

The Method of Fundamental Solutions for Solving Convection-Diffusion Equations with Variable Coefficients

C.M. Fan^{1,*}, C.S. Chen¹ and J. Monroe^{1,2}

¹ *Department of Mathematics, University of Southern Mississippi, Hattiesburg, MS 39406, USA.*

² *Department of Mathematics, Spring Hill College, Mobile, AL 36608, USA.*

Received 06 October 2008; Accepted (in revised version) 06 January 2009

Available online 17 March 2009

Abstract. A meshless method based on the method of fundamental solutions (MFS) is proposed to solve the time-dependent partial differential equations with variable coefficients. The proposed method combines the time discretization and the one-stage MFS for spatial discretization. In contrast to the traditional two-stage process, the one-stage MFS approach is capable of solving a broad spectrum of partial differential equations. The numerical implementation is simple since both closed-form approximate particular solution and fundamental solution are easy to find than the traditional approach. The numerical results show that the one-stage approach is robust and stable.

AMS subject classifications: 35J25, 65N35

Key words: Meshless method, method of fundamental solutions, particular solution, singular value decomposition, time-dependent partial differential equations.

1 Introduction

Through various types of reduction techniques, numerical solution of a given time-dependent partial differential equation can be obtained by converting it to a series of elliptic equations which can be solved by standard numerical methods. There are many reduction techniques which include Laplace transform method [4, 19], Fourier transform method [9], and discretization in time methods [7, 10, 11, 17]. Among these reduction techniques, the discretization in time methods appear to be the most popular approach. In this paper, we will focus on the method of discretization in time to

*Corresponding author.

URL: <http://www.math.usm.edu/cschen/>

Email: cs.chen@usm.edu (C.S. Chen), cmfan@ntou.edu.tw (C.M. Fan), monroe@shc.edu (J. Monroe)

reduce the given convection-diffusion problem to a series of elliptic partial differential equations.

For the purpose of solving time-dependent problems, instead of using traditional methods such as finite element, finite difference, or boundary element methods, we propose to apply the method of fundamental solutions (MFS) [2, 8, 10, 12] coupled with the method of particular solutions (MPS) with the use of radial basis functions. Such approach for solving time-dependent problems can be found in the literature [7, 10, 11, 17]. When the fundamental solution and particular solution of a given differential operator are available, the differential equation can be solved effectively. However, they can only be obtained for a limited class of linear differential operators. The fundamental solutions for various types of differential operators are available in the literature of boundary integral equations and boundary element methods (BEM). Furthermore, the closed-form particular solutions are available only for very limited classes of differential equations [5, 10, 14]. In the BEM, the dual reciprocity method (DRM) [15] has been successful in coupling the fundamental solution and particular solution to solve various science and engineering problems. However, for differential equations with variable coefficients, the above approach requires iterations and is not very effective. Recently, combining the MFS, MPS, and the DRM, it is possible to extend the above methods for solving elliptic partial differential equations with variable coefficients without the need of meshing the domain or boundary [3]. The idea of solving the given partial differential equation by combining the fundamental solution and particular solution as a one-stage method were proposed by Balakrishnan and Ramachandran [1] and Wang and Qin [18]. However, they seems unaware of the extended applications for solving PDEs with variable coefficients. The extensive study has been given and excellent results have been reported by Chen et al. [3] using the one-stage approach. Based on the numerical technique proposed in [3, 18], it is the purpose of this paper to extend the proposed one-stage method of the MFS and MPS to solve general convection-diffusion equations.

This paper is organized as follows. In Section 2, the θ -method has been applied to discretize the time domain. The given convection-diffusion equation is reduced to a series of elliptic differential equations. The MFS coupled with the MPS in the sense of one-stage formulation is applied to solve these elliptic equations at each time step. In Section 3, we conducted extensive numerical tests on two examples to demonstrate the convergence, stability, and high accuracy of the numerical algorithm mentioned in Section 2. In Section 4, we summarize the impact of each parameter to be used in the implementation.

2 Convection-diffusion equations

In this section, we consider the following general non-homogeneous time-dependent convection-diffusion equation in the closed domain $\Omega \subset \mathbb{R}^2$ bounded by $\partial\Omega$ given by

$$\frac{\partial u(\mathbf{x}, t)}{\partial t} = k\Delta u(\mathbf{x}, t) + (\mathbf{v} \cdot \nabla) u(\mathbf{x}, t) + f(\mathbf{x}, t), \quad \mathbf{x} \in \Omega, \quad (2.1)$$

$$\mathcal{B}u(\mathbf{x}, t) = g(\mathbf{x}, t), \quad \mathbf{x} \in \partial\Omega, \quad (2.2)$$

$$u(\mathbf{x}, 0) = h(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad (2.3)$$

where $\mathbf{v}=(v_x, v_y)$ is the convective velocity, k is the diffusivity, Δ is the Laplacian, and $f(\mathbf{x}, t), g(\mathbf{x}, t), h(\mathbf{x})$ are given functions. \mathcal{B} is the boundary operator and (2.2) is the given boundary condition.

The method of lines is used whereby time is discretized using the θ - method and the spatial variables are discretized using a combination of the MFS and MPS which will be discussed in the later sections.

2.1 A finite difference time stepping algorithm

In this section, a generalized trapezoidal method (θ -method) is used to approximate the time derivative in (2.1). Let $\delta t=t^{n+1} - t^n$ be the time step and define the mesh $t^n=n\delta t, n \geq 0$. For $t^n \leq t \leq t^{n+1}$, approximate $u(\mathbf{x}, t)$ by

$$u(\mathbf{x}, t) \simeq \theta u(\mathbf{x}, t^{n+1}) + (1 - \theta)u(\mathbf{x}, t^n), \quad (2.4)$$

and

$$\frac{\partial u(\mathbf{x}, t)}{\partial t} \simeq \frac{u(\mathbf{x}, t^{n+1}) - u(\mathbf{x}, t^n)}{\delta t}, \quad (2.5)$$

where $(0 \leq \theta \leq 1)$. From (2.4), we have

$$\Delta u(\mathbf{x}, t) = \theta \Delta u(\mathbf{x}, t^{n+1}) + (1 - \theta)\Delta u(\mathbf{x}, t^n). \quad (2.6)$$

For simplicity, we denote $u(\mathbf{x}, t^n) \equiv u^n(\mathbf{x})$ and $f(\mathbf{x}, t^n) \equiv f^n(\mathbf{x})$. Using (2.5) and (2.6), (2.1) can be reformulated as follows:

$$\begin{aligned} \frac{u^{n+1}(\mathbf{x}) - u^n(\mathbf{x})}{\delta t} &= \theta \left(k\Delta u^{n+1}(\mathbf{x}) + (\mathbf{v} \cdot \nabla) u^{n+1}(\mathbf{x}) + f^{n+1}(\mathbf{x}) \right) \\ &\quad + (1 - \theta) \left(k\Delta u^n(\mathbf{x}) + (\mathbf{v} \cdot \nabla) u^n(\mathbf{x}) + f^n(\mathbf{x}) \right). \end{aligned} \quad (2.7)$$

To avoid evaluating Δu^n in (2.7), we define the following half step approach [11]

$$v^{n+1} = u^{n+1} + \frac{1 - \theta}{\theta} u^n. \quad (2.8)$$

By some algebraic manipulations, (2.7) can be rewritten as

$$\Delta v^{n+1} + (\mathbf{v} \cdot \nabla) \frac{v^{n+1}}{k} - \frac{1}{k\theta\delta t} v^{n+1} = -\frac{1}{k\theta^2\delta t} u^n - \frac{f^{n+1}}{k} - \frac{1 - \theta}{\theta} \frac{f^n}{k}. \quad (2.9)$$

Once v^{n+1} is evaluated, u^{n+1} can be obtained through (2.8). Note that forward differencing ($\theta = 0$) cannot be used in this formulation. In particular, for implicit formulation ($\theta = 1$), we have

$$(k\Delta - \frac{1}{\delta t} + v_x \frac{\partial}{\partial x} + v_y \frac{\partial}{\partial y})u^{n+1}(\mathbf{x}) = -\frac{1}{\delta t}u^n(\mathbf{x}) - f^{n+1}(\mathbf{x}) \quad (2.10)$$

Using the formulation stated above, (2.1) has been transformed into a series of inhomogeneous convection-reaction equations. To avoid the discretization of domain, the MFS and the MPS have been employed. This approach results in a mesh free method.

2.2 The method of particular solutions (MPS)

In the past, the MPS has been widely employed to solve (2.9) without the convection term $(\mathbf{v} \cdot \nabla)v^{n+1}/k$ [4, 11, 17, 19]. In such cases, the closed-form approximate particular solutions of Helmholtz-type equations are available. However, with the general convection term, the derivation of a closed-form particular solution for (2.9) is not available. In this paper, the convection term is moved to the right-hand side of (2.9) and is treated as a forcing term. As a result, the iterative procedure is required. In this section, we employ the newly established numerical approach of keeping only the Laplacian term on the left-hand side and moving all other terms to the right-hand side; i.e., [3, 18]

$$\begin{aligned} \Delta v^{n+1} &= -(\mathbf{v} \cdot \nabla) \frac{v^{n+1}}{k} + \frac{1}{k\theta\delta t}v^{n+1} - \frac{1}{k\theta^2\delta t}u^n - \frac{1}{k}f^{n+1} - \frac{1-\theta}{k\theta}f^n \\ &= F(\mathbf{x}, u^n, v^{n+1}, (\mathbf{v} \cdot \nabla)v^{n+1}, f^n, f^{n+1}). \end{aligned} \quad (2.11)$$

The convection-reaction equation in (2.11) can be viewed as a non-linear Poisson equation. As far as the Poisson equation is concerned, the solution v^{n+1} in (2.11) can be written as the sum of the approximate particular solution, $\hat{v}_p^{n+1}(\mathbf{x})$, and the approximate homogeneous solution, $\hat{v}_h^{n+1}(\mathbf{x})$:

$$v^{n+1}(\mathbf{x}) \simeq \hat{v}_p^{n+1}(\mathbf{x}) + \hat{v}_h^{n+1}(\mathbf{x}). \quad (2.12)$$

The basic idea of the MPS is to approximate the inhomogeneous part in (2.11) using a linear combination of the RBF, $\phi(r)$, as follows:

$$F(\mathbf{x}, u^n, v^{n+1}, (\mathbf{v} \cdot \nabla)v^{n+1}, f^n, f^{n+1}) \simeq \sum_{j=1}^{np} a_j \phi(r_j), \quad (2.13)$$

where $r_j = \|\mathbf{x} - \mathbf{x}_j\|$ and $\{\mathbf{x}_j\}_{j=1}^{np}$ are the center points, which are located randomly inside the computational domain. $\phi: \mathbb{R}_+ \rightarrow \mathbb{R}$ is a univariate function. $\{a_j\}_{j=1}^{np}$ are the unknown coefficients to be determined.

Then, another RBF, $\Phi(r)$, should be derived analytically by satisfying the Poisson equation, in which the inhomogeneous term is the original RBF, $\phi(r)$,

$$\Delta\Phi(r) = \phi(r). \quad (2.14)$$

Once the $\Phi(r)$ is obtained analytically, the particular solution can be expressed as a linear combination of $\Phi(r)$

$$\hat{v}_p^{n+1}(\mathbf{x}) = \sum_{j=1}^{np} a_j \Phi(r_j). \quad (2.15)$$

For Poisson equation (2.14) the derivation of $\Phi(r)$ for a given $\phi(r)$ is straightforward [5, 10]. MQ, $\phi(r) = \sqrt{r^2 + c^2}$, is one of the most popular RBFs. The closed-form particular solution of MQ and its derivatives are as follows [13]:

$$\Phi(r) = \frac{1}{9} (4c^2 + r^2) \sqrt{r^2 + c^2} - \frac{c^3}{3} \ln(c + \sqrt{r^2 + c^2}), \quad (2.16)$$

$$\frac{\partial \Phi(r)}{\partial x} = \frac{x (c\sqrt{r^2 + c^2} + 2c^2 + r^2)}{3 (c + \sqrt{r^2 + c^2})}, \quad (2.17)$$

$$\frac{\partial \Phi(r)}{\partial y} = \frac{y (c\sqrt{r^2 + c^2} + 2c^2 + r^2)}{3 (c + \sqrt{r^2 + c^2})}, \quad (2.18)$$

where c is the shape parameter. In the RBFs literature, it is well-known that the quality of the interpolation depends on the proper choice of the shape parameter. Despite the effectiveness of the MQ, it is a challenge to find the optimal shape parameter. We will address this issue in the section with numerical results. Other RBFs can also be used and the derivation of their particular solutions is straightforward. In the section of numerical results, we will also use conical RBFs $\phi=r^{2n-1}$ for the evaluation of particular solutions. In this case, by direct integration, we can easily obtain

$$\Phi(r) = \frac{r^{2n+1}}{(2n+1)^2}. \quad (2.19)$$

Unlike the two-stage scheme [10], it is important to note that $\{a_j\}_{j=1}^{np}$ can not be obtained directly from (2.13) since F contains unknown functions. Combining the MPS and the MFS with some kind of reformulation as we shall see in the next subsection, we will be able to find $\{a_j\}_{j=1}^{np}$ so that the particular solution can be evaluated.

2.3 The method of fundamental solutions (MFS)

Since the fundamental solution satisfies the governing equation, the approximate homogeneous solution \hat{v}_h^{n+1} in (2.12) can be expressed as a combination of the fundamental solution with different strengths, $\{b_j\}_{j=1}^{nh}$,

$$\hat{v}_h^{n+1}(\mathbf{x}) = \sum_{j=1}^{nh} b_j G(\rho_j), \quad \mathbf{x} \in \Omega \cup \partial\Omega, \quad (2.20)$$

where $\rho_j = \|\mathbf{x} - \mathbf{y}_j\|$ is the distance between the field point \mathbf{x} and the source point, $\{\mathbf{y}_j\}_{j=1}^{nh}$, which are located outside the solution domain. In this paper, $G(\rho)$ is the fundamental solution of the Laplace equation. How to place the source points to obtain optimal solution is still an outstanding research problem. The typical distribution of source point and boundary point can be found in Fig. 1.

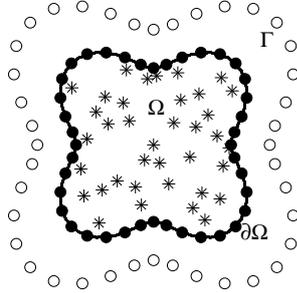


Figure 1: Interpolation points (*), boundary collocation points (•), and source points (o) on the fictitious boundary.

The two-dimensional fundamental solution and its derivatives of the Laplace equation are given by

$$G(\rho) = \ln \rho, \quad \frac{\partial G(\rho)}{\partial x} = \frac{x}{\rho^2}, \quad \frac{\partial G(\rho)}{\partial y} = \frac{y}{\rho^2}. \tag{2.21}$$

From (2.12), we have

$$\begin{aligned} v^{n+1}(\mathbf{x}) &\simeq \hat{v}^{n+1}(\mathbf{x}) = \hat{v}_p^{n+1}(\mathbf{x}) + \hat{v}_h^{n+1}(\mathbf{x}) \\ &= \sum_{j=1}^{np} a_j \Phi(r_j) + \sum_{j=1}^{nh} b_j G(\rho_j), \end{aligned} \tag{2.22}$$

$$\frac{\partial \hat{v}^{n+1}(\mathbf{x})}{\partial x} = \sum_{j=1}^{np} a_j \frac{\partial \Phi(r_j)}{\partial x} + \sum_{j=1}^{nh} b_j \frac{\partial G(\rho_j)}{\partial x}, \tag{2.23}$$

$$\frac{\partial \hat{v}^{n+1}(\mathbf{x})}{\partial y} = \sum_{j=1}^{np} a_j \frac{\partial \Phi(r_j)}{\partial y} + \sum_{j=1}^{nh} b_j \frac{\partial G(\rho_j)}{\partial y}. \tag{2.24}$$

Substituting (2.22)-(2.24) into (2.9), we obtain the following equation

$$\sum_{j=1}^{np} a_j \mathcal{L} \Phi(r_{ij}) + \sum_{j=1}^{nh} b_j \mathcal{L} G(\rho_{ij}) = F^{n+1}(\mathbf{x}_i), \quad i = 1, 2, \dots, np, \tag{2.25}$$

where

$$\mathcal{L} = \Delta + (\mathbf{v} \cdot \nabla) \frac{1}{k} - \frac{1}{k\theta\delta t}, \tag{2.26}$$

$$F^{n+1}(\mathbf{x}_i) = -\frac{1}{k\theta^2\delta t} u^n(\mathbf{x}_i) - \frac{1}{k} f^{n+1}(\mathbf{x}_i) - \frac{1-\theta}{k\theta} f^n(\mathbf{x}_i). \tag{2.27}$$

We notice that $\Delta G(\rho_{ij})=0$ and $\Delta\Phi(r_{ij})=\phi(r_{ij})$. Then, (2.25) becomes

$$\sum_{j=1}^{np} a_j \Psi(r_{ij}) + \sum_{j=1}^{nh} b_j \Theta(\rho_{ij}) = F^{n+1}(\mathbf{x}_i), \quad i = 1, 2, \dots, np, \tag{2.28}$$

where

$$\Psi(r_{ij}) = \phi(r_{ij}) + \frac{1}{k}(\mathbf{v} \cdot \nabla)\Phi(r_{ij}) - \frac{1}{k\theta\delta t}\Phi(r_{ij}), \tag{2.29}$$

$$\Theta(\rho_{ij}) = \frac{1}{k}(\mathbf{v} \cdot \nabla)G(\rho_{ij}) - \frac{1}{k\theta\delta t}G(\rho_{ij}). \tag{2.30}$$

Hence, the evaluation of second order derivatives with respect to x and y can be avoided during the proposed numerical process.

The boundary condition in (2.2) becomes

$$\sum_{j=1}^{np} a_j \mathcal{B}\Phi(r_{ij}) + \sum_{j=1}^{nh} b_j \mathcal{B}G(\rho_{ij}) = g^{n+1}(\mathbf{x}_i), \quad i = np + 1, np + 2, \dots, np + nh. \tag{2.31}$$

From (2.28) and (2.31), we have thus formulated the following system of equations of order $(np + nh) \times (np + nh)$,

$$\begin{bmatrix} \Psi(r_{ij}) & \Theta(\rho_{ij}) \\ \mathcal{B}\Phi(r_{ij}) & \mathcal{B}G(\rho_{ij}) \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} = \begin{bmatrix} F^{n+1}(\mathbf{x}_i) \\ g^{n+1}(\mathbf{x}_i) \end{bmatrix}, \tag{2.32}$$

where $\mathbf{a} = [a_1, a_2, a_3, \dots, a_{np}]^T$ and $\mathbf{b} = [b_1, b_2, b_3, \dots, b_{nh}]^T$.

The above matrix system is solved by singular value decomposition (SVD) with truncation [3,6,16] to stabilize the solution in our following numerical tests. The small singular values will be truncated in the solution process. If the convective velocity is a time-independent function, the inverse matrix, which is produced by SVD, can be used repeatedly in every time step. That means we only have to invert the matrix system once for all time steps.

3 Numerical results

To demonstrate the efficiency and accuracy of the numerical method mentioned in the previous sections, we give two examples. The first example is a standard diffusion equation and the second one is a convection-diffusion equation with nonconstant coefficients. All of the numerical results were compared with the analytical solutions and, for the sake of completeness, we numerically examined several factors that may affect the performance of the proposed method.

For systematically locating the source points in the MFS for all of the following numerical tests, we use the following formula:

$$\mathbf{x}^s = \mathbf{x}^b + \sigma (\mathbf{x}^b - \mathbf{x}^c), \tag{3.1}$$

where \mathbf{x}^s , \mathbf{x}^b , and \mathbf{x}^c denote the source, boundary, and central nodes, respectively. σ , which is a pre-defined parameter, determines the distance between source and the boundary, $\partial\Omega$. The relation between σ and the accuracy of the results will be shown in the numerical results.

In addition, the root-mean-square error (RMSE) and the root-mean-square error of the derivative with respect to x (RMSE x) are used in the paper to show the accuracy of the solutions. They are defined as follows:

$$RMSE = \sqrt{\frac{1}{n_1} \sum_{j=1}^{n_1} (\hat{u}_j - u_j)^2}, \quad (3.2)$$

$$RMSEx = \sqrt{\frac{1}{n_1} \sum_{j=1}^{n_1} \left(\frac{\partial \hat{u}_j}{\partial x} - \frac{\partial u_j}{\partial x} \right)^2}, \quad (3.3)$$

where n_1 is the number of testing nodes chosen randomly within the domain. \hat{u}_j denotes the approximate solution at the j^{th} node. From the numerical results, we found that the RMSE y is similar to RMSE x . In order to avoid duplication, we only show the RMSE and RMSE x in the following tests.

When MQ is chosen as the basis function for interpolation, it is well-known that finding the optimal shape parameter c is a challenge. In [3], it is known that the given boundary conditions can be used to find the optimal shape parameter since the boundary conditions are known and can be regarded as part of the analytical solution. Hence, we repeat the first time step using different shape parameters and record the RMSE along the boundary, which is denoted as RMSE b . The definition of RMSE b is defined as follows:

$$RMSEb = \sqrt{\frac{1}{n_2} \sum_{j=1}^{n_2} (\hat{u}_j - u_j)^2}, \quad (3.4)$$

where n_2 is the number of testing nodes chosen along the boundary. The set of testing nodes for RMSE b should not coincide with the set of boundary collocation nodes in the MFS.

In the following examples, we denote γ as the truncated singular value, np as the number of interior points, nh as the number of boundary points and n_1, n_2 as the number of testing points showed in (3.2)-(3.4). Due to the ill-conditioning of the MFS, TSVD is required for solving (2.32) [3]. For the time stepping algorithm, we choose $\theta = 1$; i.e., implicit Euler method.

Example 3.1. Consider the following diffusion equation with boundary and initial conditions in a peanut shaped domain:

$$\frac{\partial u(\mathbf{x}, t)}{\partial t} = \Delta u(\mathbf{x}, t) + \sin(x) \sin(y) (2 \cos(t) - \sin(t)), \quad \mathbf{x} \in \Omega, \quad t > 0, \quad (3.5)$$

$$u(\mathbf{x}, t) = \sin(x) \sin(y) \cos(t), \quad \mathbf{x} \in \partial\Omega, \quad t > 0, \quad (3.6)$$

$$u(\mathbf{x}, 0) = \sin(x) \sin(y), \quad \mathbf{x} \in \Omega. \tag{3.7}$$

The analytical solution is given as follows:

$$u(\mathbf{x}, t) = \sin(x) \sin(y) \cos(t). \tag{3.8}$$

The computational domain, interior nodes, and boundary nodes are shown in Fig. 2. Unless otherwise specified, the numerical results in figures and tables for this example are based on the following parameters: $np=74, nh=30, n_1=145, n_2=50, \sigma=5, \delta t=0.1$ and $\gamma=10^{-9}$.

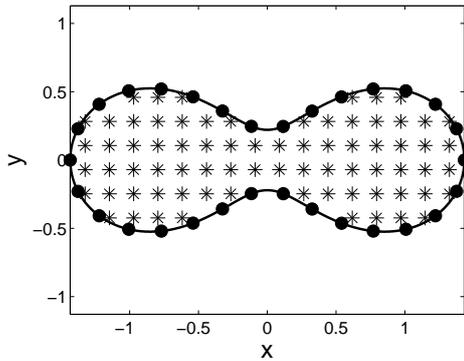


Figure 2: Computational domain, interpolation nodes (*) and boundary nodes (●) adopted in Example 1.

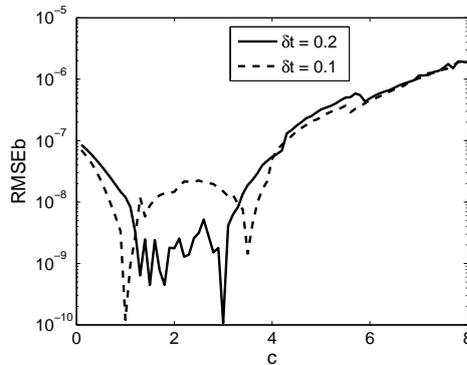


Figure 3: Profile of RMSEb versus shape parameter c for $\delta t = 0.1, 0.2$.

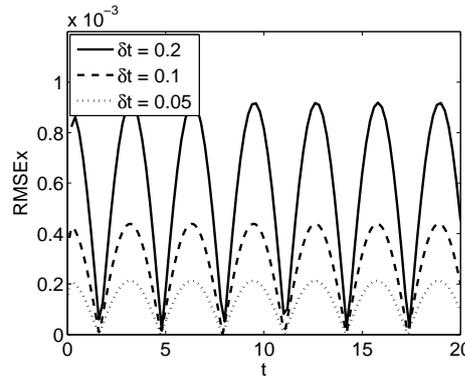
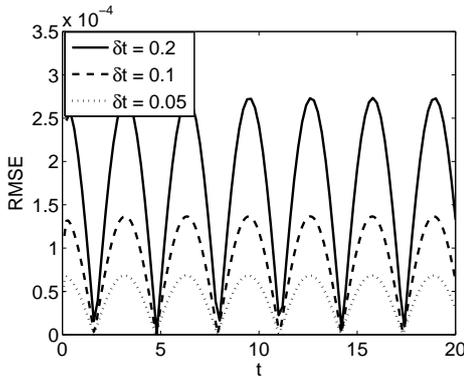


Figure 4: Profiles of RMSE (left) and RMSEx (right).

In order to identify the optimal shape parameter of MQ, we repeatedly tested the method in the first time step using the boundary conditions for various shape parameters. The resultant RMSEb and its corresponding shape parameter are presented in Fig. 3 for $\delta t=0.2$ and 0.1 .

The selection of optimal shape parameter of MQ is based on the smallest RMSEb of the initial time step. In addition, the RMSE and RMSEx from $t=0$ to $t=20$ are shown in Fig. 4 and Table 1 using various time steps $\delta t=0.2, 0.1, 0.05$. In Fig. 4, we observe

Table 1: RMSE and RMSE_x obtained by different time steps, δt , for $t = 1, 5, 20$.

	$\delta t = 0.2$	$\delta t = 0.1$	$\delta t = 0.05$
$t = 1$			
RMSE	1.67E-04	7.98E-05	3.89E-05
RMSE _x	5.65E-04	2.55E-04	1.21E-04
$t = 5$			
RMSE	5.46E-05	3.17E-05	1.69E-05
RMSE _x	1.79E-04	1.03E-04	5.33E-05
$t = 20$			
RMSE	1.32E-04	6.22E-05	3.01E-05
RMSE _x	4.50E-04	1.98E-04	9.37E-05

Table 2: RMSE and RMSE_x obtained using various number of nodes, np and nh , for $t = 1, 5, 20$.

	$np = 52$ $nh = 20$	$np = 74$ $nh = 30$	$np = 101$ $nh = 40$
$t = 1$			
RMSE	1.16E-04	7.98E-05	7.98E-05
RMSE _x	7.34E-04	2.55E-04	2.50E-04
$t = 5$			
RMSE	4.92E-05	3.17E-05	3.17E-05
RMSE _x	3.31E-04	1.03E-04	1.00E-04
$t = 20$			
RMSE	9.05E-05	6.22E-05	6.22E-05
RMSE _x	5.67E-04	1.98E-04	1.95E-04

the error for $\delta t=0.2$ is approximately twice as large as the error of $\delta t=0.1$ and the same situation is also observed for $\delta t=0.1$ and 0.05 . The numerical results are consistent with the discretized scheme in time we adopted.

We further investigated the effects of the total number of nodes, source location in the MFS, truncated singular value, and different types of RBFs. The RMSE and RMSE_x, obtained by using different numbers of nodes, are presented in Table 2 where we used 52, 74, and 101 interior nodes and 20, 30, and 40 boundary nodes. Excellent results are obtained using only 52 interior nodes and 20 boundary nodes. We also observed that the difference between the tests using $np=74, nh=30$ and $np=101, nh=40$ is negligible. This implies that the method converges rapidly and few nodes are required to achieve good accuracy.

It is known that the location of sources in the MFS formulation affects its performance. In Table 3, we observed little difference for various values of σ . This implies that the solution of the one-stage approach is very stable in terms of the location of sources [3].

In Table 4, we observed little difference in RMSE and RMSE_x using the cut-off singular values $\gamma=10^{-7}, 10^{-9}$, and 10^{-11} . The TSVD cannot regularize the matrix if γ is too small. The numerical simulation diverges in this example for $\gamma < 10^{-13}$.

Besides MQ, we also used the conical RBFs, r^{2n-1} . The numerical results are shown

Table 3: RMSE and RMSEx obtained by different location of sources, σ , for $t = 1, 5$ and 20 .

	$\sigma = 3$	$\sigma = 5$	$\sigma = 10$	$\sigma = 20$
$t = 1$				
RMSE	8.00E-05	7.98E-05	7.97E-05	8.00E-05
RMSEx	2.98E-04	2.55E-04	2.49E-04	2.45E-04
$t = 5$				
RMSE	3.18E-05	3.17E-05	3.17E-05	3.18E-05
RMSEx	1.24E-04	1.03E-04	9.98E-05	9.85E-05
$t = 20$				
RMSE	6.25E-05	6.22E-05	6.22E-05	6.24E-05
RMSEx	2.31E-04	1.98E-04	1.94E-04	1.91E-04

Table 4: RMSE and RMSEx obtained by different truncated singular values, γ , for $t = 1, 5$ and 20 .

γ	1.0E-07	1.0E-09	1.0E-11	1.0E-13
$t = 1$				
RMSE	8.65E-05	7.98E-05	7.98E-05	1.79E-04
RMSEx	1.88E-04	2.55E-04	2.48E-04	2.37E-03
$t = 5$				
RMSE	3.56E-05	3.17E-05	3.17E-05	8.78E-05
RMSEx	6.97E-05	1.03E-04	9.98E-05	1.19E-03
$t = 20$				
RMSE	6.73E-05	6.22E-05	6.22E-05	1.37E-04
RMSEx	1.48E-04	1.98E-04	1.93E-04	1.80E-03

in Table 5. From the table, we find both RBFs can reach similar accuracy. This implies that the challenge of choosing the optimal shape parameter can be alleviated using conical RBFs.

Example 3.2. We consider the convection-diffusion equation with its boundary and initial conditions as follows:

$$\frac{\partial u(\mathbf{x}, t)}{\partial t} = \Delta u(\mathbf{x}, t) - \cos(y) \frac{\partial u(\mathbf{x}, t)}{\partial x} - x \sin(x) \frac{\partial u(\mathbf{x}, t)}{\partial y} + f(\mathbf{x}, t), \quad \mathbf{x} \in \Omega, \quad (3.9)$$

$$u(\mathbf{x}, t) = (y \sin(\pi x) + x \cos(\pi y)) \cos(t), \quad \mathbf{x} \in \partial\Omega, \quad (3.10)$$

$$u(\mathbf{x}, 0) = y \sin(\pi x) + x \cos(\pi y), \quad \mathbf{x} \in \Omega. \quad (3.11)$$

The analytical solution is given by

$$u(\mathbf{x}, t) = (y \sin(\pi x) + x \cos(\pi y)) \cos(t). \quad (3.12)$$

The inhomogeneous term, $f(\mathbf{x}, t)$, can be chosen according to the analytical solution.

The computational domain, interior interpolation nodes, and boundary nodes are presented in Fig. 5. Unless otherwise specified, the numerical results in figures and tables for this example are based on the following parameters: $np=111, nh=60, n_1=75, n_2=40, \sigma=5, \delta t=0.1$ and $\gamma=10^{-9}$.

Table 5: RMSE and RMSE_x obtained by different radial basis function, $\phi(r)$, for $t = 1, 5$, and 20.

	$\sqrt{r^2 + c^2}$	r^5	r^7	r^9
$t = 1$				
RMSE	7.98E-05	8.01E-05	8.00E-05	7.98E-05
RMSE _x	2.55E-04	2.73E-04	2.64E-04	2.50E-04
$t = 5$				
RMSE	3.17E-05	3.19E-05	3.18E-05	3.17E-05
RMSE _x	1.03E-04	1.12E-04	1.07E-04	1.00E-04
$t = 20$				
RMSE	6.22E-05	6.25E-05	6.24E-05	6.23E-05
RMSE _x	1.98E-04	2.13E-04	2.05E-04	1.95E-04

Table 6: RMSE and RMSE_x obtained by different time increments, δt , for $t = 1, 5$ and 20.

	$\delta t = 0.2$	$\delta t = 0.1$	$\delta t = 0.05$	$\delta t = 0.01$
$t = 1$				
RMSE	1.69E-03	8.08E-04	4.02E-04	9.38E-05
RMSE _x	4.74E-03	2.13E-03	1.00E-03	4.46E-04
$t = 5$				
RMSE	4.29E-04	2.49E-04	1.38E-04	3.38E-05
RMSE _x	1.42E-03	7.02E-04	3.58E-04	2.29E-04
$t = 20$				
RMSE	1.37E-03	6.46E-04	3.18E-04	7.43E-05
RMSE _x	3.80E-03	1.69E-03	7.91E-04	3.391E-04

To choose the optimal shape parameter, the RMSE_b is calculated and recorded in the first time step from $c=0.1$ to $c=5$. The results are shown in Fig. 6 for $\delta t=0.1, 0.05$. We used the same technique as the last example to choose the optimal shape parameter in all of the following tests.

The profiles of RMSE and RMSE_x with respect to time are shown in Fig. 7 for $\delta t=0.2, 0.1$, and 0.05. From this figure, we find the error of $\delta t=0.2$ is twice the error of

Table 7: RMSE and RMSE_x obtained by different number of nodes, np and nh , for $t = 1, 5$ and 20.

	$np = 60$ $nh = 30$	$np = 111$ $nh = 60$	$np = 144$ $nh = 80$
$t = 1$			
RMSE	8.49E-04	8.08E-04	8.01E-04
RMSE _x	2.14E-03	2.13E-03	2.14E-03
$t = 5$			
RMSE	2.65E-04	2.49E-04	2.47E-04
RMSE _x	7.14E-04	7.02E-04	7.13E-04
$t = 20$			
RMSE	6.78E-04	6.46E-04	6.41E-04
RMSE _x	1.70E-03	1.69E-03	1.70E-03

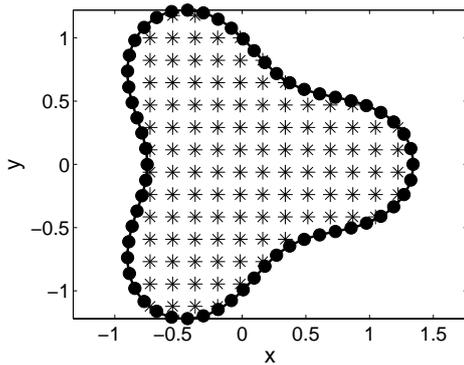


Figure 5: Computational domain, interpolation nodes (*) and boundary nodes (●) adopted in Example 2.

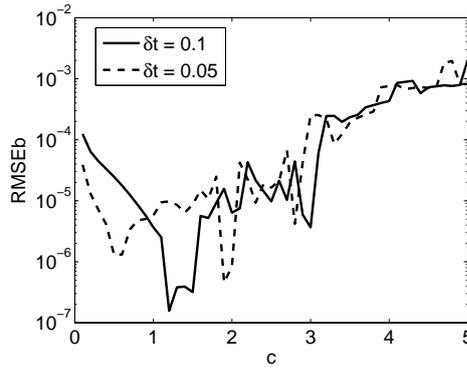


Figure 6: Profiles of RMSEb with respect to different shape parameters c for $\delta t = 0.05$ and $\delta t = 0.1$.

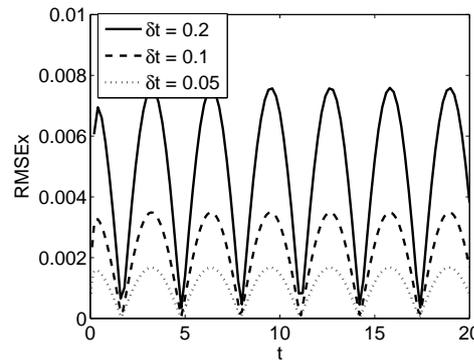
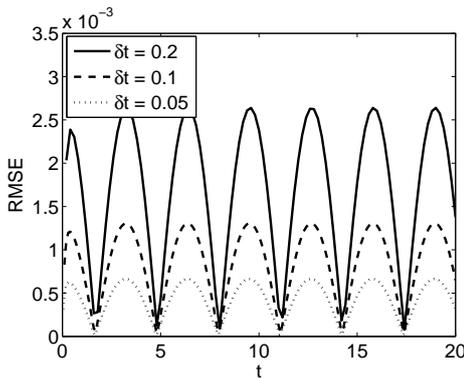


Figure 7: Profiles of RMSE (left) and RMSEx (right) from $t = 0$ to $t = 20$.

$\delta t=0.1$ and the error of $\delta t=0.1$ is twice the error of $\delta t=0.05$.

Similar to the first example, we numerically investigated the effect of time steps, number of nodes, and type of RBF in Tables 6-8. Since the location of source points in the MFS and the truncated singular value did not affect our numerical results very much in this example, we did not show it here. In Table 6, the RMSE and RMSEx are tabulated for $\delta t=0.2, 0.1, 0.05, 0.01$. Better RMSE and RMSEx are obtained by using the smaller time step which validates the stability of the proposed numerical scheme for time-dependent convection-diffusion equation.

We further tested the numerical simulation by using different numbers of nodes in Table 7. 60, 111, and 144 interpolation nodes as well as 30, 60, and 80 boundary nodes are considered in the table. It can be found that the accuracy is almost the same when interior and boundary nodes are larger than 111 and 60 respectively. The proposed scheme converges very fast when a relatively small number of nodes are adopted. Furthermore, the results in Table 8 are obtained by using different RBFs for interpolating the inhomogeneous term. We used MQ, r^5 , r^7 , and r^9 for the simulations. Only slight differences between the results can be observed in this table. Therefore, we can use different RBFs to avoid finding the optimal shape parameter of MQ for solving

Table 8: RMSE and RMSEx obtained by different radial basis function, $\phi(r)$, for $t = 1, 5$ and 20 .

	$\sqrt{r^2 + c^2}$	r^5	r^7	r^9
$t = 1$				
RMSE	8.08E-04	8.24E-04	8.27E-04	8.06E-04
RMSEx	2.13E-03	2.01E-03	2.06E-03	2.12E-03
$t = 5$				
RMSE	2.49E-04	2.54E-04	2.57E-04	2.49E-04
RMSEx	7.02E-04	6.48E-04	6.65E-04	6.99E-04
$t = 20$				
RMSE	6.46E-04	6.59E-04	6.60E-04	6.44E-04
RMSEx	1.69E-03	1.60E-03	1.64E-03	1.69E-03

time-dependent convection-diffusion equations.

4 Concluding remarks

Based on the idea of the one-stage scheme developed in [3], convection-diffusion partial differential equations with variable coefficients can be successfully solved with high accuracy using a time stepping algorithm. As a result, the MFS has been further extended to solving time-dependent problems with variable coefficients. In the past, a similar approach using the MFS and MPS could only deal with time-dependent problems with constant coefficients [7, 11]. In that sense, the MFS has made significant progress toward solving a broader class of partial differential equations. In the current one-stage scheme, due to the new formulation, the derivation of closed-form particular solutions only involve Laplacian and thus are easier to achieve. Similarly, we only require $\ln \rho$ as the fundamental solution in the new formulated differential equation. The tested examples show the numerical results are very stable and accurate. However, due to the ill-conditioning of the MFS, a regularization technique such as TSVD is required to ensure its stability.

There are several parameters that may affect the accuracy of the solution. Based on our extensive numerical tests, we conclude

1. The location of source points in the MFS does not have a major impact on the solution.
2. The singular value in the TSVD has a good range of cut off values.
3. Apparently, there is not much of a difference in accuracy between using MQ or conical RBFs, r^{2n-1} . This alleviates the difficulty of choosing the optimal shape parameter of MQ.
4. A small number of interpolation and boundary points are sufficient to produce accurate results. This implies that the solution converges rapidly.
5. The length of the time step affects the accuracy.

The mentioned approach is meshless and is very easy to implement. The method is especially attractive for three-dimensional problems. With the excellent numerical

results presented in this paper, we believe the method can be further applied to more challenging science and engineering problems.

Acknowledgements

The second author acknowledges the support of NATO Collaborative Linkage Grant under reference ESP.CLG.982891.

References

- [1] K. BALAKRISHNAN AND P. A. RAMACHANDRAN, *Osculatory interpolation in the method of fundamental solution for nonlinear Poisson problems*, Journal of Computational Physics, 172 (2001), pp. 1-18.
- [2] A. BOGOMOLNY, *Fundamental solutions method for elliptic boundary value problems*, SIAM Journal on Numerical Analysis, 22 (1985), pp. 644-669.
- [3] C. S. CHEN, A. KARAGEORGHIS AND Y. S. SMYRLIS, *The method of fundamental solutions for solving elliptic partial differential equations with variable coefficients*, in The Method of Fundamental Solutions — A Meshless Method, ed. Smyrlis, Y. S., Chen, C. S. and A. Karageorghis, Dynamics Publisher, (2008), pp. 75-105.
- [4] C. S. CHEN, M. A. GOLBERG AND Y. F. RASHED, . *A mesh free method for linear diffusion equations*, Numerical Heat Transfer, Part B, 33 (1998), pp. 469-486.
- [5] A. H. D. CHENG, *Particular solution of Laplacian, Helmholtz-type, and polyharmonic operators involving higher order radial basis functions*, Engineering Analysis with Boundary Elements, 24 (2000), pp. 531-538.
- [6] H. A. CHO, C. S. CHEN AND M. A. GOLBERG *Some comments on mitigating the ill-conditioning of the method of fundamental solutions*, Engineering Analysis with Boundary Elements, 30 (2006), pp. 405-410.
- [7] H. A. CHO, M. A. GOLBERG, A. S. MULESHKOV AND X. LI, *Trefftz methods for time dependent partial differential equations*, Computers, Materials & Continua, 1 (2004), pp. 1-37.
- [8] G. FAIRWEATHER AND A. KARAGEORGHIS, *The method of fundamental solutions for elliptic boundary value problems*, Advances in Computational Mathematics, 9 (1998), pp. 69-95.
- [9] E. V. GLUSHKOV, N. V. GLUSHKOVA AND C. S. CHEN, *Semi-analytical solution to heat-transfer problems using Fourier transform technique, radial basis functions and the method of fundamental solutions*, Numerical Heat Transfer, Part B, 52 (2007), pp. 409-427.
- [10] M. A. GOLBERG AND C. S. CHEN, *The method of fundamental solutions for potential, Helmholtz and diffusion problems*, in Boundary Integral Methods - Numerical and Mathematical Aspects, ed. M.A. Golberg, Computational Mechanics Publications, (1998), pp. 103-176.
- [11] M. INGBER, C. S. CHEN AND J. A. TANSKI, *A mesh free approach using radial basis functions and parallel domain decomposition for solving three dimensional diffusion equations*, International Journal for Numerical Methods in Engineering, 60 (2004), pp. 2183-2201.
- [12] V. D. KUPRADZE AND M. A. ALEKSIDZE, *The method of functional equations for the approximate solution of certain boundary value problems*, U.S.S.R. Computational Mathematics and Mathematical Physics, 4 (1964), pp. 82-126.

- [13] J. LI, Y. C. HON AND C. S. CHEN, *Numerical comparisons of two meshless methods using radial basis functions*, *Engineering Analysis with Boundary Elements*, 26 (2002), pp. 205-225.
- [14] A. S. MULESHKOV, M. A. GOLBERG AND C. S. CHEN, *Particular solutions of Helmholtz-type operators using higher order polyharmonic splines*, *Computational Mechanics*, 23 (1999), pp. 411-419.
- [15] P. W. PARTRIDGE, C. A. BREBBIA AND L. C. WROBEL, *The Dual Reciprocity Boundary Element Method*. Computational Mechanics Publications, London, (1992).
- [16] P. A. RAMACHANDRAN, *Method of fundamental solutions: singular value decomposition analysis*, *Communications in Numerical Methods in Engineering*, 18 (2002), pp. 789-801.
- [17] 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*, *International Journal for Numerical Methods in Engineering*, 74 (2008), pp. 1621-1644.
- [18] H. WANG AND Q. H. QIN, *A meshless method for generalized linear or nonlinear Poisson-type problems*, *Engineering Analysis with Boundary Elements*, 30 (2006), pp. 515-521.
- [19] S. ZHU, P. SATRAVAHA AND X. LU, *Solving linear diffusion equations with the dual reciprocity method in Laplace space*, *Engineering Analysis with Boundary Elements*, 13 (1995), pp. 1-10.