Commun. Comput. Phys.
doi: 10.4208/cicp.OA-2018-0005

Vol. **26**, No. 1, pp. 233-264
July 2019

A Novel Method for Solving Time-Dependent 2D Advection-Diffusion-Reaction Equations to Model Transfer in Nonlinear Anisotropic Media

Ji Lin[1], Sergiy Reutskiy[1,2], C. S. Chen[3] and Jun Lu[4,5,*]

[1] *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.*
[2] *State Institution "Institute of Technical Problems of Magnetism of the National Academy of Sciences of Ukraine", Industrialnaya St.,19, 61106, Kharkov, Ukraine.*
[3] *Department of Mathematics, University of Southern Mississippi, Hattiesburg, MS 39406, USA.*
[4] *Nanjing Hydraulic Research Institute, Nanjing 210029, China.*
[5] *State Key Laboratory of Hydrology Water Resources and Hydraulic Engineering, Nanjing 210098, China.*

Received 9 January 2018; Accepted (in revised version) 27 July 2018

Abstract. This paper presents a new numerical technique for solving initial and boundary value problems with unsteady strongly nonlinear advection diffusion reaction (ADR) equations. The method is based on the use of the radial basis functions (RBF) for the approximation space of the solution. The Crank-Nicolson scheme is used for approximation in time. This results in a sequence of stationary nonlinear ADR equations. The equations are solved sequentially at each time step using the proposed semi-analytical technique based on the RBFs. The approximate solution is sought in the form of the analytical expansion over basis functions and contains free parameters. The basis functions are constructed in such a way that the expansion satisfies the boundary conditions of the problem for any choice of the free parameters. The free parameters are determined by substitution of the expansion in the equation and collocation in the solution domain. In the case of a nonlinear equation, we use the well-known procedure of quasilinearization. This transforms the original equation into a sequence of the linear ones on each time layer. The numerical examples confirm the high accuracy and robustness of the proposed numerical scheme.

AMS subject classifications: 65N35, 65N40, 65Y20

Key words: Advection diffusion reaction, time-dependent, fully nonlinear, anisotropic media, Crank-Nicolson scheme, meshless method.

*Corresponding author. *Email addresses:* `linji861103@126.com`, `linji@hhu.edu.cn` (J. Lin), `sergiy.reutskiy2016@yandex.ru` (S. Reutskiy), `cschen.math@gmail.com` (C. S. Chen), `lujunhhu1981@126.com` (J. Lu)

1 Introduction

The governing equation of a variety of physical problems in engineering and science is expressed by the advection-diffusion-reaction (ADR) equation. The ADR equation is a second order parabolic partial differential equation (PDE). In this paper we consider the ADR equation in the form:

$$\frac{\partial C(\mathbf{x},t)}{\partial t}=div[\boldsymbol{Q}(\mathbf{x},t,C)]+\nabla\cdot(\mathbf{a}(\mathbf{x},t)C)+R\left(\mathbf{x},C,C_x,C_y,t\right)=0,\quad \mathbf{x}=(x_1,x_2)\in\Omega, \tag{1.1}$$

where $C(\mathbf{x},t)$ is the variable of interest (such as the concentration of pollutant for mass transfer and the temperature for heat transfer etc). The diffusion term $div[\boldsymbol{Q}(\mathbf{x},t,C)]$ describes the micro transport of $C(\mathbf{x},t)$ due to its gradients. Here $\boldsymbol{Q}(\mathbf{x},t,C)$ is the flux vector of $C(\mathbf{x},t)$

$$\boldsymbol{Q}(\mathbf{x},t,C)=\widehat{D}(\mathbf{x},C,t)\nabla C(\mathbf{x},t). \tag{1.2}$$

In the general case of anisotropic media, the diffusivity $\widehat{D}$ is the second order tensor which can be represented as a symmetric matrix whose entries are bounded functions:

$$\widehat{D}(\mathbf{x},C,t)=\begin{pmatrix} D_{11}(\mathbf{x},C,t), D_{12}(\mathbf{x},C,t)\\ D_{21}(\mathbf{x},C,t), D_{22}(\mathbf{x},C,t)\end{pmatrix}, \tag{1.3}$$

where $D_{21}=D_{12}$, $D_{11}D_{22}>D_{12}D_{21}$ from Onsagar's reciprocity relation which provides the elliptic type of the differential operator in the right hand side of the equation. The advection term $\nabla\cdot(\mathbf{a}(\mathbf{x},t)C)$ describes the macro transfer of the quantities, where $\mathbf{a}(\mathbf{x},t)=(a_1(\mathbf{x},t),a_2(\mathbf{x},t))$ is the velocity of the media, i.e., is the velocity field that the quantity C is moving with. For incompressible media, the velocity vector satisfies the condition $div[\mathbf{a}(\mathbf{x})]=0$. The term $R\left(\mathbf{x},C,C_x,C_y,t\right)$ describes "sources" or "sinks" of $C(\mathbf{x},t)$ (results of the chemical reactions, heat sources etc.). Below we represent this term in the form

$$R\left(\mathbf{x},C,C_x,C_y,t\right)=q\left(\mathbf{x},C,C_x,C_y,t\right)-f(\mathbf{x},t).$$

In engineering applications, the ADR equation expresses heat transfer and transport of mass and chemicals into porous or nonporous media [1]. The systems of ADR equations are common mathematical models used to describe the transport of contamination in atmosphere [2] and groundwater [3], radiation of microwaves [4], climate modelling [5], batch culture of biofilm [6] and wetland hydrology [7]. In most cases it is difficult and also time consuming to solve such problems explicitly. Therefore, it is necessary to obtain their approximate solutions by using some efficient numerical methods. The finite difference method (FDM) and the finite element (FEM) techniques [8] are classical tools for the numerical modeling of the ADR problem. A detailed review of the classic methods involving FEMs and FDMs can be found in [9]. Recent developments of these techniques can be found in [10, 11] and references therein. Spectral methods

(Galerkin, tau and collocation methods etc.) are other techniques which are widely used (see, e.g., [12, 13] and references therein).

In the last two decades there has been considerable interest in developing efficient meshless algorithms for solving partial differential equations [14–19]. In particular, concerning time-dependent ADR problems, Gharib et al. presented in [20] the meshless generalized reproducing kernel particle method to simulate time-dependent ADR problems with variable coefficients in a general $n-$dimensional space. Recently, there has been a research boom in developing meshless methods for solving engineering problems with RBFs [21]. In the 1990s, Kansa made the first attempt to extend RBFs for solving partial differential equations in fluid mechanics [22, 23]. After that, Kansa's method has been widely used in science and engineering for solving heat conduction problems, elastic problem, wave propagation problems, etc. The RBF-based methods have also been used for the simulation of ADR problems. Dehghan and Mohammadi have proposed [24] two numerical methods based on RBFs for solving the time-dependent linear and nonlinear Fokker-Planck equations in two dimensions. The compactly supported (CS) RBF and the local RBF methods have been used for solving the advection-diffusion equations in [25, 26]. Varun et al. proposed the RBF-finite difference method for solving the coupled problem of chemical transport in a fluid [27]. The thin plate spline radial basis function scheme for the advection diffusion problem was proposed by Boztosun et al. in [28]. A semi-analytical RBF collocation technique was proposed for steady-state strongly nonlinear ADR problems with variable coefficients in [29, 30]. The RBF finite collocation approach was proposed for capturing the sharp fronts for time-dependent advection problems in [31]. In [32] Askari and Adibi have presented the RBFs in combination with the method of lines for solving the advection diffusion equations. In [33] Dehghan and Shirzadi have studied a meshless method based on RBFs for solving stochastic advection-diffusion equations.

This paper presents a novel semi-analytical technique for solving the ADR equation based on the use of the RBFs for the space approximation of the solution and the Crank-Nicolson scheme for approximation in time. The second order Crank-Nicolson scheme is applied to transform the original Eq. (1.1) into a sequence of steady-state ADR equations. An improved version of the backward substitution method (BSM) is used for solving the stationary problem in this paper. This belongs to the category of the RBF-based meshless methods. The BSM was first proposed for solving the multi-point problems [34]. Then it was extended to steady-state heat conduction problems and multi-term fractional partial differential equations with time variable coefficients [35, 36]. Recently the BSM has been extended to the simulation of nonlinear 2D steady state ADR problems [29, 37] and the telegraph equation with variable coefficients [38]. It should be noted here that in the original BSM, the problem is transformed into a system of Laplace problem which are solved by the meshless method of fundamental solutions. However, we cannot form such general Laplace systems for general problems such as problems in anisotropic media. And, the original method can not be extended to problems whose governing equations do not have the Laplace operators. Furthermore, the method of fundamental solutions is

applied to solve the corresponding Laplace system. The optimal determination of the sources nodes in the method of fundamental solution remains an open issue. In this paper, a new version of this method has been proposed. The solution of the problems is divided into the approximation of the boundary data and the correcting functions by using the radial basis functions. The final solution is approximated by the summation of the primary approximation with series of basis functions which consist of a correcting function and a radial basis function with some free parameters. Then these free parameters are obtained by enforcing the approximations to satisfy the governing equations. The main difference between the new version BSM and the original BSM is that the approximations of the boundary data and the correcting functions are only on the boundary which don't have to satisfy any governing equations. In such a way the new version of the BSM can be easily extended to anisotropic problems which avoiding to solve the Laplace systems. As for the non-linear problems, the original equations are transformed into a system of linear ADR equations with variable coefficients in anisotropic media by using the quasilinearization technique [39] which can be solved by the proposed meshless scheme. It should be noted here that, since the approximations of the boundary data and the correcting functions do not have to satisfy the governing equations, the proposed method can be easily extended to other general fully nonlinear problems.

The rest of this paper is organized as follows. In Section 2 the mathematical description of the linear and nonlinear ADR problems is presented. We describe the main algorithm of the present method is in Section 3. Section 4 shows the accuracy of the proposed method, using numerical examples and making comparisons with other methods. The short conclusions and remarks are given in Section 5.

2 Mathematical description of the problem

Using Eqs. (1.2), (1.3) and Onsagar's reciprocity relation, the ADRE can be recast as follows:

$$\begin{aligned}\frac{\partial C(\mathbf{x},t)}{\partial t}=&D_{11}(\mathbf{x},C,t)\frac{\partial^2 C}{\partial x_1^2}+2D_{12}(\mathbf{x},C,t)\frac{\partial^2 C}{\partial x_1\partial x_2}+D_{22}(\mathbf{x},C,t)\frac{\partial^2 C}{\partial x_2^2}\\&+\left(\frac{dD_{11}(\mathbf{x},C,t)}{dx_1}+\frac{dD_{12}(\mathbf{x},C,t)}{dx_2}-a_1(\mathbf{x},t)\right)\frac{\partial C}{\partial x_1}\\&+\left(\frac{dD_{12}(\mathbf{x},C,t)}{dx_1}+\frac{dD_{22}(\mathbf{x},C,t)}{dx_2}-a_2(\mathbf{x},t)\right)\frac{\partial C}{\partial x_2}\\&-div\mathbf{a}(\mathbf{x},t)C+q\left(\mathbf{x},C,\frac{\partial C}{\partial x_1},\frac{\partial C}{\partial x_2},t\right)-f(\mathbf{x},t),\end{aligned}\tag{2.1}$$

where $\frac{d}{dx_k}D_{ij}(\mathbf{x},C,t)$ denotes the total derivative:

$$\frac{d}{dx_k}D_{ij}(\mathbf{x},C,t)=\frac{\partial D_{ij}(\mathbf{x},C,t)}{\partial x_k}+\frac{\partial D_{ij}(\mathbf{x},C,t)}{\partial C}\frac{\partial C}{\partial x_k}.$$

Remark 2.1. The general form of Eq. (2.1) includes many important equations. For example: the Fitzhugh-Nagumo equation [40], the Fokker-Planck equation [24], the generalized Burgers-Fisher equation [41] and others.

Dealing with a linear problem

$$\widehat{D}(\mathbf{x},C,t)=\widehat{D}(\mathbf{x},t),R\left(\mathbf{x},C,C_x,C_y,t\right)=q(\mathbf{x},t)C-f(\mathbf{x},t),$$

we use the simplified equation:

$$\begin{aligned}\frac{\partial C(\mathbf{x},t)}{\partial t}=&D_{11}(\mathbf{x},t)\frac{\partial^2 C}{\partial x_1^2}+2D_{12}(\mathbf{x},t)\frac{\partial^2 C}{\partial x_1\partial x_2}+D_{22}(\mathbf{x},t)\frac{\partial^2 C}{\partial x_2^2}\\&+\left(\frac{dD_{11}(\mathbf{x},t)}{dx_1}+\frac{dD_{12}(\mathbf{x},t)}{dx_2}-a_1(\mathbf{x},t)\right)\frac{\partial C}{\partial x_1}\\&+\left(\frac{dD_{12}(\mathbf{x},t)}{dx_1}+\frac{dD_{22}(\mathbf{x},t)}{dx_2}-a_2(\mathbf{x},t)\right)\frac{\partial C}{\partial x_2}\\&+(q(\mathbf{x},t)-div\mathbf{a}(\mathbf{x},t))C-f(\mathbf{x},t),\end{aligned}\tag{2.2}$$

or in the compact form:

$$\frac{\partial C(\mathbf{x},t)}{\partial t}=L(\mathbf{x},t)[C(\mathbf{x},t)]-f(\mathbf{x},t),\tag{2.3}$$

where L represents the linear differential operator, as follows:

$$\begin{aligned}L=&D_{11}(\mathbf{x},t)\frac{\partial^2}{\partial x_1^2}+2D_{12}(\mathbf{x},t)\frac{\partial^2}{\partial x_1\partial x_2}+D_{22}(\mathbf{x},t)\frac{\partial^2}{\partial x_2^2}\\&+\left(\frac{dD_{11}(\mathbf{x},t)}{dx_1}+\frac{dD_{12}(\mathbf{x},t)}{dx_2}-a_1(\mathbf{x},t)\right)\frac{\partial}{\partial x_1}\\&+\left(\frac{dD_{12}(\mathbf{x},t)}{dx_1}+\frac{dD_{22}(\mathbf{x},t)}{dx_2}-a_2(\mathbf{x},t)\right)\frac{\partial}{\partial x_2}+(q(\mathbf{x},t)-div\mathbf{a}(\mathbf{x},t)).\end{aligned}\tag{2.4}$$

The following Dirichlet boundary condition (BC) is presented:

$$C(\mathbf{x},t)=g_1(\mathbf{x},t),\quad \mathbf{x}\in\mathbf{\Gamma}_1,\tag{2.5}$$

and the Neumann BC for the boundary flux $Q_n(\mathbf{x},t)$

$$Q_n(\mathbf{x},t)=g_2(\mathbf{x},t),\quad \mathbf{x}\in\mathbf{\Gamma}_2,\tag{2.6}$$

where $\mathbf{n}=(n_1,n_2)$ denotes the unit outward normal vector, $\Gamma_1\cap\Gamma_2=\emptyset$, and $\Gamma_1\cup\Gamma_2=\partial\Omega$ and $g_1(\mathbf{x},t)$, $g_2(\mathbf{x},t)$ are known functions. The normal component of the heat flux on the boundary with the unit outward normal vector $\mathbf{n}=(n_1,n_2)$ has the form

$$Q_n(\mathbf{x},t)=\left(D_{11}(\mathbf{x},C,t)\frac{\partial u}{\partial x_1}+D_{12}(\mathbf{x},C,t)\frac{\partial C}{\partial x_2}\right)n_1+\left(D_{21}(\mathbf{x},C,t)\frac{\partial C}{\partial x_1}+D_{22}(\mathbf{x},C,t)\frac{\partial C}{\partial x_2}\right)n_2.\tag{2.7}$$

Below, for the sake of simplicity we denote the boundary as follows

$$\mathcal{B}[C(\mathbf{x},t)]=g(\mathbf{x},t), \quad \mathbf{x}\in\partial\Omega. \tag{2.8}$$

Besides this $C(\mathbf{x},t)$ satisfies the initial condition at $t=0$

$$C(\mathbf{x},0)=h(\mathbf{x}), \quad \mathbf{x}\in\Omega. \tag{2.9}$$

3 Main algorithm

3.1 Linear advection diffusion reaction problems

In order to solve Eq. (2.3), we first discretize the time domain. For this purpose we apply the Crank-Nicolson method. The main reason for choosing this method is its good convergence order. Theoretically, the Crank-Nicolson method is unconditionally stable [42].

Applying the Crank-Nicolson scheme to Eq. (2.3), we obtain the following systems:

$$\frac{C^{n+1}(\mathbf{x})-C^n(\mathbf{x})}{\Delta t}=\frac{1}{2}\left(L\left(\mathbf{x},t^{n+1}\right)\left[C^{n+1}(\mathbf{x})\right]+L(\mathbf{x},t^n)[C^n(\mathbf{x})]\right)-f\left(\mathbf{x},t^{n+1/2}\right), \tag{3.1}$$

where the superscripts n and $n+1$ denote successive time levels, $t^n=n\Delta t$, and we denote $C(\mathbf{x},t^n)=C^n(\mathbf{x})$. Next, changing the order of members, we get the equation for $C^{n+1}(\mathbf{x})$

$$\begin{aligned}\mathcal{L}^{n+1}\left[C^{n+1}(\mathbf{x})\right]&\equiv L\left(\mathbf{x},t^{n+1}\right)\left[C^{n+1}(\mathbf{x})\right]-\frac{2}{\Delta t}C^{n+1}(\mathbf{x})\\&=-L(\mathbf{x},t^n)[C^n(\mathbf{x})]-\frac{2}{\Delta t}C^n(\mathbf{x})+2f\left(\mathbf{x},t^{n+1/2}\right)\equiv F^{n+1}(\mathbf{x}).\end{aligned} \tag{3.2}$$

with the boundary condition

$$\mathcal{B}\left[C^{n+1}(\mathbf{x})\right]=g\left(\mathbf{x},t^{n+1}\right)\equiv g^{n+1}(\mathbf{x}), \quad \mathbf{x}\in\partial\Omega. \tag{3.3}$$

At the first time step we get the following equation:

$$\begin{aligned}\mathcal{L}^1\left[C^1(\mathbf{x})\right]&\equiv L\left(\mathbf{x},t^1\right)\left[C^1(\mathbf{x})\right]-\frac{2}{\Delta t}C^1(\mathbf{x})\\&=-L\left(\mathbf{x},t^0\right)\left[C^0(\mathbf{x})\right]-\frac{2}{\Delta t}C^0(\mathbf{x})+2f\left(\mathbf{x},t^{1/2}\right)\equiv F^1(\mathbf{x}),\end{aligned} \tag{3.4}$$

subject to the boundary condition

$$\mathcal{B}\left[C^1(\mathbf{x})\right]=g\left(\mathbf{x},t^1\right)\equiv g^1(\mathbf{x}), \quad \mathbf{x}\in\partial\Omega, \tag{3.5}$$

where $C^0(\mathbf{x})=h(\mathbf{x})$ is given by the initial condition. Suppose that $C_p^1(\mathbf{x})$ is a sufficiently smooth function which satisfies the boundary conditions of Eq. (2.8):

$$\mathcal{B}\left[C_p^1(\mathbf{x})\right]=g^1(\mathbf{x}), \quad \mathbf{x}\in\partial\Omega. \tag{3.6}$$

We denote

$$C^1(\mathbf{x}) = C_p^1(\mathbf{x}) + w^1(\mathbf{x}). \tag{3.7}$$

Then $w^1(\mathbf{x})$ is a solution of the problem with the same differential operator but with a new source term and with homogeneous boundary conditions on $\partial\Omega$

$$\mathcal{L}^1\left[w^1(\mathbf{x})\right] = F^1(\mathbf{x}) - \mathcal{L}^1\left[C_p^1(\mathbf{x})\right] = \widetilde{F}^1(\mathbf{x}), \quad \mathbf{x}\in\Omega, \tag{3.8}$$

$$\mathcal{B}\left[w^1(\mathbf{x})\right] = 0, \quad \mathbf{x}\in\partial\Omega. \tag{3.9}$$

Let $\phi_m(\mathbf{x})$ be a system of basis functions defined in the solution domain Ω. Throughout the paper we shall use RBFs of different kinds as the basis system. Let us define the corrected basis functions

$$\Phi_m(\mathbf{x}) = \phi_m(\mathbf{x}) + \omega_m(\mathbf{x}), \tag{3.10}$$

where the correcting functions $\omega_m(\mathbf{x})$ are chosen in such a way that $\Phi_m(\mathbf{x})$ satisfies the homogeneous boundary condition

$$\mathcal{B}[\Phi_m(\mathbf{x})] = 0, \quad \mathbf{x}\in\partial\Omega, \tag{3.11}$$

i.e.,

$$\mathcal{B}[\omega_m(\mathbf{x})] = -\mathcal{B}[\phi_m(\mathbf{x})], \quad \mathbf{x}\in\partial\Omega. \tag{3.12}$$

From Eq. (3.11) it follows that any linear combination

$$w^1(\mathbf{x}) = \sum_{m=1}^{\infty} q_m^1 \Phi_m(\mathbf{x}) \tag{3.13}$$

satisfies the homogeneous boundary condition Eq. (3.9). We assume that the solution of the problem Eqs. (3.8), (3.9) can be represented in the form Eq. (3.13) over the functions $\Phi_m(\mathbf{x})$.

Let us denote the functions $\varphi_m^1(\mathbf{x})$, $\mathbf{x}\in\Omega$ as follows:

$$\varphi_m^1(\mathbf{x}) = \mathcal{L}^1[\Phi_m(\mathbf{x})]. \tag{3.14}$$

It is easy to find that if

$$\sum_{m=1}^{\infty} q_m \varphi_m^1(\mathbf{x}) = \widetilde{F}^1(\mathbf{x}), \tag{3.15}$$

then Eq. (3.13) is a solution of the equation

$$\mathcal{L}^1\left[w^1\right] = \widetilde{F}^1(\mathbf{x}), \tag{3.16}$$

and the sum Eq. (3.7) satisfies the problem Eqs. (3.8), (3.9) with any choice of the parameters q_m^1.

We consider the truncated series

$$w_M^1(\mathbf{x}) = \sum_{m=1}^{M} q_m^1 \Phi_m(\mathbf{x}), \tag{3.17}$$

as an approximate solution of the problem. To get the parameters q_m we apply the collocation procedure to the Eq. (3.15) inside the solution domain

$$\sum_{m=1}^{M} q_m^1 \varphi_m^1(\mathbf{x}_j) = \widetilde{F}^1(\mathbf{x}_j), \quad \mathbf{x}_j \in \Omega, \quad j=1,\cdots,N \geq M. \tag{3.18}$$

After determining the unknown coefficients $\{q_m^1\}_{m=1}^M$, the approximate solution $C_M^1(\mathbf{x})$ of the problem Eqs. (3.8), (3.9) can be written as the sum $C_M^1(\mathbf{x}) = C_p^1(\mathbf{x}) + w_M^1(\mathbf{x})$.

At the next time steps we get a similar problem

$$\mathcal{L}^{n+1}\left[C^{n+1}(\mathbf{x})\right] = F^{n+1}(\mathbf{x}), \tag{3.19}$$

$$\mathcal{B}\left[C^{n+1}(\mathbf{x})\right] = g^{n+1}(\mathbf{x}), \quad \mathbf{x} \in \partial\Omega. \tag{3.20}$$

So, we can repeat all the manipulations Eqs. (3.6)-(3.18) with the new $\mathcal{L}^{n+1}[\cdots]$, F^{n+1} and g^{n+1} at each time step. As a result we get the approximate solution

$$C_M^{n+1}(\mathbf{x}) = w_M^{n+1}(\mathbf{x}) + C_p^{n+1}(\mathbf{x}), \tag{3.21}$$

where $C_p^{n+1}(\mathbf{x})$ is a sufficiently smooth function which approximates the boundary data

$$\mathcal{B}\left[C_p^{n+1}(\mathbf{x})\right] = g^{n+1}(\mathbf{x}), \quad \mathbf{x} \in \partial\Omega, \tag{3.22}$$

$$w_M^{n+1}(\mathbf{x}) = \sum_{m=1}^{M} q_m^{n+1} \Phi_m(\mathbf{x}), \tag{3.23}$$

and the coefficients $\{q_m^{n+1}\}_{m=1}^M$ are determined by the use of the collocation procedure inside the solution domain

$$\sum_{m=1}^{M} q_m^{n+1} \varphi_m^{n+1}(\mathbf{x}_j) = \widetilde{F}^{n+1}(\mathbf{x}_j), \quad \mathbf{x}_j \in \Omega, \quad j=1,\cdots,N \geq M. \tag{3.24}$$

Here

$$\varphi_m^{n+1}(\mathbf{x}) = \mathcal{L}^{n+1}[\Phi_m(\mathbf{x})] \tag{3.25}$$

and the functions $\Phi_m(\mathbf{x})$ are the same at all time levels.

3.2 Approximation of the boundary data

It should be noted that: 1) the functions $C_p^n(\mathbf{x})$, $\omega_m(\mathbf{x})$ should not necessarily satisfy any equation inside the solution domain. These are sufficiently smooth functions which only approximate the boundary data. Thus, to obtain them we can use the approximation by any system of functions which is complete in Ω; 2) this is not a 2D approximation over the domain. It is a 1D approximation of the boundary date over the boundary $\partial\Omega$. The following system of trigonometric functions is used for this goal:

$$\theta_k(\alpha,\mathbf{x})=\theta_{k_1,k_2}(\alpha,\mathbf{x})=\sin\left(k_1\pi\frac{x_1+\alpha}{2\alpha}\right)\sin\left(k_2\pi\frac{x_2+\alpha}{2\alpha}\right). \tag{3.26}$$

This system forms a complete orthogonal system in $[-\alpha,\alpha]\times[-\alpha,\alpha]$ for 2D problems. Choosing α large enough to satisfy $\Omega\subset\Omega_\alpha$, we approximate the correcting functions $\omega_m(\mathbf{x})$ and $C_p^n(\mathbf{x})$ by the sums:

$$\omega_m(\mathbf{x})=\sum_{k=1}^{K}p_{m,k}\theta_k(\alpha,\mathbf{x}),\quad C_p^n(\mathbf{x})=\sum_{k=1}^{K}p_{M+1,k}^n\theta_k(\alpha,\mathbf{x}), \tag{3.27}$$

where k_1 and k_2 are given in the sequence which satisfies $k_1+k_2=const$ from 1 until the number of basis function reaches K, i.e. the trigonometric functions are posted in the order: $\theta_{1,1}$,$\theta_{2,1}$,$\theta_{1,2}$, $\theta_{3,1}$,$\theta_{2,2}$,$\theta_{1,3}$, etc. By using the collocation procedure, we get the linear systems:

$$\sum_{k=1}^{K}p_{m,k}\mathcal{B}[\theta_k(\alpha,\mathbf{y}_i)]=-\mathcal{B}[\phi_m(\mathbf{y}_i)],\quad \mathbf{y}_i\in\partial\Omega,\quad i=1,\cdots,K_1, \tag{3.28}$$

$$\sum_{k=1}^{K}p_{M+1,k}^n\mathcal{B}[\theta_k(\alpha,\mathbf{y}_i)]=g(\mathbf{y}_i,t^n),\quad \mathbf{y}_i\in\partial\Omega,\quad i=1,\cdots,K_1. \tag{3.29}$$

We take the number of the collocation points $K_1>K$. Note that the linear systems Eqs. (3.28), (3.29) have the same matrix and different right hand sides and are solved by a single call of the standard procedure.

Remark 3.1. The system of trigonometric functions Eq. (3.26) is not the only possible basis system for the approximation of the boundary data. The use of polynomials and RBFs for this goal is demonstrated in [30, 35]. From the above-mentioned illustrations, we have three linear systems Eq. (3.24) and Eqs. (3.28) and (3.29) to be solved to obtain the numerical approximations. From Eqs. (3.28) and (3.29), we can see that the same coefficient matrix $\mathcal{B}[\theta_k(\alpha,\mathbf{y}_i)]$ are used for the approximation of $\omega_m(\mathbf{x})$ and $C_p(\mathbf{x})$ with different right-hand-sides $-\mathcal{B}[\phi_m(\mathbf{y}_i)]$ and $g(\mathbf{y}_i,t^n)$. Therefore, the two linear systems are of $N\times M$ and $K_1\times K$ where $N>M$ and $K_1>K$ where N is the number of collocation nodes inside the solution domain and K_1 is the number of collocation nodes on the boundary. For some well-known methods such as Kansa's method, we will form the $(N+K_1)\times(N+$

K_1) matrix where N is the number of collocation nodes inside the solution domain and K_1 is the number of collocation nodes on the boundary. For the proposed method, we form two linear systems of $N\times M$ and $K_1\times K$ with $N>M$ and $K_1>K$. If we take $M=N$, $K=K_1$, we have the maximum dimension of the matrix of $N\times N$ and $K_1\times K_1$. If the Gaussian elimination method is used to solve such systems, we reduce the computational complexity from $\mathcal{O}\big((N+K_1)^3\big)$ to $\mathcal{O}\big(N^3+K_1^3\big)$. Furthermore, in this paper, only a few number of M and K is required which can reduce the computational cost shapely.

3.3 RBF basis systems

To solve 2D problems, we have chosen the multiquadric function, the Gaussian radial basis function, and the conical radial basis functions to construct the functions φ and Φ. The multiquadric function (MQ) is defined as

$$\phi_m(\mathbf{x})=\sqrt{r_m^2+c^2}=\sqrt{(x_1-x_{1,m})^2+(x_2-x_{2,m})^2+c^2}. \tag{3.30}$$

The Gaussian radial basis function (GRBF) is defined

$$\phi_m(\mathbf{x})=\exp\left(-\left(\frac{r_m}{c}\right)^2\right)=\exp\left(-\left(\frac{\sqrt{(x_1-x_{1,m})^2+(x_2-x_{2,m})^2}}{c}\right)^2\right). \tag{3.31}$$

The conical radial basis function (CRBF) is defined as follows:

$$\phi_m(\mathbf{x})=r_m^{13}=\big((x_1-x_{1,m})^2+(x_2-x_{2,m})^2\big)^{13/2}. \tag{3.32}$$

In the above definitions $\{(x_{1,m},x_{2,m})\}_{m=1}^M$ are the centers of the basis functions and c is an arbitrary constant called the shape parameter. Much effort has been devoted to finding the optimal shape parameters for the radial basis function based on such methods as the variable shape parameters and random shape parameters (see, e.g., [43] and the references therein). In this paper we prescribe the constant value of the shape parameter to verify the accuracy of the present method.

3.4 Fully nonlinear problems

In this subsection, we consider the use of the method described above in the case of the general nonlinear ADR equation. Let us denote

$$v_i=\frac{\partial C}{\partial x_i},\quad i=1,2,\quad \chi_{11}=\frac{\partial^2 C}{\partial x_1^2},\quad \chi_{12}=\frac{\partial^2 C}{\partial x_1\partial x_2},\quad \chi_{22}=\frac{\partial^2 C}{\partial x_2^2}.$$

By using the quasilinearization technique, we consider C, v_i, and χ_{ij} as independent variables. Using this notation, Eq. (2.1) can be written in the following form:

$$\begin{aligned}\frac{\partial C(x,t)}{\partial t}=&D_{11}(\mathbf{x},C,t)\chi_{11}+2D_{12}(\mathbf{x},C,t)\chi_{12}+D_{22}(\mathbf{x},C,t)\chi_{22}\\&+\left[\frac{\partial D_{11}(\mathbf{x},C,t)}{\partial x_1}+\frac{\partial D_{11}(\mathbf{x},C,t)}{\partial C}v_1+\frac{\partial D_{12}(\mathbf{x},C,t)}{\partial x_2}+\frac{\partial D_{12}(\mathbf{x},C,t)}{\partial C}v_2-a_1(\mathbf{x},t)\right]v_1\\&+\left[\frac{\partial D_{12}(\mathbf{x},C,t)}{\partial x_1}+\frac{\partial D_{12}(\mathbf{x},C,t)}{\partial C}v_1+\frac{\partial D_{22}(\mathbf{x},C,t)}{\partial x_2}+\frac{\partial D_{22}(\mathbf{x},C)}{\partial C}v_2-a_2(\mathbf{x},t)\right]v_2\\&-div\mathbf{a}(\mathbf{x},t)C+q(\mathbf{x},C,v_1,v_2,t)-f(\mathbf{x},t),\end{aligned}\tag{3.33}$$

or in the short form:

$$\frac{\partial C(\mathbf{x},t)}{\partial t}=L\left(\mathbf{x},C,v_i,\chi_{ij},t\right)[C(\mathbf{x},t)]-f(\mathbf{x},t),\tag{3.34}$$

where L is the nonlinear differential operator. By using the Crank-Nicolson scheme to Eq. (3.34), we obtain the following system of equations:

$$\begin{aligned}\frac{C^{n+1}(\mathbf{x})-C^n(\mathbf{x})}{\Delta t}=&\frac{1}{2}\Big\{L\left(\mathbf{x},C^{n+1},v_i^{n+1},\chi_{ij}^{n+1},t^{n+1}\right)\left[C^{n+1}(\mathbf{x})\right]\\&+L\left(\mathbf{x},C^n,v_i^n,\chi_{ij}^n,t^n\right)[C^n(\mathbf{x})]\Big\}-f\left(\mathbf{x},t^{n+1/2}\right),\end{aligned}\tag{3.35}$$

where the superscripts n and $n+1$ are successive time levels, $t^n=n\Delta t$, and we denote $C(\mathbf{x},t^n)=C^n(\mathbf{x})$. Next, reordering the terms, we get

$$\begin{aligned}&L\left(\mathbf{x},C^{n+1},v_i^{n+1},\chi_{ij}^{n+1},t^{n+1}\right)\left[C^{n+1}(\mathbf{x})\right]-\frac{2}{\Delta t}C^{n+1}(\mathbf{x})\\=&-L\left(\mathbf{x},C^n,v_i^n,\chi_{ij}^n,t^n\right)[C^n(\mathbf{x})]-\frac{2}{\Delta t}C^n(\mathbf{x})+2f\left(\mathbf{x},t^{n+1/2}\right).\end{aligned}\tag{3.36}$$

In order to obtain C^{n+1}, v_i^{n+1}, χ_{ij}^{n+1} from Eq. (3.36), the C^n, v_i^n, χ_{ij}^n in the right-hand side should be obtained in advance. It should be noted here that the C^n, v_i^n, χ_{ij}^n have been obtained from the previous time steps. Therefore the term $L\left(\mathbf{x},C^n,v_i^n,\chi_{ij}^n,t^n\right)[C^n(\mathbf{x})]$ can be obtained directly. So, the right hand-side of the equation is a known function of the space coordinate. Therefore, the quasilinearization technique is applied only to the left hand side of Eq. (3.36).

Suppose that C_0^{n+1}, $v_{i,0}^{n+1}$, $\chi_{ij,0}^{n+1}$ are the given functions of $\mathbf{x}$ which are the initial approximations of the corresponding exact values at the $n+1$ steps. Then, we have the following relations:

$$\begin{aligned}C^{n+1}&=C_0^{n+1}+(C^{n+1}-C_0^{n+1})=C_0^{n+1}+\delta C^{n+1},\\v_i^{n+1}&=v_{i,0}^{n+1}+(v_i^{n+1}-v_{i,0}^{n+1})=v_{i,0}^{n+1}+\delta v_i^{n+1},\\\chi_{ij}^{n+1}&=\chi_{ij,0}^{n+1}+(\chi_{ij}^{n+1}-\chi_{ij,0}^{n+1})=\chi_{ij,0}^{n+1}+\delta\chi_{ij}^{n+1},\end{aligned}$$

where δC^{n+1}, δv_i^{n+1}, and $\delta\chi_{ij}^{n+1}$ are correcting functions. Assuming that δC^{n+1}, δv_i^{n+1}, and $\delta\chi_{ij}^{n+1}$ are small, the left-hand side of Eq. (3.36) can be linearized by using the quasilinearization technique. Let us consider the linearization of the first term in $L\left(\mathbf{x},C^{n+1},v_i^{n+1},\chi_{ij}^{n+1},t^{n+1}\right)$ of Eq. (3.36):

$$
\begin{aligned}
&D_{11}\left(\mathbf{x},C^{n+1},t^{n+1}\right)\chi_{11}^{n+1}=D_{11}\left(\mathbf{x},C_0^{n+1}+\delta C^{n+1},t^{n+1}\right)\left(\chi_{11,0}^{n+1}+\delta\chi_{11}^{n+1}\right) && (3.37)\\
\simeq&\left(D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)+\partial_C D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\delta C^{n+1}\right)\left(\chi_{11,0}^{n+1}+\delta\chi_{11}^{n+1}\right) && (3.38)\\
\simeq&D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\chi_{11,0}^{n+1}+\partial_C D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\chi_{11,0}^{n+1}\delta C^{n+1}\\
&+D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\delta\chi_{11}^{n+1} && (3.39)\\
=&D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\chi_{11,0}^{n+1}+\partial_C D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\chi_{11,0}^{n+1}(C^{n+1}-C_0^{n+1})\\
&+D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)(\chi_{11}^{n+1}-\chi_{11,0}^{n+1}) && (3.40)\\
=&D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\chi_{11}^{n+1}+\partial_C D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\chi_{11,0}^{n+1}C^{n+1}\\
&-\partial_C D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\chi_{11,0}^{n+1}C_0^{n+1}. && (3.41)
\end{aligned}
$$

It is noted here that the second power of correcting functions $(\delta C^{n+1}\delta\chi_{11}^{n+1})$ is ignored from Eq. (3.38) to Eq. (3.39) since the correcting functions are small. The final expression is a linear equation which contains only two unknowns: χ_{11}^{n+1} and C^{n+1}. The last term of the expression is a known function of $\mathbf{x}$. In the same way, the next two terms of the Eq. (3.36) can be linearized as follows:

$$
\begin{aligned}
&2D_{12}\left(\mathbf{x},C^{n+1},t^{n+1}\right)\chi_{12}^{n+1}\\
\simeq&2D_{12}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\chi_{12}^{n+1}+2\partial_C D_{12}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\chi_{12,0}^{n+1}C^{n+1}\\
&-2\partial_C D_{12}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\chi_{12,0}^{n+1}C_0^{n+1}, && (3.42)\\
&D_{22}\left(\mathbf{x},C^{n+1},t^{n+1}\right)\chi_{22}^{n+1}\\
\simeq&D_{22}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\chi_{22}^{n+1}+\partial_C D_{22}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\chi_{22,0}^{n+1}C^{n+1}\\
&-\partial_C D_{22}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\chi_{22,0}^{n+1}C_0^{n+1}, && (3.43)
\end{aligned}
$$

Eq. (3.42) and Eq. (3.43) are linear equations which contain three unknowns χ_{12}^{n+1}, χ_{22}^{n+1}, and C^{n+1}.

Let us consider the next part of Eq. (3.36):

$$
\begin{aligned}
&\left[\frac{\partial D_{11}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial x_1}+\frac{\partial D_{11}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial C}v_1^{n+1}\right]v_1^{n+1}\\
&\quad+\left[\frac{\partial D_{12}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial x_2}+\frac{\partial D_{12}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial C}v_2^{n+1}-a_1\left(\mathbf{x},t^{n+1}\right)\right]v_1^{n+1}\\
=&\overset{(1a)}{\frac{\partial D_{11}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial x_1}v_1^{n+1}}+\overset{(2a)}{\frac{\partial D_{11}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial C}\left(v_1^{n+1}\right)^2}\\
&\quad+\overset{(3a)}{\frac{\partial D_{12}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial x_2}v_1^{n+1}}+\overset{(4a)}{\frac{\partial D_{12}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial C}v_1^{n+1}v_2^{n+1}}-\overset{\text{linear term}}{a_1\left(\mathbf{x},t^{n+1}\right)v_1^{n+1}}.
\end{aligned}
$$

The nonlinear terms are transformed as follows using the same way by ignoring the high power of the correcting functions:

$$
\begin{aligned}
(1a)\simeq&\partial_{x_1,C}D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)v_{1,0}^{n+1}C^{n+1}+\partial_{x_1}D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)v_1^{n+1}\\
&-\partial_{x_1,C}D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)v_{1,0}^{n+1}C_0^{n+1},\\
(2a)\simeq&\partial_{C,C}D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\left(v_{1,0}^{n+1}\right)^2C^{n+1}+2v_{1,0}^{n+1}\partial_C D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)v_1^{n+1}\\
&-\partial_C D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\left(v_{1,0}^{n+1}\right)^2-\partial_{C,C}D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\left(v_{1,0}^{n+1}\right)^2C_0^{n+1},\\
(3a)\simeq&\partial_{x_2,C}D_{12}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)v_{1,0}^{n+1}C^{n+1}+\partial_{x_2}D_{12}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)v_1^{n+1}\\
&-\partial_{x_2,C}D_{12}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)v_{1,0}^{n+1}C_0^{n+1},\\
(4a)\simeq&\partial_{C,C}D_{12}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)v_{1,0}^{n+1}v_{2,0}^{n+1}C^{n+1}+\partial_C D_{12}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)v_{2,0}^{n+1}v_1^{n+1}\\
&+\partial_C D_{12}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)v_{1,0}^{n+1}v_2^{n+1}-\partial_{C,C}D_{12}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)v_{1,0}^{n+1}v_{2,0}^{n+1}C_0^{n+1}\\
&-\partial_C D_{12}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)v_{1,0}^{n+1}v_{2,0}^{n+1}.
\end{aligned}
\tag{3.44}
$$

Let us consider the next term:

$$
\begin{aligned}
&\left[\frac{\partial D_{12}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial x_1}+\frac{\partial D_{12}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial C}v_1^{n+1}\right]v_2^{n+1}\\
&\quad+\left[\frac{\partial D_{22}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial x_2}+\frac{\partial D_{22}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial C}v_2^{n+1}-a_2\left(\mathbf{x},t^{n+1}\right)\right]v_2^{n+1}
\end{aligned}
$$

$$
\begin{aligned}
&=\overset{(1b)}{\frac{\partial D_{12}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial x_1}}v_2^{n+1}+\overset{(2b)}{\frac{\partial D_{12}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial C}}v_1^{n+1}v_2^{n+1}\\
&\quad+\overset{(3b)}{\frac{\partial D_{22}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial x_2}}v_2^{n+1}+\overset{(4b)}{\frac{\partial D_{22}\left(\mathbf{x},C^{n+1},t^{n+1}\right)}{\partial C}}\left(v_2^{n+1}\right)^2-\overset{\text{linear term}}{a_2\left(\mathbf{x},t^{n+1}\right)}v_2^{n+1},
\end{aligned}
$$

where the term (1b) is transformed as term (1a) with the substitutions $D_{11}\to D_{12}$, $v_1\to v_2$. The term (2b) is the same as the term (4a). The term (3b) is transformed as term (1a) with the substitutions $D_{11}\to D_{22}$, $v_1\to v_2$, $\partial_{x_1}\to\partial_{x_2}$. The term (4$b$) is transformed as term (2a) with the substitutions $D_{11}\to D_{22}$, $v_1\to v_2$. Therefore, it can be easily linearized from Eq. (3.44).

Let us consider the last nonlinear term $q\left(\mathbf{x},C^{n+1},v_1^{n+1},v_2^{n+1}\right)$ using the same method

$$
\begin{aligned}
&q\left(\mathbf{x},C^{n+1},v_1^{n+1},v_2^{n+1},t^{n+1}\right)\\
\simeq&\partial_C q\left(\mathbf{x},C_0^{n+1},v_{1,0}^{n+1},v_{2,0}^{n+1},t^{n+1}\right)C^{n+1}+\partial_{v_1}q\left(\mathbf{x},C_0^{n+1},v_{1,0}^{n+1},v_{2,0}^{n+1},t^{n+1}\right)v_1^{n+1}\\
&+\partial_{v_2}q\left(\mathbf{x},C_0^{n+1},v_{1,0}^{n+1},v_{2,0}^{n+1},t^{n+1}\right)v_2^{n+1}-\partial_C q\left(\mathbf{x},C_0^{n+1},v_{1,0}^{n+1},v_{2,0}^{n+1},t^{n+1}\right)C_0^{n+1}\\
&-\partial_{v_1}q\left(\mathbf{x},C_0^{n+1},v_{1,0}^{n+1},v_{2,0}^{n+1},t^{n+1}\right)v_{1,0}^{n+1}-\partial_{v_2}q\left(\mathbf{x},C_0^{n+1},v_{1,0}^{n+1},v_{2,0}^{n+1},t^{n+1}\right)v_{2,0}^{n+1}.
\end{aligned}
\tag{3.45}
$$

Using the formulae Eqs. (3.37)-(3.45), the original nonlinear Eq. (2.1) is transformed to the linear form:

$$
\begin{aligned}
&D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\frac{\partial^2 C^{n+1}}{\partial x_1^2}+2D_{12}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\frac{\partial^2 C^{n+1}}{\partial x_1\partial x_2}+D_{11}\left(\mathbf{x},C_0^{n+1},t^{n+1}\right)\frac{\partial^2 C^{n+1}}{\partial x_2^2}\\
&\quad+B_1\left(\mathbf{x},C_0^{n+1},v_{i,0}^{n+1},t^{n+1}\right)\frac{\partial C^{n+1}}{\partial x_1}+B_2\left(\mathbf{x},C_0^{n+1},v_{i,0}^{n+1},t^{n+1}\right)\frac{\partial C^{n+1}}{\partial x_2}\\
&\quad+\left[B_3\left(\mathbf{x},C_0^{n+1},v_{i,0}^{n+1},\chi_{ij,0}^{n+1},t^{n+1}\right)-\frac{2}{\Delta t}\right]C^{n+1}\\
&=B_4\left(\mathbf{x},C_0^{n+1},v_{i,0}^{n+1},\chi_{ij,0}^{n+1},t^{n+1}\right)-L\left(\mathbf{x},C^n,v_i^n,\chi_{ij}^n,t^n\right)\left[C^n(\mathbf{x})\right]-\frac{2}{\Delta t}C^n(\mathbf{x})+2f\left(\mathbf{x},t^{n+1/2}\right),
\end{aligned}
\tag{3.46}
$$

where the coefficients B_i depend on the initial approximations $C_0^{n+1}(\mathbf{x})$, $v_{i,0}^{n+1}(\mathbf{x})$, $\chi_{ij,0}^{n+1}(\mathbf{x})$ and, so, are the known functions. Therefore, linear Eq. (3.46) can be solved by the RBF-based method described above. The coefficients $B_i(\mathbf{x})$ are changed during the inner iterations with each fixed t^{n+1}. Note, that the term $-L\left(\mathbf{x},C^n,v_i^n,\chi_{ij}^n,t^n\right)\left[C^n(\mathbf{x})\right]-\frac{2}{\Delta t}C^n(\mathbf{x})+2f\left(\mathbf{x},t^{n+1/2}\right)$ in the right hand side of the equation and the function $C_p^{n+1}(\mathbf{x})$ are fixed during the inner iterations. Usually 3-5 iterations at each time step are enough to obtain convergent solution. It should be noted here that for problems without exact solutions, we may stop the iteration when the difference between values of two successive steps

is less than the required tolerance or simply fix the number of iterations. In this paper, we fix the number of iterations. As for the initial approximations $C_0^{n+1}(\mathbf{x})$, we take $C_0^n(\mathbf{x})=C_p(\mathbf{x})$, $C_0^n(\mathbf{x})\equiv 0$, $C_0^n(\mathbf{x})\equiv 1$ and $C_0^n(\mathbf{x})=rand$ (*rand*: Uniformly distributed pseudorandom numbers on the open interval (0,1). The $v_{i,0}^{n+1}(\mathbf{x})$ and $\chi_{ij,0}^{n+1}(\mathbf{x})$ are taken by corresponding derivatives with respective to $\mathbf{x}$.

4 Numerical examples and discussions

To validate the accuracy and efficiency of the present method we consider several examples. It is noted that the computations are carried out in MATLAB in OS windows 7 (64bit) with Intel Core I7-6500, 2.5GHz CPU and 12GB memory. To evaluate the numerical accuracy the error norm is defined in the following form:

$$L_\infty=\max_{1\le i\le N_t}|u_{\text{exact}}(\mathbf{x}_i)-u_M(\mathbf{x}_i)|, \tag{4.1}$$

where u_{exact} and u_M are the analytical and approximate solution, respectively, N_t is the number of the test points $\mathbf{x}_i$ which are randomly distributed inside the solution domain Ω.

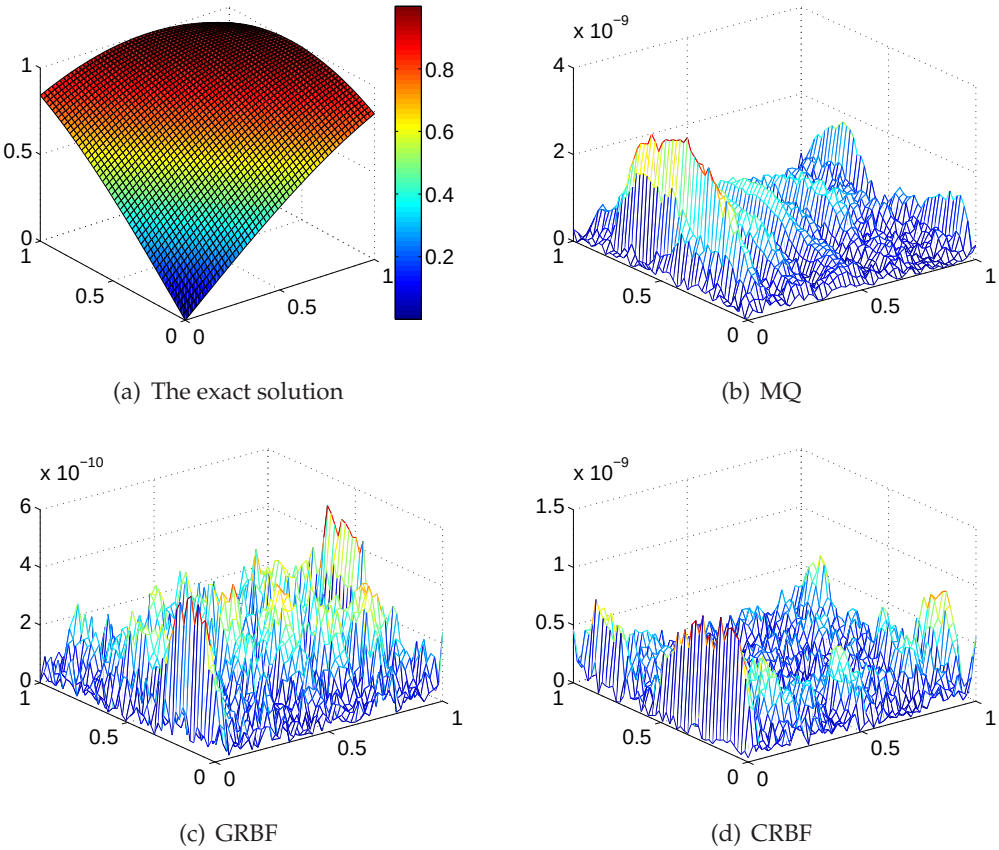
Example 4.1. As the first example we consider the following linear ADR equation

$$\frac{\partial C(x,y,t)}{\partial t}=-y\frac{\partial C(x,y,t)}{\partial x}+(y-x^2)\frac{\partial C(x,y,t)}{\partial y}+C(x,y,t)+\frac{\partial^2 C(x,y,t)}{\partial y^2}-f(x,y,t),$$
$$(x,y)\in\Omega,\quad 0<t<T, \tag{4.2}$$

in the square $\Omega=\{(x,y):0\le x,y\le 1\}$. The initial condition, the Dirichlet boundary condition on $\partial\Omega$ and the source term $f(x,y,t)$ correspond to the exact solution

$$C(x,y,t)=\sin[(x+y)]t. \tag{4.3}$$

Fig. 1 shows the maximal absolute error obtained with the use of the RBFs of different kinds (MQ, GRBF, and CRBF). The shape parameter $c=1$ is the same for the MQ and GRBF basis functions. The numerical results are obtained with the fixed number of the RBF centers $M=225$ and with $N=400$ collocation nodes inside the solution domain. The number of the trigonometric products (3.26) in the approximation of the boundary data (the functions $\omega_m(\mathbf{x})$ and $C_p^n(\mathbf{x})$) is K=150. The number of the collocation points on the boundary is $K_1=160$. The time step is $\Delta t=0.1$ and the data which are shown in the figure correspond to the time $T=1$. The figure shows that the present method provides a good approximation for all the RBFs considered. Fig. 2 displays the absolute maximal error as a function of the shape parameter c for MQ and GRBF basis functions. The error decreases with the growth of the c for small values of c and it reaches the minimal value at c_{opt}. Then the error increases with a further increase of the parameter c. This is the



(a) The exact solution (b) MQ

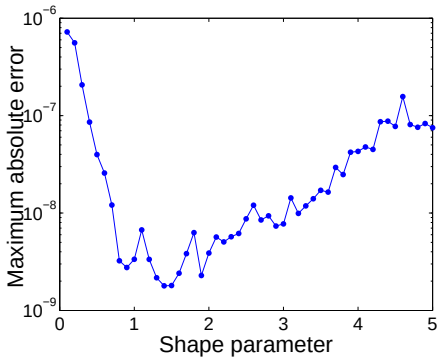
(c) GRBF (d) CRBF

Figure 1: Example 4.1. The exact solution and the absolute errors for the MQ, the GRBF and the CRBF.

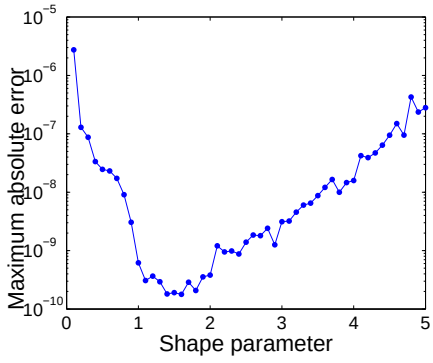
common trend for RBF-based numerical methods. However, as Fig. 2 shows, the worst maximum absolute error is about 10^{-6} which is acceptable for practical applications.

Fig. 3 demonstrates the convergence of the method with the growth of the number of the trigonometric functions K, the number of boundary nodes K_1 and with the growth of the number of centers of the RBFs M. The figure shows that the error decreases sharply with the increasing of the parameters K, K_1, and M. With the further growth of the parameters it keeps around 10^{-10}. The data shown in Fig. 4 correspond to the time $T=1000$, i.e., the steady-state solution. They are obtained using the MQ RBF ($c=1$) and the large time step $\Delta t=100$. These data demonstrate the stability of the present method.

The data placed in Table 1 correspond to the MQ RBF with two different values of the shape parameter: $c=0.6$ and $c=1$. The time step is $\Delta t=0.001$ and the final time is $T=0.1$. The number of the centers of the MQ RBFs is $M=100$, the number of the collocation points is $N=121$. The parameters of the approximation of the boundary data are: $K=20$, $K_1=44$ and $\alpha=5$. The data of the table demonstrate that the approximate solution obtained with the shape parameter $c=0.6$ is much more accurate. Dehghan and

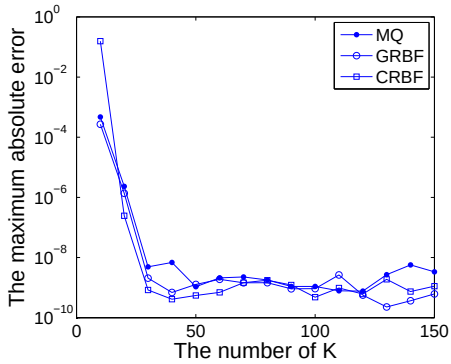


(a) MQ

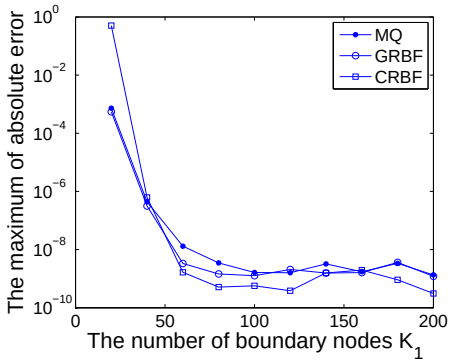


(b) GRBF

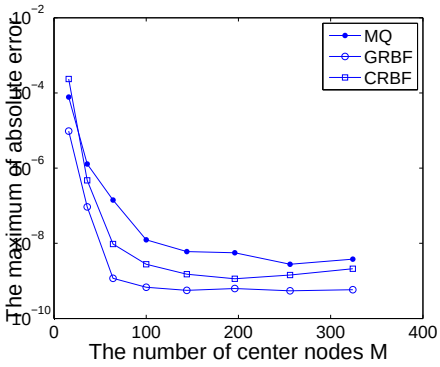
Figure 2: Example 4.1. The maximum absolute error as a function of the shape parameter for the MQ and GRBF.



(a) The maximum absolute error as a function of K

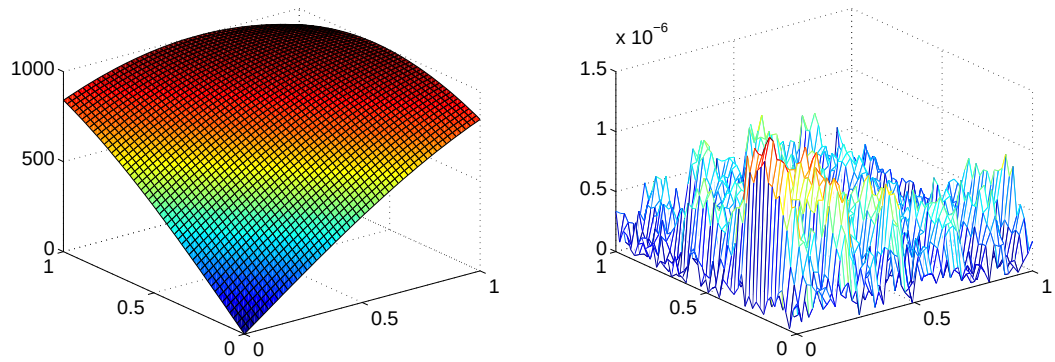


(b) The maximum absolute error as a function of K_1



(c) The maximum absolute error as a function of M

Figure 3: Example 4.1. The convergence as a function of the K, K_1 and M.



(a) Numerical results (b) Absolute errors

Figure 4: Example 4.1. Numerical results and absolute errors at $T=1000$ using $\Delta t=100$.

Table 1: Example 4.1. The maximum absolute errors at $T=0.1$ using the MQ basis.

(x,y)	*Exact*	$MQ\,(c=0.6)$	$MQ\,(c=1)$	Ref. [24]
(0.1,0.1)	0.019866933079506	4.15×10^{-8}	3.53×10^{-5}	1.45×10^{-5}
(0.2,0.2)	0.038941834230865	2.94×10^{-8}	7.40×10^{-6}	5.46×10^{-5}
(0.3,0.3)	0.056464247339504	2.29×10^{-8}	4.61×10^{-6}	1.31×10^{-4}
(0.4,0.4)	0.071735609089952	3.16×10^{-8}	7.73×10^{-6}	2.17×10^{-4}
(0.5,0.5)	0.084147098480790	3.25×10^{-8}	9.93×10^{-6}	2.82×10^{-4}

Mohammadi [24] have considered this problem by using Kansa's approach. The data of their calculation are shown in the right hand side of the table. The comparison shows that the present method provides a more accurate solution even when the shape parameter $c=1$ is used.

Finally, we display the elapsed time at the first time step size versus the number of domain collocation nodes N with M=25, K_1=20, and K=80 using the MQ basis functions $(c=1)$ in Fig. 5(left). The elapsed time versus the number of boundary collocation nodes K_1 with $M=25$, $K=20$, and N=324 is displayed in Fig. 5(right). From this figure, it is evidently that the computational complexity is reduced which is far less than the traditional methods about $\mathcal{O}\big((N+K_1)^3\big)$ such as the Kansa's method.

Example 4.2. In this example, we consider the ADR equation which models the transfer process in the anisotropic media. Let us consider Eq. (2.1) with the following coefficients

$$D_{11}(\mathbf{x},t)=(1+0.2(x_1+x_2))\exp(x_1+x_2+t),$$
$$D_{22}(\mathbf{x},t)=(1+0.2(x_1-x_2))\exp(x_1-x_2+t),$$

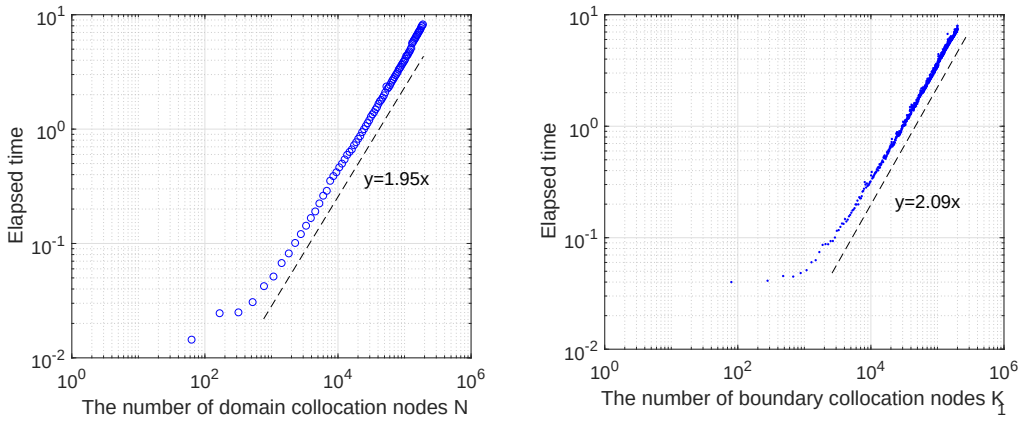


Figure 5: Example 4.1. The elapsed time versus the number of domain collocation nodes N (left) and the boundary collocation nodes K_1 (right) respectively.

$$
\begin{aligned}
&D_{12}(\mathbf{x},t)=D_{21}(\mathbf{x},t)=0.1(x_1+x_2)\exp(x_1-x_2+t),\\
&a_1(\mathbf{x},t)=-1-2x_1^2x_2-t,\quad a_2(\mathbf{x},t)=1+2x_1x_2^2-t,\\
&q(\mathbf{x},t,C,C_x,C_y)=-1-\sin^2(x_1+x_2)-0.1|\mathbf{x}|^2C^2-0.1C_xC_y,
\end{aligned}
$$

with the solution domain shown in Fig. 6 which is bounded by the following parametric equation

$$
\partial\Omega=\{(x_1,x_2)\mid x_1=\rho(s)\cos(s),x_2=\rho(s)\sin(s),0\le s\le 2\pi\},\tag{4.4}
$$

where

$$
\rho(s)=\Big(\cos(3s)+\big(2-\sin^2(3s)\big)^{1/2}\Big)^{1/3}.
$$

The source term f in Eq. (2.1) and the Dirichlet boundary conditions can be obtained from the exact solution

$$
C(x,y,t)=\exp(x-y-t)\cos(y).\tag{4.5}
$$

Table 2 shows the maximum absolute errors at the times $T=0.2$, 0.4, 0.8, 1.0, 5, 10 with different time steps Δt. The data correspond to the parameters: $K=100$, $K_1=160$, $M=54$ and $N=110$ using the MQ RBF ($c=1$). Three inner iterations are used on each time layer in the procedure of the quasilinearization. The data show that the error decreases monotonically with the decreasing of the time step size. Fig. 7 also demonstrates the accuracy of the present method. It displays the analytical solution and the absolute error of the approximate solution. Finally, Table 3 shows the maximum absolute errors at $T=1$ versus the number of iterations using different initial solutions. It is obvious that if $C_p(\mathbf{x})$ is used as the initial approximation $C_0^n(\mathbf{x})$ in the inner iterations on each time layer, then the most accurate results are obtained even with only one iteration. For other initial approximations $C_0^n(\mathbf{x})\equiv 0$, $C_0^n(\mathbf{x})\equiv 1$ and $C_0^n(\mathbf{x})=rand$ ($rand$: Uniformly distributed

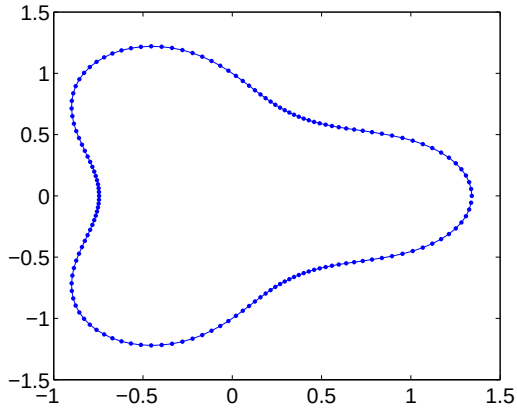


Figure 6: Example 4.2. The profile of the solution domain.

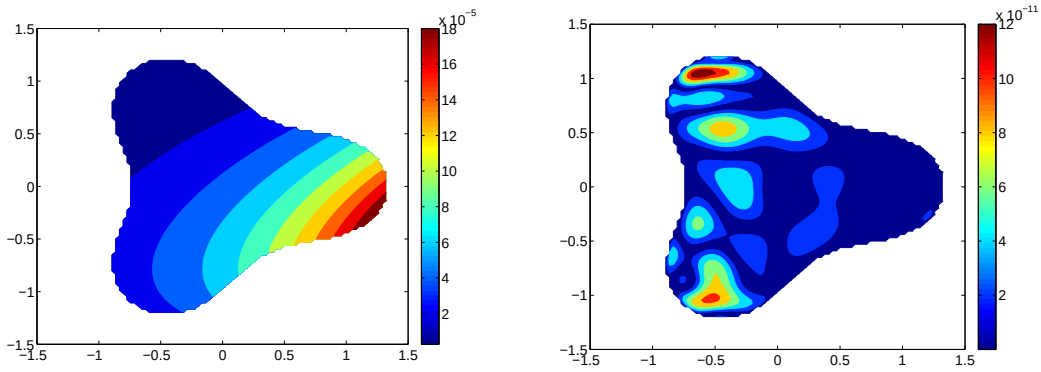
Table 2: Example 4.2. The maximum absolute error versus the time step size.

T	$\Delta t=0.1$	$\Delta t=0.05$	$\Delta t=0.01$	$\Delta t=0.001$
0.2	1.85×10^{-4}	4.81×10^{-5}	3.37×10^{-5}	1.90×10^{-6}
0.4	1.46×10^{-4}	3.88×10^{-5}	2.76×10^{-5}	1.58×10^{-6}
0.8	6.67×10^{-5}	1.95×10^{-5}	1.86×10^{-5}	1.06×10^{-6}
1	4.52×10^{-5}	1.44×10^{-5}	1.52×10^{-5}	8.73×10^{-7}
5	2.80×10^{-7}	2.71×10^{-7}	2.70×10^{-7}	1.81×10^{-8}
10	1.85×10^{-9}	1.82×10^{-9}	1.83×10^{-9}	1.27×10^{-10}

Table 3: Example 4.2. The maximum absolute error at $T=1$ versus the number of iterations using different initial solution.

Number of Iterations	$C=u_p$	$C=0$	$C=1$	$C=rand$
1	9.66×10^{-7}	3.29×10^{-4}	1.02×10^{-3}	4.03×10^{-4}
2	9.21×10^{-7}	9.40×10^{-7}	9.38×10^{-7}	9.40×10^{-7}
3	9.21×10^{-7}	9.21×10^{-7}	9.21×10^{-7}	9.21×10^{-7}
4	9.21×10^{-7}	9.21×10^{-7}	9.21×10^{-7}	9.21×10^{-7}
5	9.21×10^{-7}	9.21×10^{-7}	9.21×10^{-7}	9.21×10^{-7}

pseudorandom numbers on the open interval $(0,1)$), the approximate solution of the similar accuracy is achieved with less than three iterations on each time layer. The present method converges to the same order of accuracy for all the considered initial solutions after several iterations. This indicates the robustness of the present method. Furthermore,the computational cost versus the number of domain collocation nodes N and the boundary collocation nodes K_1 is displayed in Fig. 8 where $M=35$, $K_1=20$, K=70 and



(a) The exact solution (b) The absolute error

Figure 7: Example 4.2. The exact solution and absolute error at $T=10$ with $\Delta t=0.001$.

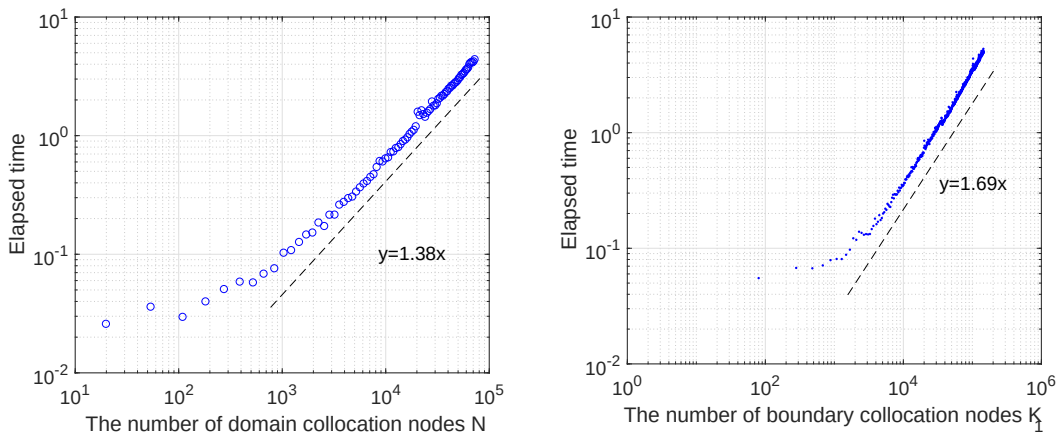


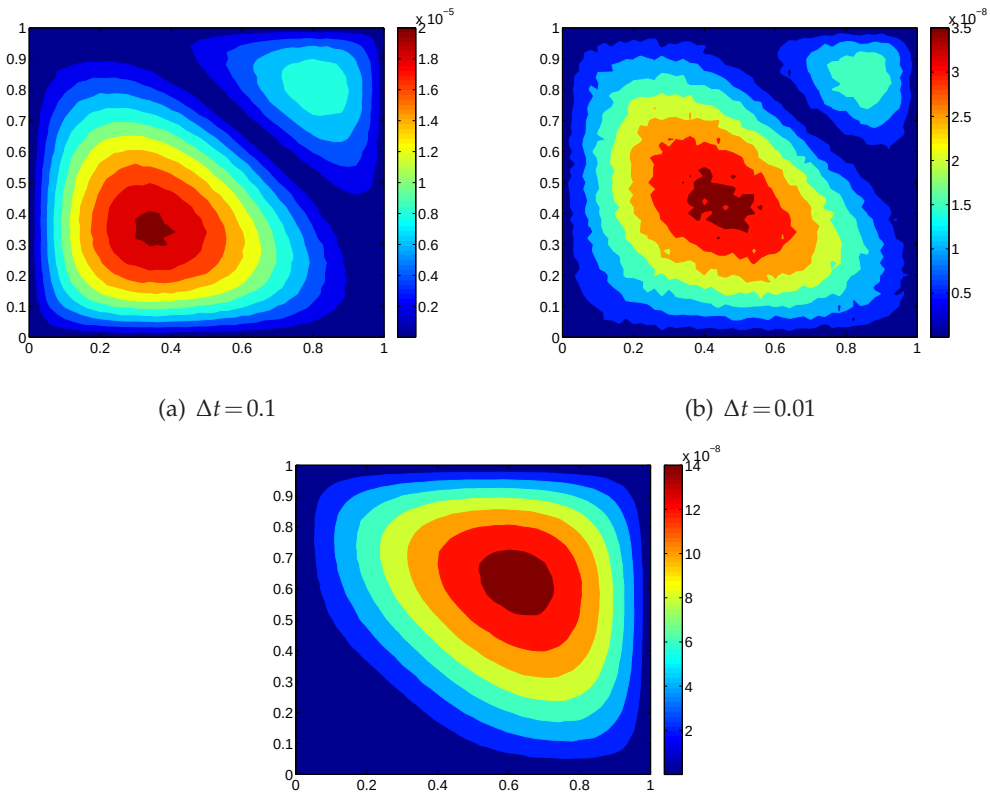
Figure 8: Example 4.2. The elapsed time versus the number of domain collocation nodes N (left) and the boundary collocation nodes K_1 (right) respectively.

$M=35$, $K=20$, N=324 are used for computations respectively. It should be noted that we only show the computational time at the first time step size using the MQ basis functions ($c=1$) with three inner iterations.

Example 4.3. In this example, we test our method when dealing with a semi-linear equation which is subject to periodic boundary conditions:

$$\begin{cases} C_t+\left(\frac{1}{2}C^2\right)_x+\left(\frac{1}{2}C^2\right)_y=C_{xx}+C_{yy}+2C+\cos(x+y+t)(1+2\sin(x+y+t)), \\ C(x,y,0)=\sin(x+y), \quad 0<x,\ y<1, \end{cases} \tag{4.6}$$

where the exact solution is $C(x,y,t)=\sin(x+y+t)$.



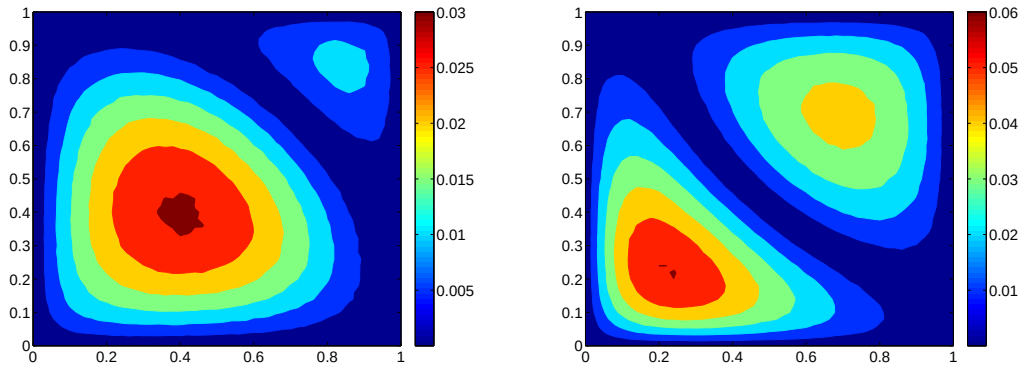
(a) $\Delta t=0.1$

(b) $\Delta t=0.01$

(c) $\Delta t=0.001$

Figure 9: Example 4.3. The absolute error at $T=1$ with $\Delta t=0.1$, $\Delta t=0.01$ and $\Delta t=0.001$.

The computation is carried up to the time $T=1$ at which the maximum absolute error is calculated. Fig. 9 displays the absolute errors of the approximate solutions corresponding to the time steps $\Delta t=0.1$, $\Delta t=0.01$, and $\Delta t=0.001$. The rest parameters are: $N=576$, $M=324$, $K=160$, $K_1=100$ and $\alpha=5$. The MQ RBF ($c=2.3$) is used. This figure shows that the error decreases from 10^{-5} to 10^{-8} as the time step decreases from $\Delta t=0.1$ to $\Delta t=0.01$. With the further diminution of Δt the error keeps the same order. This may be explained by the accumulation of rounding errors for small Δt. Note that in these calculations the step size of the spatial approximation is about 0.041. This means that the time step can be less than $\Delta t=\mathcal{O}(\Delta x)$ and this does not break the stability of the method. On the other hand, some well-known methods such as the Runge-Kutta method require the stability condition $\Delta t=\mathcal{O}(\Delta x^2)$. Therefore, applying the present method, we can use a large time step and keep the stability of the calculations. Fig. 10 displays the absolute errors at $T=100$ obtained with the time steps $\Delta t=1$ and $\Delta t=5$. It should be noted that in all the calculations shown in Fig. 9 and Fig. 10, the function $C_p(\mathbf{x})$ is used as the initial approximation $C_0^n(\mathbf{x})$ in the inner iterations and 3 iterations are applied on each time layers.



(a) $\Delta t=1$ (b) $\Delta t=5$

Figure 10: Example 4.3. The absolute errors at $T=100$ with $\Delta t=1$ and $\Delta t=5$.

Table 4: Example 4.3: The maximum absolute errors at $T=1$ versus the number of iterations using different initial solution with $\Delta t=0.01$.

Number of Iterations	$u=u_p$	$u=0$	$u=1$	$u=rand$
1	4.44×10^{-6}	2.65×10^{-2}	9.17×10^{-3}	9.96×10^{-3}
2	4.39×10^{-8}	1.03×10^{-6}	7.68×10^{-7}	2.37×10^{-7}
3	4.39×10^{-8}	3.81×10^{-7}	4.39×10^{-8}	4.39×10^{-8}
4	4.39×10^{-8}	4.39×10^{-8}	4.39×10^{-8}	4.39×10^{-8}
5	4.39×10^{-8}	4.39×10^{-8}	4.39×10^{-8}	4.39×10^{-8}

Table 4 demonstrates the maximum absolute errors versus the number of iterations using different initial approximations: $C_0^n(\mathbf{x})=C_p(\mathbf{x})$, $C_0^n(\mathbf{x})=0$, $C_0^n(\mathbf{x})=1$, and $C_0^n(\mathbf{x})=rand$. It is evident that the present method converges faster when the initial approximation $C_p(\mathbf{x})$ is applied. In this case only two inner iterations are needed for the convergence. For other initial approximations the present method converges in 3-4 inner iterations on each time layer. The computational cost versus the number of domain collocation nodes N and the boundary collocation nodes K_1 is displayed in Fig. 11 where $M=81$, $K_1=20$, K=160 and $M=81$, $K=20$, N=400 are used for computations respectively. It should be noted that we only show the computational time at the first time step size using the MQ basis functions ($c=1$) with three inner iterations.

Example 4.4. We apply the proposed method for solving the fully nonlinear ADR equation:

$$C_t+\left(\frac{1}{2}C^2\right)_x+\left(\frac{1}{2}C^2\right)_y=\nabla\cdot(C\nabla C)-C^2+f(x,y,t), \tag{4.7}$$

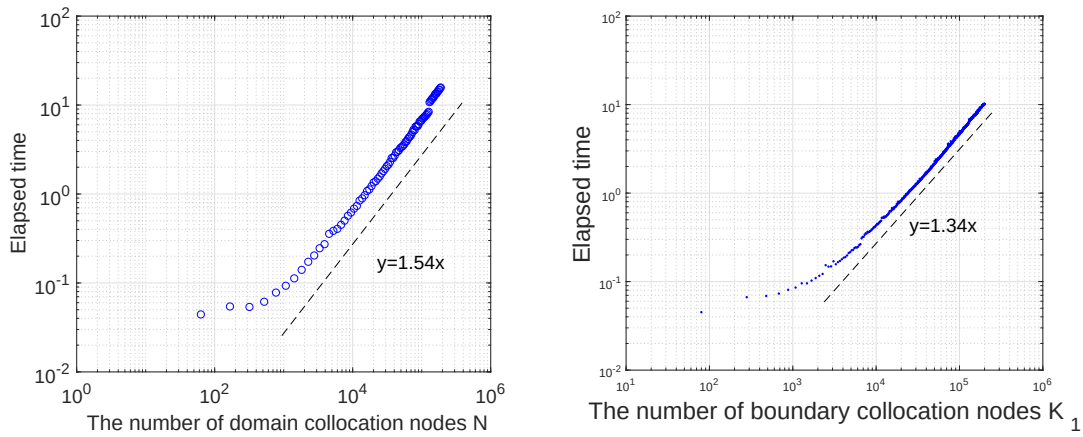


Figure 11: Example 4.3. The elapsed time versus the number of domain collocation nodes N (left) and the boundary collocation nodes K_1 (right) respectively.

in the irregular domain which is bounded by

$$\partial\Omega=\{(x_1,x_2)\,|\,x_1=\rho(s)\cos(s),x_2=\rho(s)\sin(s),0\le s\le 2\pi\}, \tag{4.8}$$

where

$$\rho(s)=\Big\{\cos(5s)+\big[18/5-\sin(5s)^2\big]^{1/2}\Big\}^{1/3}.$$

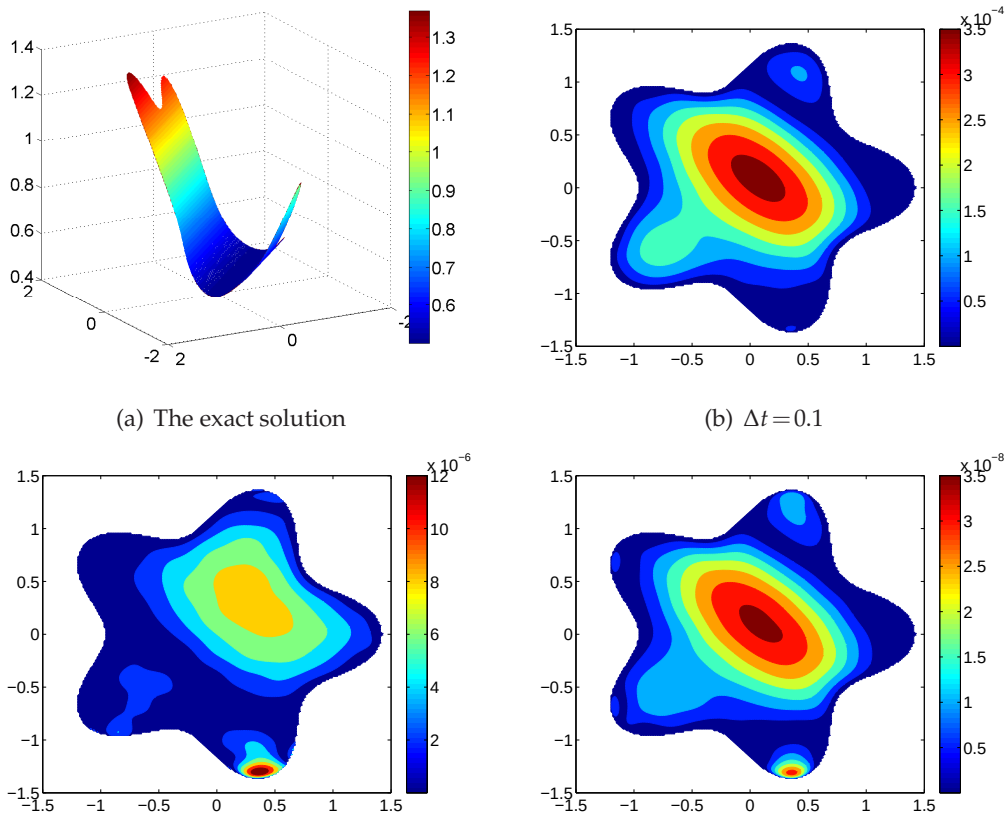
The periodic boundary condition and the initial condition correspond to the following exact solution:

$$C(x,y,t)=1+\frac{1}{2}\sin(x+y-t). \tag{4.9}$$

The source function $f(x,y,t)$ is defined correspondingly:

$$\begin{aligned} f(x,y,t)=&1.125-0.625\cos(2x+2y-2t)+0.25\sin(2x+2y-2t)\\ &+0.5\cos(x+y-t)+2\sin(x+y-t). \end{aligned} \tag{4.10}$$

Thus, in this equation all the spatial terms are nonlinear. The approximate solutions depicted in Fig. 12 are obtained at the time $T=1$ by using the present method with the following parameters: $N=256$, $M=144$, $K=120$, $K_1=100$, $\alpha=5$ and the MQ RBFs ($c=1$). Three inner iterations are used on each time layer applying the quasilinearization procedure to the nonlinear terms. Fig. 12 shows that the error of the approximate solution decreases as the second degree of the time step size. Thus, this is the error of the approximation by the Crank-Nicolson scheme which dominates in these calculations. The stability and robustness of the method provide the calculation with large time steps. This is demonstrated by the graphs in Fig. 13. The graphs show the analytical and approximate solutions at the time $T=100$. The approximate solutions are obtained using the MQ



(a) The exact solution

(b) $\Delta t=0.1$

(c) $\Delta t=0.01$

(d) $\Delta t=0.001$

Figure 12: Example 4.4. The absolute error at $T=1$ with $\Delta t=0.1$, $\Delta t=0.01$ and $\Delta t=0.001$.

RBFs ($c=1$), the GRBF ($c=1$) and CRBF with time step $\Delta t=1$. The computational cost versus the number of domain collocation nodes N and the boundary collocation nodes K_1 is displayed in Fig. 14 where $M=81$, $K_1=80$, K=20and $M=81$, $K=20$, N=184 are used for computations respectively. It should be noted that we only show the computational time at the first time step size using the MQ basis functions ($c=1$) with three inner iterations.

Example 4.5. Finally, we apply the proposed method for solving nonlinear problems Eqs. (4.7) and (4.9) in multiply-connected domain which is bounded by

$$\partial\Omega=\{(x_1,x_2)\,|\,x_1=\rho(s)\cos(s),x_2=\rho(s)\sin(s),0\le s\le 2\pi\}, \tag{4.11}$$

where

$$\rho(s)=\Big\{\cos(5s)+\big[18/5-\sin(5s)^2\big]^{1/2}\Big\}^{1/3},$$

with four holes as shown in Fig. 15.

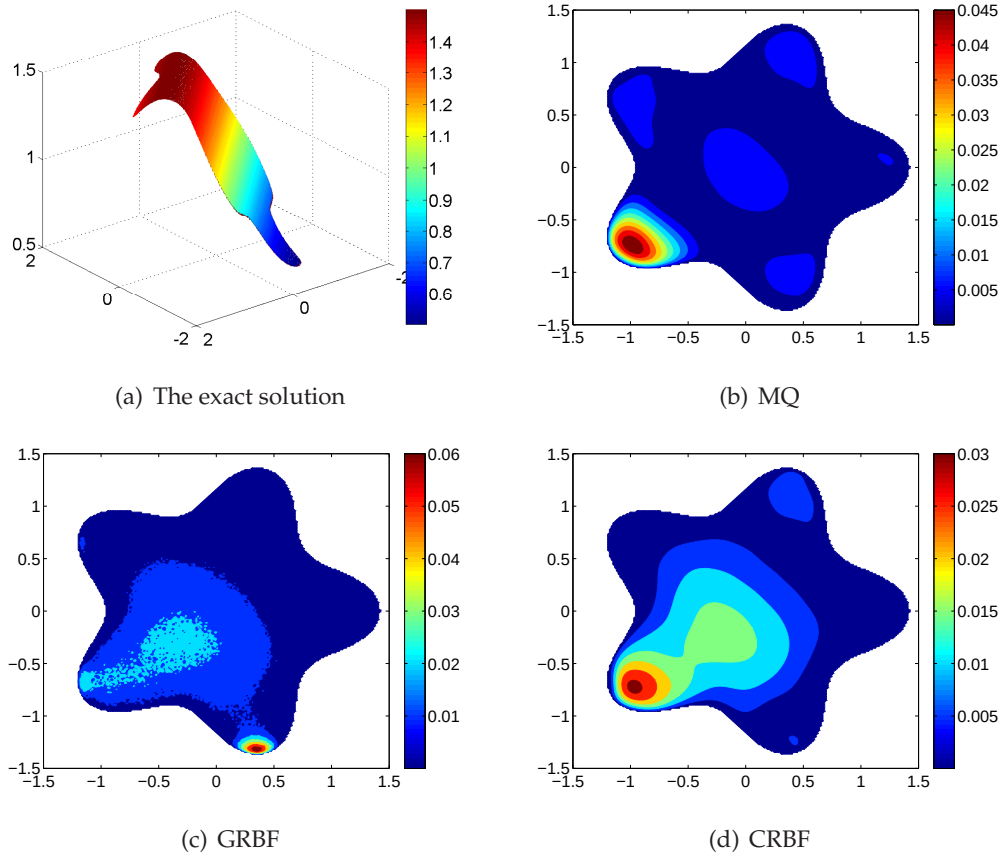


Figure 13: Example 4.4. The relative error at $T=100$ with $\Delta t=1$ using the MQ ($c=1$), GRBF ($c=1$) and CRBF basis functions.

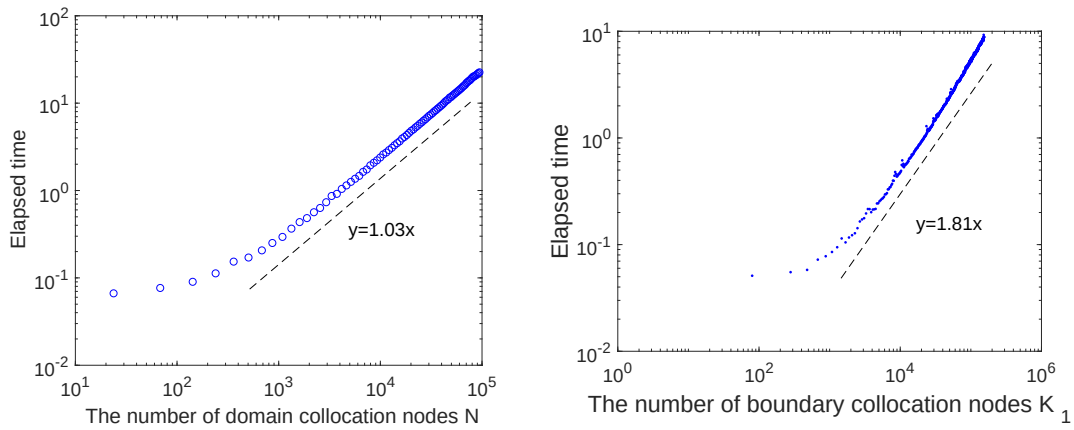


Figure 14: Example 4.4. The elapsed time versus the number of domain collocation nodes N (left) and the boundary collocation nodes K_1 (right) respectively.

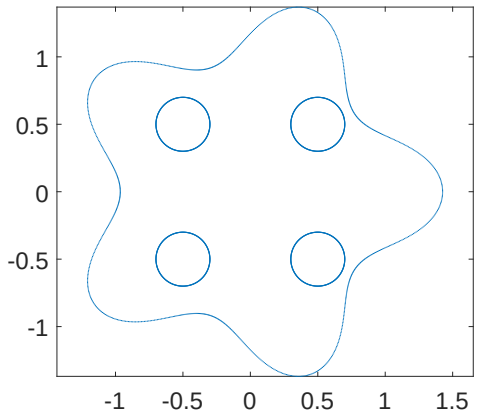
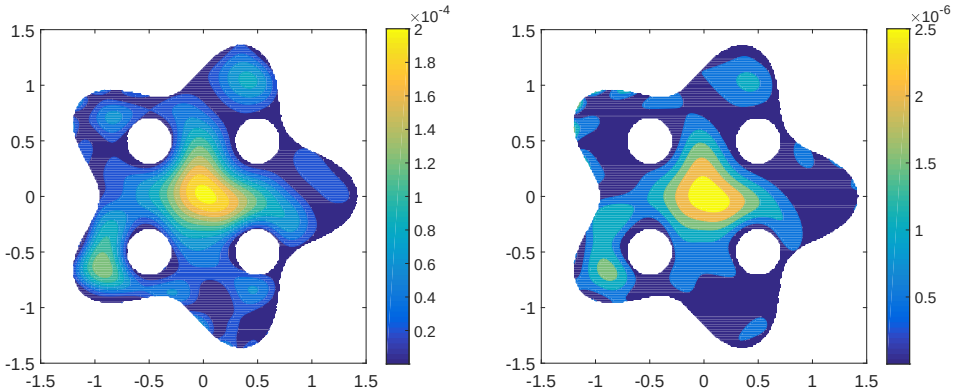


Figure 15: Example 4.5. The profile of the solution domain.

The computation is carried up to the time $T=1$ at which the domain absolute error is calculated. Fig. 16 displays the absolute errors of the approximate solutions corresponding to the time steps $\Delta t=0.1$ and $\Delta t=0.01$ using the MQ RBFs ($c=1$), the GRBF ($c=1$) and CRBF. The rest parameters are: $N=128$, $M=65$, $K=40$, $K_1=300$ and $\alpha=5$. There are 100 nodes on the boundary governed by Eq. (4.11) and 50 boundary nodes on each hole.

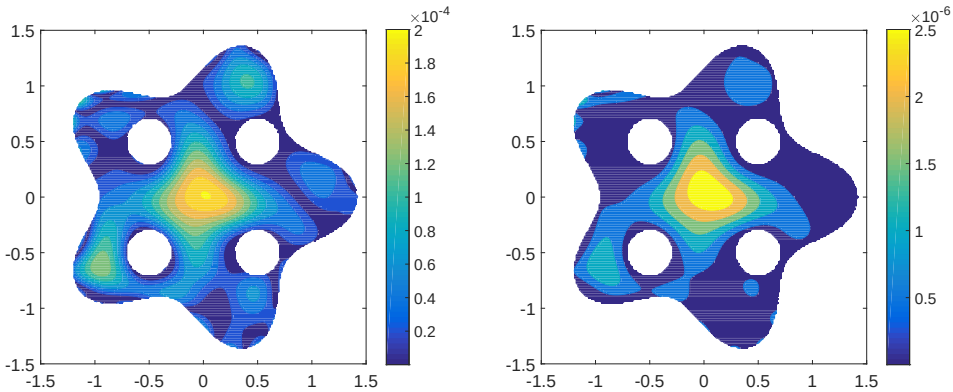
5 Conclusions

In this paper we present a novel numerical method for solving the fully nonlinear time-dependent ADR equations in arbitrary 2D domains. These equations are widely used for modeling the transfer processes in anisotropic and inhomogeneous media. So, the solution technique for solving these equations is relevant to many branches of the engineering and science. For the approximation of the time derivative in the ADR equation, we have used the Crank-Nicolson method because of its unconditional stability. As a result, we get a sequence of the stationary ADR problems. To solve the stationary ADR equation we have applied the effective meshless RBF-based technique called the BSM. We have used RBFs of three different kinds: the MQ RBF, the Gaussian RBF and the conical RBF. The key idea of the method is the use of the basis functions which satisfy the homogeneous boundary conditions of the problem. Each basis function used in the algorithm is a sum of an RBF and a special correcting function (see Eq. (3.10)) which is chosen to satisfy the homogeneous BC of the problem. This allows us to seek an approximate solution in the form which satisfies the boundary conditions of the initial problem with any choice of free parameters (see Eq. (3.13)). As a result we separate the approximation of the boundary conditions and the approximation of the ADR equation inside the solution domain. This separation provides much higher accuracy of the approximate solution when compared with other methods such as Kansa's method. In order to solve the



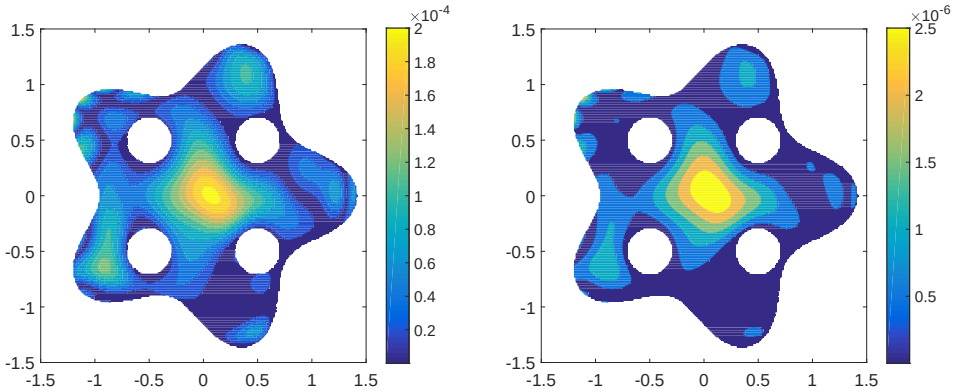
(a) MQ, $\Delta t = 0.1$

(b) MQ, $\Delta t = 0.01$



(c) GRBF, $\Delta t = 0.1$

(d) GRBF, $\Delta t = 0.01$



(e) CRBF, $\Delta t = 0.1$

(f) CRBF, $\Delta t = 0.01$

Figure 16: Example 4.5. The relative errors at $T = 1$ using the MQ ($c = 1$), GRBF ($c = 1$) and CRBF basis functions.

nonlinear ADR equation we use the well-known procedure of quasilinearization. This transforms the original equation into a sequence of linear ones at each time layer. As the numerical experiments have shown, 2-3 inner iterations are enough to reach the approximate solution on the time layer. The numerical experiments were carried out to test the accuracy, stability, convergence and robustness of the proposed method. We have compared the numerical results obtained in the paper with the exact solutions and with the data obtained by the use of other numerical techniques. The numerical results demonstrate that the present method is accurate, convergent, stable, and robust in solving ADR problems. The present method also can be extended to 3D problems. This will be the subject of further studies.

Acknowledgments

The authors thank the editor Prof. Tao Tang and anonymous reviewers for their constructive comments on the manuscript. The research of the authors was supported by the Fundamental Research Funds for the Central Universities (No. 2018B16714), the National Natural Science Foundation of China (Nos. 11702083, 11572111, 51679150, 51579153, 51739008, 51527811), the State Key Laboratory of Mechanics and Control of Mechanical Structures (Nanjing University of Aeronautics and Astronautics) (No. MCMS-0218G01), the China Postdoctoral Science Foundation (No. 2017M611669), the China Postdoctoral Science Special Foundation (No. 2018T110430), the Postdoctoral Foundation of Jiangsu Province (No. 1701059C), the National Key R&D Program of China (No. 2016YFC0401902), and the Fund Project of NHRI (Nos. Y417002, Y417015).

References

[1] Faruk, C. Porous media transport phenomena. John Wiley & Sons, Inc., Hoboken, New Jersey, 2011.
[2] Pudykiewicz, J.A. Numerical solution of the reaction-advection-diffusion equation on the sphere. Journal of Computational Physics, 213(1)(2006), 358–390.
[3] Bear, J., & Cheng, A.H.-D. Modeling Groundwater Flow and Contaminant Transport. Springer Dordrecht Heidelberg, London New York, 2010.
[4] Kurpiewski, A., & Jaroszynski, M. Radiation spectra of advection dominated accretion flows around kerr black holes. ACTA Astronomica, 50(2000) 79–91.
[5] Stocker, T. Introduction to Climate Modelling. Springer Heidelberg Dordrecht, London New York, 2011.
[6] Balsa-Canto, E., Lopez-Nunez, A., & Vzquez, C. Numerical methods for a nonlinear reaction–diffusion system modelling a batch culture of biofilm. Applied Mathematical Modelling, 41(2017) 164–179.
[7] Kazezyılmaz-Alhan, C.M., & Jr., M.A.M. On numerical modeling of the contaminant transport equations of the wetland hydrology and water quality model WETSAND. Applied Mathematical Modelling, 40(2016) 4260–4267.

[8] Chai, Y.B., Gong, Z.X., Li, W., Li, T.Y., Zhang, Q.F., Zou, Z.H., & Sun, Y.B. Application of smoothed finite element method to two-dimensional exterior problems of acoustic radiation. International Journal of Computational Methods, 15 (2018)1850029.
[9] Tadmor, E. A review of numerical methods for nonlinear partial differential equations. Bulletin of the American Mathematical Society, 49(2012) 507–554.
[10] Mohebbi, A., & Dehghan, M. High-order compact solution of the one-dimensional heat and advection–diffusion equations. Applied Mathematical Modelling, 34(2010) 3071–3084.
[11] Nishikawa, H. First, second, and third order finite-volume schemes for advection–diffusion, Journal of Computational Physics, 273(2014) 287–309.
[12] Hesthaven, J.S., Gottlieb, S., & Gottlieb, D. Spectral methods for time-dependent problems. Cambridge University Press, Cambridge, 2007.
[13] Li, X.L. Three-dimensional complex variable element-free Galerkin method. Applied Mathematical Modelling, 63(2018) 148–171.
[14] Lin, J., Chen, W., & Wang, F.Z. A new investigation into regularization techniques for the method of fundamental solutions. Mathematics and Computers in Simulation, 81(2011) 1144–1152.
[15] Fu, Z.J., Xi, Q., Chen, W., & Cheng, A.H.D. A boundary-type meshless solver for transient heat conduction analysis of slender functionally graded materials with exponential variations. Computers & Mathematics with Applications, 76(4)(2018) 760–773
[16] Gu, Y., He, X., Chen, W., & Zhang, C. Analysis of three-dimensional anisotropic heat conduction problems on thin domains using an advanced boundary element method. Computers & Mathematics with Applications, (2018)(75) 33–44.
[17] Li, J.P., & Chen, W. A modified singular boundary method for three-dimensional high frequency acoustic wave problems. Applied Mathematical Modelling, 54(2018) 189–201
[18] Lin, J., Zhang, C.Z., Sun, L.L., & Lu, J. Simulation of seismic wave scattering by embedded cavities in an elastic half-plane using the novel singular boundary method. Advances in Applied Mathematics and Mechanics, 10(2)(2018) 322–342.
[19] Lin, J., Chen, C.S., Wang, F.J., & Dangal, T. Method of particular solutions using polynomial basis functions for the simulation of plate bending vibration problems. Applied Mathematical Modelling, 49(2017) 452–469.
[20] Gharib, M., Khezri, M., & Foster, S.J. Meshless and analytical solutions to the time-dependent advection-diffusion-reaction equation with variable coefficients and boundary conditions. Applied Mathematical Modelling, 49(2017) 220–242.
[21] Lin, J., Chen, C.S., Liu, C.S., & Lu, J. Fast simulation of multi-dimensional wave problems by the sparse scheme of the method of fundamental solutions. Computers and Mathematics with Applications, 72(3)(2016) 555–567.
[22] Kansa, E.J. Multiquadrics—a scattered data approximation scheme with applications to computational fluid-dynamics. I. surface approximation and partial dervative estimates. Computers & Mathematics with Applications, 19(8–9)(1990) 127–145.
[23] Kansa, E.J. Multiquadrics—a scattered data approximation scheme with applications to computational fluid-dynamics—ii solutions to parabolic, hyperbolic and elliptic partial differential equations. Computers & Mathematics with Applications, 19(8)(1990) 147–161.
[24] Dehghan, M., & Mohammadi, V. The numerical solution of Fokker-Planck equation with radial basis functions (RBFs) based on the meshless technique of Kansa's approach and Galerkin method. Engineering Analysis with Boundary Elements, 47(1)(2014) 38–63.
[25] Boztosun, I., Charafi, A., & Boztosun, D. On the numerical solution of linear advection-diffusion equation using compactly supported radial basis functions in meshfree methods

for partial differential equations. in: Meshfree Methods for Partial Differential Equations M. Griebel, M.A. Sehweitzer (Editors), Springer Berlin Heidelberg, 2003:63–73.

[26] Sarra, S.A. A local radial basis function method for advection-diffusion-reaction equations on complexly shaped domains. Applied Mathematics & Computation, 218(19)(2012) 9853–9865.

[27] Varun, S., Wright, G.B., Fogelson, A.L., & Kirby, R.M. A radial basis function (RBF) finite difference method for the simulation of reaction-diffusion equations on stationary platelets within the augmented forcing method. International Journal for Numerical Methods in Fluids, 75(1)(2014) 1–22.

[28] Boztosun, I., Charafi, A., Zerroukat, M., & Djidjeli, K. Thin-plate spline radial basis function scheme for advection-diffusion problems. Electronic Journal of Boundary Elements, BETEQ 2001(2002) 267–282.

[29] Reutskiy, S.Y., & Lin, J. A semi-analytic collocation technique for steady-state strongly nonlinear advection-diffusion-reaction equations with variable coefficients. International Journal for Numerical Methods in Engineering, 112(2017) 2004–2024.

[30] Reutskiy, S.Y., & Lin, J. A meshless radial basis function method for steady-state advection-diffusion-reaction equation in arbitrary 2d domains. Engineering Analysis with Boundary Elements, 79(2017) 49–61.

[31] Stevens, D., & Power, H. The radial basis function finite collocation approach for capturing sharp fronts in time dependent advection problems. Journal of Computational Physics, 298(2015) 423–445.

[32] Askari, M., & Adibi, H., Numerical solution of advection-diffusion equation using meshless method of lines. Journal of Thermal Analysis & Calorimetry, 124(2)(2017) 1–8.

[33] Dehghan, M., & Shirzadi, M. Meshless simulation of stochastic advection-diffusion equations based on radial basis functions. Engineering Analysis with Boundary Elements, 53(1)(2015) 18–26.

[34] Reutskiy, S.Y. The backward substitution method for multipoint problems with linear Volterra-Fredholm integro-differential equations of the neutral type. Journal of Computational and Applied Mathematics, 296(2016) 724–738.

[35] Reutskiy, S.Y. A meshless radial basis function method for 2D steady-state heat conduction problems in anisotropic and inhomogeneous media. Engineering Analysis with Boundary Elements, 66(2016) 1–11.

[36] Reutskiy, S.Y. A new semi-analytical collocation method for solving multi-term fractional partial differential equations with time variable coefficients. Applied Mathematical Modelling, 45(2017) 238–254.

[37] Lin, J., Reutskiy, S.Y., & Lu, J. A novel meshless method for fully nonlinear advection-diffusion-reaction problems to model transfer in anisotropic media, Applied Mathematics and Computation, 339(2018) 459–476.

[38] Lin, J., He, Y.X., Reutskiy, S.Y., & Lu, J. An effective semi-analytical method for solving telegraph equation with variable coefficients. The European Physical Journal Plus, 133(2018) 290.

[39] Mandelzweig, V.B., & Tabakin, F. Quasilinearization approach to nonlinear problems in physics with application to nonlinear odes. Computer Physics Communications, 141(2)(2001) 268–281.

[40] Abbasbandy, S. Soliton solutions for the Fitzhugh–Nagumo equation with the homotopy analysis method. Applied Mathematical Modelling, 32(2008) 2706–2714.

[41] Zhao, T., Li, C., Zang, Z.,& Wu, Y. Chebyshev–Legendre pseudo-spectral method for the

generalized Burgers–Fisher equation. Applied Mathematical Modelling, 36(2012) 1046–1056.
[42] Mitchell, A.R., & Griffiths, D.F. The finite difference methods in partial differential equations. Wiley; 1980.
[43] Sarra, S.A., & Sturgill, D., A random variable shape parameter strategy for radial basis function approximation methods. Engineering Analysis with Boundary Elements, 33(2009) 1239–1245.
[44] Fok, J., Guo, B., & Tang, T. Combined Hermite spectral-finite difference method for the Fokker-Planck equation. Mathematics of Computation, 71(240)(2002) 1497–1528.