

  is the Cassini-shaped domain.

- $$r(\vartheta) = 2 + \frac{1}{2}\sin(6\vartheta), \sigma(\vartheta) = \vartheta + \frac{1}{2}\sin(6\vartheta) \tag{27}$$

  is the gear-shaped domain.

Two additional domains in which the parametric equations are not available considered in this section are the L-shaped and the corner-shaped domains. The profiles of these domains are shown in Figure 1.

The root-mean-squared error ($RMSE$), the root-mean-squared error of the derivative with respect to x ($RMSEx$), the maximum absolute error (MAE), and the relative error (Rel Err) are used to measure the accuracy of the solutions. They are defined as follows

$$RMSE = \sqrt{\frac{1}{n_t}\sum_{j=1}^{n_t}(\hat{u}_j - u_j)^2},$$

$$RMSEx = \sqrt{\frac{1}{n_t}\sum_{j=1}^{n_t}(\frac{\partial\hat{u}_j}{\partial x} - \frac{\partial u_j}{\partial x})^2},$$

$$MAE = \max_{1 \le j \le n_t} |\hat{u}_j - u_j|$$

and

$$Rel\ Err = \sqrt{\frac{\sum_{j=1}^{n_t} (\hat{u}_j - u_j)^2}{\sum_{j=1}^{n_t} u_j^2}}$$

where n_t is the number of test points in the domain and $\hat{u}_j$ and u_j are the approximate solution and exact solution at the j^{th} test point, respectively.

The generation of the particular solutions in (8) with respect to all polynomial basis functions requires symbolic computation. In the spirit of reproductive research, we provide a MATLAB© code in the Appendix for the generation of the particular solution with the differential operator shown in Example 4. Prior to solving the partial differential equation, we symbolically compute and save all the particular solutions in a table for later use since the generation of a particular solution as shown in Algorithm 1 is the most time consuming part of the solution process. Once the polynomial particular solution is produced and saved in a table, the given boundary value problem can be solved efficiently.

Example 1. In this example, we consider the following differential equation in the unit square.

$$\Delta u(x,y) - \frac{\partial^2 u(x,y)}{\partial x \partial y} + \frac{\partial u(x,y)}{\partial y} - u(x,y) = f(x,y),\ (x,y) \in \Omega, \tag{28}$$

$$u(x,y) = g(x,y),\ (x,y) \in \partial\Omega, \tag{29}$$

where $f(x,y)$ and $g(x,y)$ are given based on the following analytical solution

$$u(x,y) = e^{2x} \cos(y),\ (x,y) \in \overline{\Omega}.$$

The number of interior points, boundary points and the test points are 400, 108 and 230 respectively. In Figure 2, we show the condition number and RMSE for various orders of the polynomial basis functions with and without the use of the multiple scale technique. From these figures, we clearly see that the multiple scale technique plays an important role in the reduction of the condition number of the collocation matrix. We also observe the improvement of the accuracy when using higher order polynomial basis. One important feature of the proposed algorithm is the numerical stability. When the order of polynomial basis becomes higher, the numerical accuracy remains stable. It is clear that without implementing the multiple scale technique, the proposed approach would fail due to an extremely high condition number.

Table 1 shows the RMSE and maximum errors for various sets of interior and boundary points with a polynomial basis of order 11. From this table, we observe that the increasing of the interior and boundary points does not contribute to the increasing of accuracy. In contrast, Figure 2 shows that the increasing of the order of the polynomial basis function significantly improves the accuracy. For a polynomial basis with order 11, there are (11+1)(11+2)/2=78 basis functions. Hence, the minimum number of the interior and boundary points should be at least 78. From the first row of Table 1, it is shown that we can achieve good accuracy using only 80 interior points and 20 boundary points. It is noted here that it takes 76.64 seconds for generating the particular solutions of 30 order of the polynomial basis functions while it takes less than one second for obtaining the numerical approximations as shown in Table 1. Since the computational cost is relatively low for such a small number of collocation points, we will double this number in the numerical implementation.

Table 1: Example 1: The RMSE and Maximum errors for different numbers of interior and boundary points with polynomial basis of order 11.

(n_i, n_b)	RMSE	MAE	Elapsed time
$(81, 20)$	$5.265e-08$	$4.530e-07$	$1.847e-01$
$(121, 28)$	$6.405e-08$	$1.760e-07$	$1.870e-01$
$(169, 64)$	$5.749e-10$	$2.389e-09$	$2.096e-01$
$(361, 88)$	$1.666e-09$	$5.702e-09$	$2.238e-01$
$(576, 116)$	$8.325e-09$	$2.497e-08$	$2.342e-01$
$(1089, 316)$	$1.468e-08$	$4.912e-08$	$2.430e-01$
$(1444, 556)$	$6.749e-09$	$2.658e-08$	$2.527e-01$
$(2304, 636)$	$2.056e-08$	$7.318e-08$	$2.639e-01$

Since the multiple scale technique is essential in overcoming the ill-conditioning of the resultant matrix and meanwhile improves the numerical accuracy, we will continue to use such a technique in the rest of the examples in this section.

Example 2. Let us consider the following Helmholtz problem:

$$\Delta u(x,y) + u(x,y) = f(x,y), \ (x,y) \in \Omega, \quad (30)$$

$$u(x,y = g(x,y), \ (x,y) \in \partial\Omega, \quad (31)$$

where $f(x,y)$ and $g(x,y)$ are given based on the following analytical solution

$$u(x,y) = \sin(2x)\cos(2y), \ (x,y) \in \overline{\Omega}.$$

The domain is the amoeba-like domain as shown in Figure 1(a).

For the numerical approximation, the number of interior points, boundary points and test points are 412, 100 and 257 respectively. As we have mentioned in the previous example, the number of collocation points depends on the order of the polynomial basis functions. For simplicity, we choose the same number of collocation points for the case of polynomial order equal to 30. To demonstrate the effectiveness of the proposed algorithm, we make a comparison with the MPS using the MQ (Multiquadric) radial basis function. In Figure 3, we observe that our proposed approach is not only more accurate but also more stable than the MPS using MQ. As shown in Figure 3(b), the higher order of polynomial does not cause any problem in stability due to the use of multiple scale scheme. On the other hand, the uncertainty of the shape parameter as shown in Figure 3(a) is an additional challenge for the MPS using radial basis functions.

Example 3. Let us consider the following mixed boundary value problem:

$$\Delta u(x,y) + u(x,y) = f(x,y), \ \ (x,y) \in \Omega, \quad (32)$$

$$u(x,y) = g(x,y), \ \ (x,y) \in \partial\Omega^D, \quad (33)$$

$$\frac{\partial u}{\partial n} = \nabla u \cdot \mathbf{n}, \ \ (x,y) \in \partial\Omega^N, \quad (34)$$

where $\mathbf{n}$ is the unit outward normal vector, $f(x,y)$ and $g(x,y)$ are given based on the following analytical solution

$$u(x,y) = e^{2x+2y}, \ (x,y) \in \overline{\Omega}.$$

The boundaries $\partial\Omega^D$ and $\partial\Omega^N$ denote the boundaries on which the Dirichlet and Neumann conditions are applied respectively such that $\partial\Omega = \partial\Omega^D \cup \partial\Omega^N, \partial\Omega^D \cap \partial\Omega^N = \emptyset$. As shown in Figure 4, $\partial\Omega^N$ is located in the fourth quadrant; i.e., $3\pi/2 \leq \theta < 2\pi$.

In the numerical implementation, we choose 545 uniformly distributed interior points, 116 points on the Dirichlet boundary, 34 points on the Neumann boundary, and 350 randomly selected test points inside the domain.

We observed the similar results in accuracy and stability for the previous example. Due to the mixed boundary conditions, the accuracy is even more sensitive to the shape parameter in the case of the MPS using MQ as shown in Figure 5(a). On the other hand, as shown in Figure 5(b), not only the accuracy using a polynomial basis is much better than the MPS using MQ but also the stability, which is extremely important in the numerical computation.

Example 4. In this example, we perform the numerical tests on various geometric domains as shown in Figure 1. We consider the following partial differential equation.

$$2\frac{\partial^2 u(x,y)}{\partial x^2} + 3\frac{\partial^2 u(x,y)}{\partial x \partial y} + 3\frac{\partial^2 u(x,y)}{\partial y^2} + 7u(x,y) = f(x,y), \ (x,y) \in \Omega, \tag{35}$$

$$u(x,y) = g(x,y), \ (x,y) \in \partial\Omega, \tag{36}$$

where $f(x,y)$ and $g(x,y)$ are given based on the following analytical solution

$$u(x,y) = e^{2x+3y}, \ (x,y) \in \overline{\Omega}.$$

In Table 2, we present results for five different irregular domains as shown in Figure 1. Since the area of these five domains are quite different, the maximum forcing terms and exact solutions have large discrepancy. Hence, it is more appropriate to use relative error to measure the accuracy of our computation. Overall, from Table 2, it appears that the smoothness of the boundary has little impact on the numerical accuracy. The order of polynomial remains to be a dominating factor on the performance of the proposed algorithm.

Table 2: Example 4: RMSE and relative error (Rel Err) for different computational domains.

Domains	n_i	n_b	RMSE	Rel Err	Polynomial Order
Amoeba	294	100	2.072e-06	6.974e-09	18
Cassini	313	100	6.430e-08	4.892e-09	17
Gear-Shaped	296	100	5.952e-05	1.562e-07	21
L-shaped	300	96	2.641e-10	6.443e-11	13
Corner-Shaped	330	90	8.867e-08	2.979e-09	13

Example 5. In this example, we consider the following fourth order boundary value problem from [15]:

$$
\begin{aligned}
(\Delta^2 - 100)u(x,y) &= f(x,y), \quad (x,y) \in \Omega, \qquad (37)\\
u(x,y) &= g(x,y), \quad (x,y) \in \partial\Omega, \qquad (38)\\
\Delta u(x,y) &= h(x,y), \quad (x,y) \in \partial\Omega, \qquad (39)
\end{aligned}
$$

where $f(x,y), g(x,y)$, and $h(x,y)$ are given based on the following analytical solution:

$$u(x,y) = \sin(\pi x)\cosh(y) - \cos(\pi x)\sinh(y), \ (x,y) \in \overline{\Omega}.$$

The computational domain is the Cassini (three) as shown in Figure 4.

In order to compare our results to those of [15], we choose the same number of collocation points in both approaches. To measure the numerical accuracy, we choose 230 random test points to compute the errors. In Table 3, we observe that the proposed algorithm is far more accurate than the results obtained in [15] where polyharmonic splines of order 3 were used. In this example, the proposed algorithm is also very effective for solving fourth order partial differential equations in an irregular domain.

Table 3: Example 5: Comparison of RMSE and RMSEx with polynomial basis functions and polyharmonic splines.

	Polyharmonic Splines ($r^6 \ln r$)		Polynomial Basis Functions		
(n_i, n_b)	RMSE	RMSEx	RMSE	RMSEx	order
(126,80)	2.440e-05	8.585e-05	8.382e-11	2.968e-10	22
(208,140)	5.887e-06	3.482e-05	5.441e-12	1.539e-11	23
(374,200)	1.643e-06	1.710e-05	3.605e-13	1.295e-12	23

6 Conclusions

In this work, polynomial particular solutions for the general linear differential operators with constant coefficients have been derived. This is a further improvement of the previous work [5, 8] where the differential operator contains no convective terms. Instead of confining the collocation points on the Gauss-Lobatto points in a rectangular domain, we are allowed to choose the collocation points in an arbitrary fashion using our proposed algorithm. A multiple scale technique is required to alleviate the ill-conditioning problem. Since the polynomial particular solution is available, the MPS can be easily employed to solve various types of elliptic partial differential equations. Another advantage of the proposed algorithm is that there is no parameter to be adjusted and the algorithm is very stable and highly accurate. Once the particular solutions are generated and stored in a table, the computation is very efficient.

With further study, it is possible to extend our proposed algorithm to solving partial differential equations with variable coefficients and 3D problems. The relaxation on the condition $a_6 \neq 0$ in Theorem 1 is also an open research topic. The current paper has opened up some more outstanding research topics that are currently under investigation by our research group.

Acknowledgements

The third author thanks the National Natural Science Foundation of Jiangsu Province (No. BK20150795) and the Fundamental Research Funds for the Central Universities (No. 2015B11914).

Appendix

The following MATLAB© code is based on Algorithm 1 to generate all the particular solutions corresponding to polynomial basis functions for the differential operator shown in Example 4. By symbolic computation, we save all these particular solutions in a table for the efficient computation of particular solutions in the MPS.

```
syms x y
order = 30; % highest order of polynomial basis
par=cell(order+1,order+1);
count=0;
for i=0:order
    for j=0:i
        p1=x.^(i-j).*y.^j;
```

```
        p=p1;
        sum1=0; coef=1;
        for k=1:i
           newp=2*diff(p,x,2)+3*diff(diff(p,x,1),y,1)+3*diff(p,y,2);
           coef=-coef/7;
           sum1=sum1+coef*newp;
           p=newp;
        end
         sum1=(1/7)*(sum1+p1);
        par{j+1,i+1}=inline(sum1,'x','y');
    end
end
save ('par_30.mat', 'par')
```

References

[1] K.E. Atkinson. The numerical evaluation of particular solutions for Poisson's equation. *IMA*, 5:319–338, 1985.

[2] W. Chen, Z.J. Fu, C.S. Chen. Recent Advances in Radial Basis Function Collocation Methods. Springer Berlin Heidelberg, 2014.

[3] C.S. Chen, C.M Fan, and P.H. Wen. The method of particular solutions for solving elliptic problems with variable coefficients. *The International Journal of Computational Methods*, 8:545–559, 2011.

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

[5] C.S. Chen, Sungwook Lee, and C.S. Huang. Derivation of particular solution using Chebyshev polynomial based functions. *International Journal of Computational Methods*, 4(1):15–32, 2007.

[6] A.H.-D. Cheng. Particular solutions of Laplacian, Helmholtz-type, and polyharmonic operators involving higher order radial basis functions. *Engineering Analysis with Boundary Elements*, 24:531–538, 2000.

[7] M.A. Golberg, C.S. Chen, and Y.F. Rashed. The annihilator method for computing particular solutions to partial differential equations. *Engineering Analysis with Boundary Elements*, 23:275–279, 1999.

[8] M.A. Golberg, A.S. Muleshkov, C.S. Chen, and A.H.-D. Cheng. Polynomial particular solutions for certain kind of partial differential operators. *Numerical Methods for Partial Differential Equations*, 19:112–133, 2003.

[9] C. Liu. A highly accurate MCTM for inverse Cauchy problems of Laplace equations in arbatrary plane domains. *CMES: Computer Modeling in Engineering & Sciences*, 35:91–111, 2008.

[10] C. Liu. A multiple-scale Trefftz method for an incomplete Cauchy problem of biharmonic equation. *Engineering Analysis with Bounddary Elements*, 37:1445–1456, 2013.

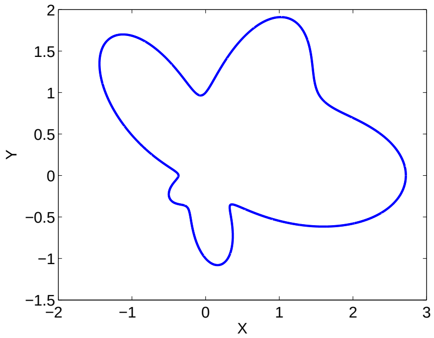
[11] A.S. Muleshkov, M.A. Golberg, and C.S. Chen. Particular solutions of Helmholtz-type operators using higher order polyharmonic splines. *Computational Mechanics*, 23:411–419, 1999.

[12] A.S. Muleshkov, Goldberg M.A., and Chen C. S. Particular solutions of the multi-Helmholtz-type equation. *Engineering Analysis with Bounddary Elements*, 31(7):624–30, 2007.

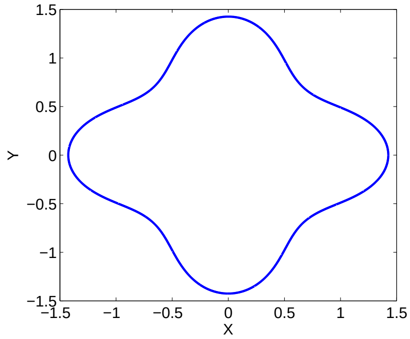
[13] C.C. Tsai. Particular solutions of Chebyshev polynomials for polyharmonic and poly-Helmholtz equations. *CMES: Computer Modeling in Engineering & Sciences*, 27(3):151–62, 2008.

[14] P.H. Wen and C.S. Chen. The method of particular solutions for solving scalar wave equations. *The International Journal for Numerical Methods in Biomedical Engineering*, 26:1878–1889, 2010.

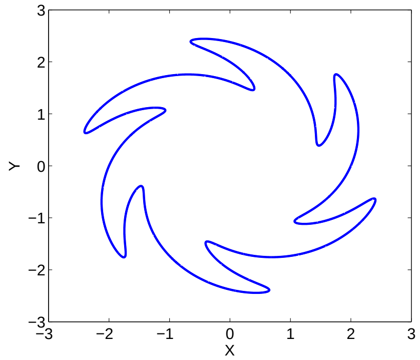
[15] G. Yao, C.S. Chen, and C.C. Tsai. A revisit on the derivation of the particular solution for the differential operator $\Delta^2 - \lambda^2$. *Advances in Applied Mathematics and Mechanics*, 1:750–768, 2009.



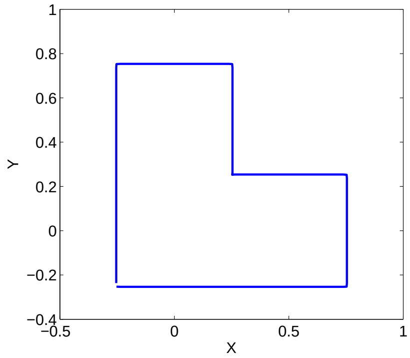
(a) Amoeba-like



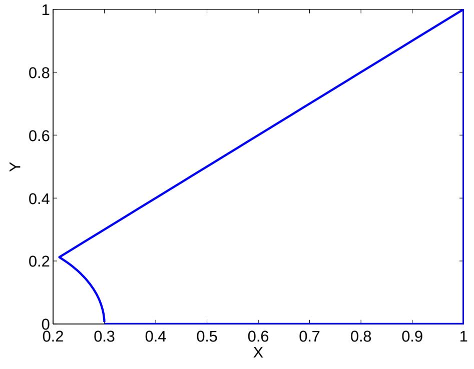
(b) Cassini-shaped



(c) Gear-shaped



(d) L-shaped



(e) Corner-shaped

Figure 1: The profiles of the computational domains.

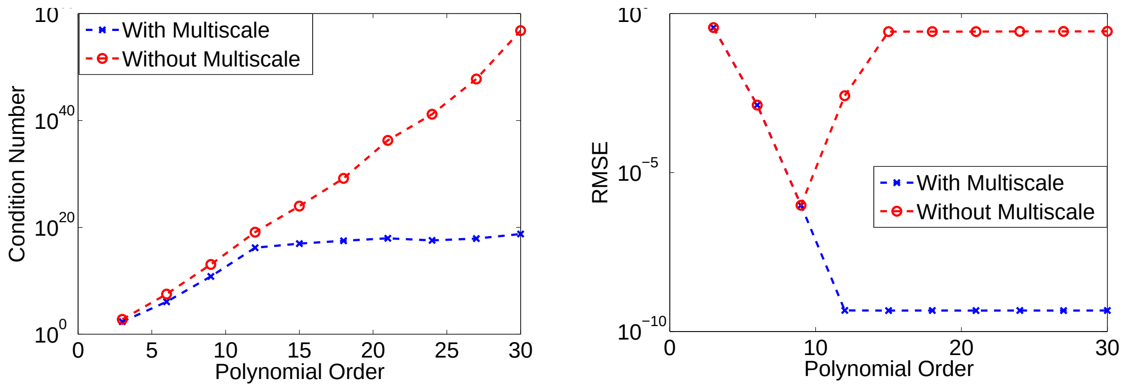
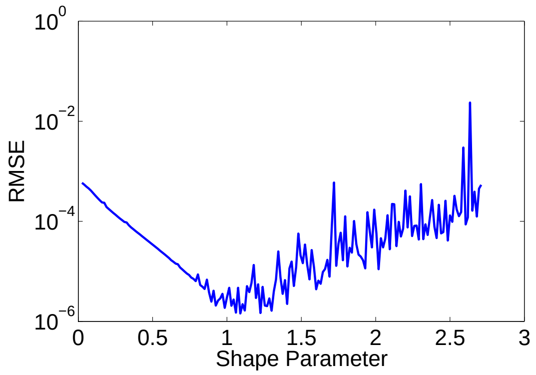
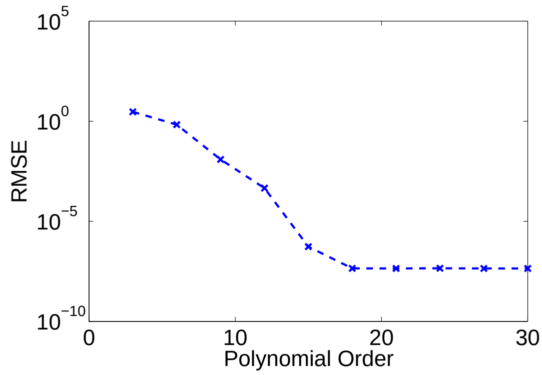


Figure 2: Example 1: The profiles of condition numbers and RMSE with and without using multiple scale technique.



(a) RBF



(b) Polynomial

Figure 3: Example 2: (a) RMSE versus the shape parameter using RBF. (b) RMSE versus the polynomial order using polynomial basis.

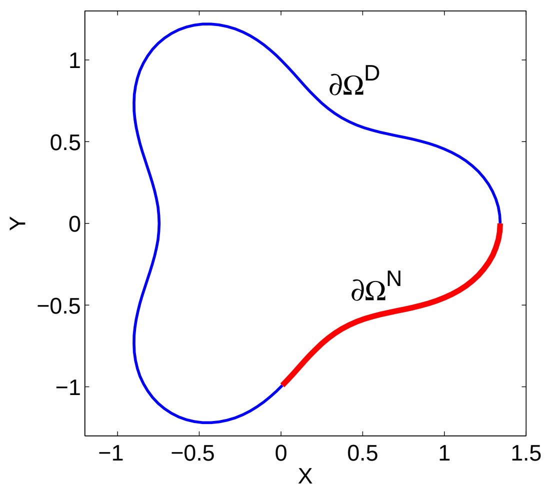
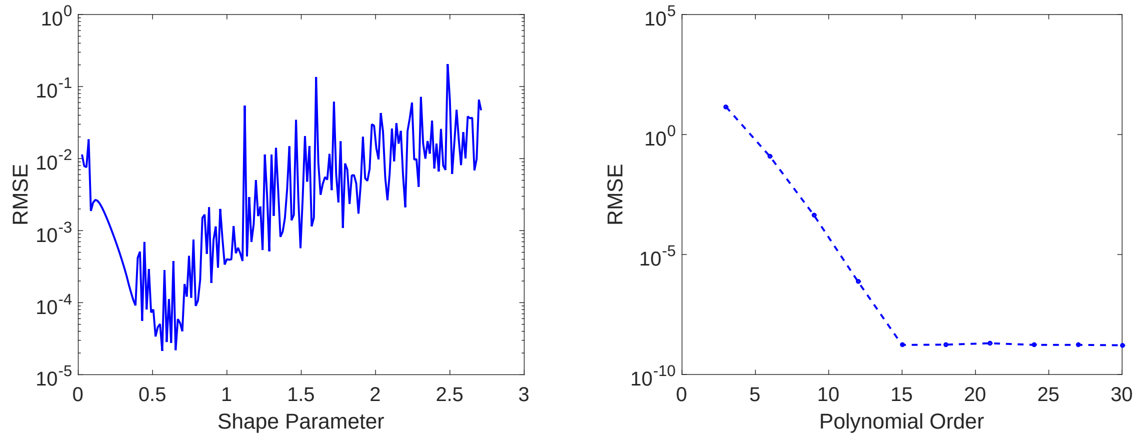


Figure 4: Example 3: The profile of the domain and its boundary conditions.



(a) RBF

(b) Polynomial

Figure 5: Example 3: (a) RMSE versus the shape parameter of the MQ. (b) RMSE versus the polynomial order.