IJASST
International Journal of
Applied Sciences and Smart Technologies

Volume 07
Issue 02

p-ISSN 2655-8564, e-ISSN 2685-943, pp. 309-344, 2025

Numerical Solution of Two-Dimensional Advection Diffusion Equation for Multiphase Flows in Porous Media Using a Novel Meshfree Method of Lines

F.O. Ogunfiditimi[1] and J.A. Kazeem[1*]

[1] *Department of Mathematics, University of Abuja, Federal Capital Territory, Abuja, Nigeria.*
[*] *Corresponding Author E-mail: jamiu.kazeem@yahoo.com*

(Received 25-05-2025; Revised 23-06-2025; Accepted 08-07-2025)

Abstract.

This study proposes a novel Meshfree Method of Lines (MFMOL), in strong form formulation, to solve multiphase flows of solute transport modelled by two-dimensional (2D) Advection Diffusion Equations (ADE). The method uses a consistent and stable Augmented Radial Basis Point Interpolation Method (ARPIM) for spatial variables discretization of the models, while the time variable is left continuous, resulting in a system of Ordinary Differential Equations (ODEs) with initial conditions, which is solved numerically, via Matlab ode solver. The new method is proposed to overcome the challenges of numerical instabilities and large deformation due to complex domain, and distorted or low-quality meshes that attracts remeshing, all encountered by traditional Finite Element Method (FEM), Finite Difference Method (FDM) and Finite Volume Method (FVM). Also, the MFMOL is used in strong form formulation without any stabilization techniques for the convective terms in solute transport models, contrary to other methods like FEM, FDM, FVM and meshfree Finite Point Method (FPM) that require the stabilization techniques for fluid flow problems to guarantee acceptable results. The efficiency and accuracy of the new method were established and validated by using it to solve 2D diffusive and advective flow problems in the complex domain of porous structures. The results obtained agreed with the existing exact solutions, using less computational efforts, costs and time, compared with mesh-based methods and others that require stabilization for the convective terms. These features established the superior performance of the new method to the mesh-based methods and others that require special stabilization techniques for solving 2D multiphase flow of solute transport in porous media and other transient fluid flow problems.

Keywords: Advection Diffusion Equations (ADE), Augmented Radial Basis Point Interpolation Method (ARPIM), Meshfree Method of Lines (MFMOL), Multiphase Flows, Porous Media.

This work is licensed under a Creative Commons Attribution 4.0 International License

1 Introduction

Considering the complexity of numerous practical engineering and industrial problems modelled by partial differential equations (PDEs), numerical methods have become alternatives as analytical solutions are not obtainable [1]. For boundary value problems in complex domains, Finite Element Method (FEM) which is mesh-based had been a versatile and robust computational tool to get the approximate solutions [2]. The other commonly used mesh-based methods are the Boundary Element Method (BEM) which minimizes the dimension of the problem and has higher accuracy compared to FEM, Finite Difference Method (FDM) which is based on regular nodes for discretization, Finite Volume Methods (FVM) and others [1], [3], [4], [5]. However, the mesh-based methods have some limitations. These include instabilities and large deformation due to mesh-based interpolation for the complex domain, distorted or low-quality meshes, leading to higher errors, which necessitate remeshing at the expense of additional computational efforts, costs and time [6]. To overcome these limitations associated with the reliance on meshes to construct the approximating functions, meshfree methods with great adaptability for discontinuities, and large deformation of the complex domain geometry were introduced [7]. Over some years, various authors had studied different meshfree methods: the Radial Basis Function Method of Lines [8], the Element Free Galerkin (EFG) method [9], Hybrid Interpolating Meshless Method [10], the Diffuse Element Method (DEM) [11], the Local Radiant Point Interpolation Method (LRPIM) [12], the Improved Element Free Galerkin Method [13], Fracture Mapping [14], Meshless Method of Lines (MMOL) [15].

Meshfree methods have recorded quite considerable success in engineering and industrial applications. However, each of them has its associated strengths, weaknesses and conditions of applicability [2]. Meshfree methods with moving least squares (MLS) shape functions such as the DEM, EFG, Meshless local Petrov–Galerkin (MLPG) method [16], Finite Point Method [17] and others, sometimes experience singular moment matrices which break down the entire method. Using a large number of nodal points to control the singularity may lead to a reduction in the sparsity of the moment matrix of the method, and this affects

the computational efficiency of the method. Also, using fewer nodal points to avoid the singularity of the moment matrix may also cause incompatibility or continuity problems, especially when Galerkin's weak form is used. However, the singularity of the moment matrix can be controlled using the T2L scheme for node selection at an additional computational time [18]. In addition, meshfree methods with MLS shape functions and those with integral representation shape functions such as the smoothed particle hydrodynamics (SPH) method [19], and the Reproducing kernel particle methods [20], do not possess the Kronecker delta function property and this leads to the imposition of essential boundary conditions to the models using direct interpolation method at additional expense [18]. In getting rid of these difficulties, [21] introduced a more efficient Radial Point Interpolation Method (RPIM) that uses a few arbitrarily scattered nodal points, with the guaranteed existence of a nonsingular moment matrix, subject to avoidance of some specific shape parameters [22], [23], [24]. However, RPIM cannot reproduce linear polynomials, as the pure radial function fails the standard patch test to check its performance, as widely used to check the performances of standard methods like FEM. This affects the consistency of the RPIM [18]. To restore the consistency issue, [21], [22] introduced the addition of polynomial terms to RPIM.

Multiphase flows of heterogeneous mixtures of two or more phases, such as solid-liquid, gas-liquid or gas-solid, are encountered in numerous industrial and scientific applications which include: gas and petroleum flows, aerosol flows in the environment, boiling and condensation processes, gas-solid and slurry flow in pipelines, particle and fiber flows in airways, nanofluids, fluidized bed reactors and others [25]. In multiphase flows, the advection-diffusion equation (ADE) is used to model the transport of properties such as temperature, salinity, or the concentration of a solute across distinct phases and interfaces. Modelling multiphase flows in porous media with advection-diffusion equations can be challenging due to complex domain geometries, which makes mesh-based methods ineffective for simulations due to their various limitations [18]. Meshfree methods in strong forms are known to be direct, fast and efficient in the simulation of multiphase flow of solute

transport, but less stable compared to the weak forms, which are more stable but with higher computational procedures and time [18], [26]. In literature, special attention had been given to the stabilization of convective terms and the Neumann boundary conditions in transport models to ensure accuracy of the results [27]. To achieve the effective stability of the computational methods whether in weak form formulations or strong form formulations, some stabilization techniques such as anisotropic balancing diffusion, finite increment calculus, petrov-Galerkin functions weighting, adaptive meshing or remeshing, operator splitting, characteristic time integration, upwind finite difference derivatives and others, had been used by various authors to get acceptable results [27], [28], [29], [30].

In this paper, a new method MFMOL, is proposed to solve 2D multiphase flow of solute transport in porous media, using strong form formulations without the use of any stabilization technique. In this proposed method, an Augmented Radial Basis Point Interpolation Method (ARPIM) is used in discretizing the space variables of the models and the subsidiary conditions to reduce the PDEs to a system of ODEs, subject to the required initial conditions. Then, the resulting ODEs are numerically integrated in time with Matlab ode 45 solvers. The stability of the proposed method is rooted in the consistency of the ARPIM, and its effective accuracy for function fitting, for discretizing the spatial variable of the flow models. In addition, the proposed method possesses a guaranteed nonsingular moment matrix and satisfies the Kronecker delta function property for easy imposition of essential boundary conditions, and these make the new method a potential computational technique for solving multiphase flows of solute transport in complex domains. The new method is validated by applying it to solve 2D diffusive and advective flow problems in porous media, using strong form formulations without any stabilization for the convective terms. The results obtained agreed with the existing exact solutions, with less computational efforts and time, and enhanced stability, compared with mesh-based methods and others, that require a special stabilization technique for solving 2D multiphase flow of solute transport in porous media and other transient fluid flow problems. The features of the new MFMOL

establish its superior performance over the mesh-based methods and others for the solution of 2D multiphase flow of solute transport in porous media and transient fluid flow problems.

2 Material and Method

The governing equation describing two-dimensional solute transport in a nonreactive, source-free, homogeneous and isotropic porous medium is given by the two-dimensional Advection Diffusion Equation (ADE) [31]

$$\frac{\partial u(x,y,t)}{\partial t} + v_1 \frac{\partial u(x,y,t)}{\partial x} + v_2 \frac{\partial u(x,y,t)}{\partial y} = D_1 \frac{\partial^2 u(x,y,t)}{\partial x^2} + D_2 \frac{\partial^2 u(x,y,t)}{\partial y^2} \quad (1)$$

$$(x, y, t) \in [0, L_1] \times [0, L_2] \times [0, T]$$

$$u(x, y, 0) = \eta(x, y) \qquad (x, y) \in [0, L_1] \times [0, L_2] \quad (2)$$

$$u(0, y, t) = \beta_1(y, t), \quad u(L_1, y, t) = \gamma_1(y, t) \qquad t \in [0, T] \quad (3)$$

$$u(x, 0, t) = \beta_2(x, t), \quad u(x, L_2, t) = \gamma_2(x, t) \qquad t \in [0, T] \quad (4)$$

where $u(x, y, t)$ $[ML^{-3}]$ is the flux averaged (flowing) solute concentration at position (x, y) and time t. $v_1[LT^{-1}]$ is the average linear velocity in the direction of x coordinate, $v_2[LT^{-1}]$ is the average linear velocity in the direction of y coordinate, $D_1[L^2T^{-1}]$ is the dispersion coefficient in the direction of x coordinate, $D_2[L^2T^{-1}]$ is the dispersion coefficient in the direction of y coordinate, $t[T]$ is the time and $x[L]$ and $y[L]$ are the spatial coordinate. $\eta(x, y)$, $\beta_1(y, t)$, $\beta_2(x, t)$, $\gamma_1(y, t)$ and $\gamma_2(x, t)$ are known functions.

To use the Meshfree Method of Lines (MFMOL) method for the numerical solution of equations (1) – (4), the following procedures are followed:

Let $\Omega^2 = [0, L_1] \times [0, L_2]$ be the complex domain represented by the nodal collocation points $[\boldsymbol{x}_i]_{i=1}^n$, where $\boldsymbol{x}_i = (x_i, y_i), i = 2, 3 \ldots n-1$ are the interior nodes and, $\boldsymbol{x}_i$, $i = 1$ and n are the boundary nodes. For a field variable $u(\boldsymbol{x}, t)$ defined in the complex domain Ω^2, the

augmented Radial Basis Functions Point Interpolation Approximation (ARPIM) at the star node $\boldsymbol{x}_\alpha \in \Omega^2$, takes the form [18]:

$$u(\boldsymbol{x}_\alpha, t) = \boldsymbol{\phi}^T(\boldsymbol{x}_\alpha)\boldsymbol{u} = \sum_{i=1}^{n} \phi_i(\boldsymbol{x}_\alpha)\, u_i \qquad \alpha = 1,2, \dots n \tag{5}$$

$$= [\phi_1(\boldsymbol{x}_\alpha)\ \phi_2(\boldsymbol{x}_\alpha) \dots\ \phi_k(\boldsymbol{x}_\alpha) \dots\ \phi_n(\boldsymbol{x}_\alpha)]\ [u_1\ u_2\ u_k \dots u_n]^T \tag{6}$$

Where

$$\phi_k(\boldsymbol{x}_\alpha) = \sum_{i=1}^{n} R_i\,(\boldsymbol{x}_\alpha)A_{ik} + \sum_{j=1}^{m} p_j\,(\boldsymbol{x}_\alpha)B_{jk},\ \ \alpha, k = 1,2, \dots n \tag{7}$$

is the shape function for the kth node in the local support domain of $\boldsymbol{x}_\alpha$. n and m are the numbers of nodes and the polynomial basis, respectively, and $\boldsymbol{u} = [u_1\ u_2\ u_k \dots u_n]^T$ is the vector of all the field nodal variables at the n local nodes.

The shape function $\phi_k(\boldsymbol{x}_\alpha)$ for the kth node in the local support domain of $\boldsymbol{x}_\alpha$ is expressed as nonsingular radial basis function point interpolation (RPIM) terms: $\sum_{i=1}^{n} R_i\,(\boldsymbol{x}_\alpha)A_{ik}$, augmented by polynomial basis terms: $\sum_{j=1}^{m} p_j\,(\boldsymbol{x}_\alpha)B_{jk}$, which restores the consistency and stability of the RPIM terms. A_{ik} and B_{jk} are the coefficients of the radial basis $R_i(\boldsymbol{x}_\alpha)$ and the polynomial basis $p_j(\boldsymbol{x}_\alpha)$, respectively.

Explicitly, equation (7) is expressed as:

$$\phi_k(\boldsymbol{x}_\alpha) = \begin{pmatrix} R_1(\boldsymbol{r}_{11}) & R_2(\boldsymbol{r}_{12}) \cdots & R_n(\boldsymbol{r}_{1n}) \\ R_1(\boldsymbol{r}_{21}) & R_2(\boldsymbol{r}_{22}) \cdots & R_n(\boldsymbol{r}_{2n}) \\ \vdots & \vdots \quad \cdots & \vdots \\ R_1(\boldsymbol{r}_{n1}) & R_2(\boldsymbol{r}_{n2}) \cdots & R_n(\boldsymbol{r}_{nn}) \end{pmatrix} \begin{pmatrix} A_{1k} \\ A_{2k} \\ \vdots \\ A_{nk} \end{pmatrix} + \begin{pmatrix} p_1(\boldsymbol{x}_1) & p_2(\boldsymbol{x}_1) & \cdots p_m(\boldsymbol{x}_1) \\ p_1(\boldsymbol{x}_2) & p_2(\boldsymbol{x}_2) & \cdots p_m(\boldsymbol{x}_2) \\ \vdots & \vdots & \cdots \ \vdots \\ p_1(\boldsymbol{x}_n) & p_2(\boldsymbol{x}_n) & \cdots p_m(\boldsymbol{x}_n) \end{pmatrix} \begin{pmatrix} B_{1k} \\ B_{2k} \\ \vdots \\ B_{mk} \end{pmatrix}$$

(8)

Or

$$\phi_k(\boldsymbol{x}_\alpha) = \boldsymbol{R}A_{ik} + \boldsymbol{P}B_{jk}\ , i = 1,2, \dots n \text{ and } j = 1,2, \dots m \tag{9}$$

where $R_i(\boldsymbol{r}_{kj})$ is the multiquadric radial basis function, with optimal shape parameters: $c = 1.42$ and $q = 1.03$ [21] given as:

$$R_i(\boldsymbol{r}_{kj}) = \left(\boldsymbol{r}_{kj}^2 + c^2\right)^q = \left(\boldsymbol{r}_{kj}^2 + 1.42^2\right)^{1.03} \quad k, j = 1,2, \dots n \tag{10}$$

The radial distance $\boldsymbol{r}_{kj}$ of a fixed star node $\boldsymbol{x}_k = (x_k, y_k)$ from other nodes $\boldsymbol{x}_j = \left(x_j, y_j\right)$ arbitrarily distributed in the complex domain Ω^2 and its boundary, is expressed by the Euclidean norm as [18]:

$$\boldsymbol{r}_{kj} = \|D_k - D_j\| = \sqrt{\left(x_k - x_j\right)^2 + \left(y_k - y_j\right)^2}, \; D_k = (x_k, y_k), \; D_j = (x_j, y_j). \tag{11}$$

The moment matrix $\boldsymbol{P}$ for the polynomial basis $p_j(\boldsymbol{x}) = [p_1(x) \; p_2(x) \; p_3(x) \dots \; p_m(x)]$, where m is the number of monomials in $p_j(\boldsymbol{x})$ is expressed as [18]:

$$\boldsymbol{P} = \begin{pmatrix} p_1(x_1) & p_2(x_1) & \cdots & p_m(x_1) \\ p_1(x_2) & p_2(x_2) & \cdots & p_m(x_2) \\ \vdots & \vdots & \cdots & \vdots \\ p_1(x_n) & p_2(x_n) & \cdots & p_m(x_n) \end{pmatrix} = \begin{bmatrix} 1 & x_1 & y_1 & \dots & x_1^{m-1} y_1^{m-1} \\ 1 & x_2 & y_2 & \dots & x_2^{m-1} y_2^{m-1} \\ \vdots & \vdots & \vdots & & \ddots \; \vdots \\ 1 & x_n & y_n & \dots & x_n^{m-1} y_n^{m-1} \end{bmatrix} \tag{12}$$

where m and n are, respectively, the number of polynomial basis and the number of nodal points.

In equation (7), the constant matrices A_{ik} and B_{jk} are the $(i,k)th$ and $(j,k)th$ elements of matrices $\boldsymbol{K}_a$ and $\boldsymbol{K}_b$, respectively, which are given as [18]:

$$\boldsymbol{K}_a = \boldsymbol{R}^{-1} - \boldsymbol{R}^{-1}\boldsymbol{P}\boldsymbol{K}_b \quad \text{and} \quad \boldsymbol{K}_b = [\boldsymbol{P}^T\boldsymbol{R}^{-1}\boldsymbol{P}]^{-1}\boldsymbol{P}^T\boldsymbol{R}^{-1} \tag{13}$$

Putting equation (5) into equations (1) – (4) gives the strong form formulations of the models as:

$$\frac{du_i}{dt} + v_1 \frac{\partial}{\partial x}\left(\sum_{i=1}^{n} \phi_i(\boldsymbol{x}_\alpha) u_i\right) + v_2 \frac{\partial}{\partial y}\left(\sum_{i=1}^{n} \phi_i(\boldsymbol{x}_\alpha) u_i\right) =$$

$$D_1 \frac{\partial^2}{\partial x^2}\left(\sum_{i=1}^{n} \phi_i(\boldsymbol{x}_\alpha) u_i\right) + D_2 \frac{\partial^2}{\partial y^2}\left(\sum_{i=1}^{n} \phi_i(\boldsymbol{x}_\alpha) u_i\right)$$

(14)

$$u(x_\alpha, y_\alpha, 0) = \eta(x_\alpha, y_\alpha) \quad \alpha = 1,2, \dots n \tag{15}$$

$$u(0, y_1, t_1) = \sum_{i=1}^{n} \phi_i(\boldsymbol{x}_1)\, u_i = \beta_1(y_1, t_1), \quad u(L_1, y_n, t_n) = \sum_{i=1}^{n} \phi_i(\boldsymbol{x}_n)\, u_i = \gamma_1(y_n, T)$$

(16)

$$u(x_1, 0, t_1) = \sum_{i=1}^{n} \phi_i(\boldsymbol{x}_1)\, u_i = \beta_2(x_1, t_1), \quad u(x_n, L_2, t_n) = \sum_{i=1}^{n} \phi_i(\boldsymbol{x}_n)\, u_i = \gamma_2(x_n, T)$$

(17)

Using equations (5) and (7) in equation (14), as well as, in equations (16) - (17) gives:

$$\frac{du_i}{dt} + v_1\left(\sum_{i=1}^{n} \frac{\partial R_i(\boldsymbol{x}_\alpha)}{\partial x} A_{ik} + \sum_{j=1}^{m} \frac{\partial p_i(\boldsymbol{x}_\alpha)}{\partial x} B_{jk}\right)\boldsymbol{u} + v_2\left(\sum_{i=1}^{n} \frac{\partial R_i(\boldsymbol{x}_\alpha)}{\partial y} A_{ik} + \sum_{j=1}^{m} \frac{\partial p_i(\boldsymbol{x}_\alpha)}{\partial y} B_{jk}\right)\boldsymbol{u}$$

$$= D_1\left(\sum_{i=1}^{n} \frac{\partial^2 R_i(\boldsymbol{x}_\alpha)}{\partial x^2} A_{ik} + \sum_{j=1}^{m} \frac{\partial^2 p_i(\boldsymbol{x}_\alpha)}{\partial x^2} B_{jk}\right)\boldsymbol{u} + D_2\left(\sum_{i=1}^{n} \frac{\partial^2 R_i(\boldsymbol{x}_\alpha)}{\partial y^2} A_{ik} + \sum_{j=1}^{m} \frac{\partial^2 p_i(\boldsymbol{x}_\alpha)}{\partial y^2} B_{jk}\right)\boldsymbol{u}$$

(18)

$$\begin{bmatrix} g_1 \\ g_n \end{bmatrix} = \begin{bmatrix} \tilde{u}(0, y_1, t) \\ \tilde{u}(L_1, x_n, T) \end{bmatrix} = \begin{bmatrix} \phi_1(\boldsymbol{x}_1) & \phi_2(\boldsymbol{x}_1) & \dots & \phi_n(\boldsymbol{x}_1) \\ \phi_1(\boldsymbol{x}_n) & \phi_2(\boldsymbol{x}_n) & \dots & \phi_n(\boldsymbol{x}_n) \end{bmatrix}\boldsymbol{u} = \begin{bmatrix} \beta_1(y_1, t_1) \\ \gamma_1(y_n, T) \end{bmatrix}$$

(19)

$$\begin{bmatrix} h_1 \\ h_n \end{bmatrix} = \begin{bmatrix} \tilde{u}(x_1, 0, t) \\ \tilde{u}(x_n, L_2, T) \end{bmatrix} = \begin{bmatrix} \phi_1(\boldsymbol{x}_1) & \phi_2(\boldsymbol{x}_1) & \dots & \phi_n(\boldsymbol{x}_1) \\ \phi_1(\boldsymbol{x}_n) & \phi_2(\boldsymbol{x}_n) & \dots & \phi_n(\boldsymbol{x}_n) \end{bmatrix}\boldsymbol{u} = \begin{bmatrix} \beta_2(x_1, t_1) \\ \gamma_2(x_n, T) \end{bmatrix}$$

(20)

$$u(x_\alpha, y_\alpha, 0) = \eta(x_\alpha, y_\alpha) \tag{21}$$

where $\boldsymbol{u} = [u_1 \ u_2 \dots u_n]^T$ and $\alpha, k = 1,2, \dots n$ (22)

Using equations (5) - (8) in equation (18) gives the Augmented Radial Basis Function Point Interpolation Method (ARPIM) approximation for the space variable of the field variable $u(\boldsymbol{x}, t)$ as:

$$\frac{du_i}{dt} + v_1 \begin{pmatrix} \phi_{1x}(\boldsymbol{x}_1) & \phi_{2x}(\boldsymbol{x}_1) & \cdots & \phi_{nx}(\boldsymbol{x}_1) \\ \phi_{1x}(\boldsymbol{x}_2) & \phi_{2x}(\boldsymbol{x}_2) & \cdots & \phi_{nx}(\boldsymbol{x}_2) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{1x}(\boldsymbol{x}_n) & \phi_{2x}(\boldsymbol{x}_n) & \cdots & \phi_{nx}(\boldsymbol{x}_n) \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} +$$

$$v_1 \begin{pmatrix} \phi_{1y}(\boldsymbol{x}_1) & \phi_{2y}(\boldsymbol{x}_1) & \ldots & \phi_{ny}(\boldsymbol{x}_1) \\ \phi_{1y}(\boldsymbol{x}_2) & \phi_{2y}(\boldsymbol{x}_2) & \cdots & \phi_{ny}(\boldsymbol{x}_2) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{1y}(\boldsymbol{x}_n) & \phi_{2y}(\boldsymbol{x}_n) & \cdots & \phi_{ny}(\boldsymbol{x}_n) \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} = D_1 \begin{pmatrix} \phi_{1xx}(\boldsymbol{x}_1) & \phi_{2xx}(\boldsymbol{x}_1) & \cdots & \phi_{nxx}(\boldsymbol{x}_1) \\ \phi_{1xx}(\boldsymbol{x}_2) & \phi_{2xx}(\boldsymbol{x}_2) & \cdots & \phi_{nxx}(\boldsymbol{x}_2) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{1xx}(\boldsymbol{x}_n) & \phi_{2xx}(\boldsymbol{x}_n) & \cdots & \phi_{nxx}(\boldsymbol{x}_n) \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} +$$

$$D_2 \begin{pmatrix} \phi_{1yy}(\boldsymbol{x}_1) & \phi_{2yy}(\boldsymbol{x}_1) & \ldots & \phi_{nyy}(\boldsymbol{x}_1) \\ \phi_{1yy}(\boldsymbol{x}_2) & \phi_{2yy}(\boldsymbol{x}_2) & \cdots & \phi_{nyy}(\boldsymbol{x}_2) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{1yy}(\boldsymbol{x}_n) & \phi_{2yy}(\boldsymbol{x}_n) & \cdots & \phi_{nyy}(\boldsymbol{x}_n) \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} \tag{23}$$

Considering the Kronecker delta function property: $\phi_i(x_j) = \begin{cases} 1, & i = j\,, \quad i,j = 1,2,3,\ldots,n \\ 0, & i \neq j\,, \quad i,j = 1,2,3,\ldots,n \end{cases}$, while substituting the boundary condition (19) – (20) into equation (23) gives:

$$\frac{du_i}{dt} + v_1 \begin{pmatrix} \phi_{1x}(\boldsymbol{x}_1) & \phi_{2x}(\boldsymbol{x}_1) & \cdots & \phi_{nx}(\boldsymbol{x}_1) \\ \phi_{1x}(\boldsymbol{x}_2) & \phi_{2x}(\boldsymbol{x}_2) & \cdots & \phi_{nx}(\boldsymbol{x}_2) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{1x}(\boldsymbol{x}_n) & \phi_{2x}(\boldsymbol{x}_n) & \cdots & \phi_{nx}(\boldsymbol{x}_n) \end{pmatrix} \begin{pmatrix} \beta_1 \\ u_2 \\ \vdots \\ \gamma_1 \end{pmatrix} + v_1 \begin{pmatrix} \phi_{1y}(\boldsymbol{x}_1) & \phi_{2y}(\boldsymbol{x}_1) & \ldots & \phi_{ny}(\boldsymbol{x}_1) \\ \phi_{1y}(\boldsymbol{x}_2) & \phi_{2y}(\boldsymbol{x}_2) & \cdots & \phi_{ny}(\boldsymbol{x}_2) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{1y}(\boldsymbol{x}_n) & \phi_{2y}(\boldsymbol{x}_n) & \cdots & \phi_{ny}(\boldsymbol{x}_n) \end{pmatrix} \begin{pmatrix} \beta_2 \\ u_2 \\ \vdots \\ \gamma_2 \end{pmatrix}$$

$$= D_1 \begin{pmatrix} \phi_{1xx}(\boldsymbol{x}_1) & \phi_{2xx}(\boldsymbol{x}_1) & \cdots & \phi_{nxx}(\boldsymbol{x}_1) \\ \phi_{1xx}(\boldsymbol{x}_2) & \phi_{2xx}(\boldsymbol{x}_2) & \cdots & \phi_{nxx}(\boldsymbol{x}_2) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{1xx}(\boldsymbol{x}_n) & \phi_{2xx}(\boldsymbol{x}_n) & \cdots & \phi_{nxx}(\boldsymbol{x}_n) \end{pmatrix} \begin{pmatrix} \beta_1 \\ u_2 \\ \vdots \\ \gamma_1 \end{pmatrix} + D_2 \begin{pmatrix} \phi_{1yy}(\boldsymbol{x}_1) & \phi_{2yy}(\boldsymbol{x}_1) & \ldots & \phi_{nyy}(\boldsymbol{x}_1) \\ \phi_{1yy}(\boldsymbol{x}_2) & \phi_{2yy}(\boldsymbol{x}_2) & \cdots & \phi_{nyy}(\boldsymbol{x}_2) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{1yy}(\boldsymbol{x}_n) & \phi_{2yy}(\boldsymbol{x}_n) & \cdots & \phi_{nyy}(\boldsymbol{x}_n) \end{pmatrix} \begin{pmatrix} \beta_2 \\ u_2 \\ \vdots \\ \gamma_2 \end{pmatrix}$$

(24)

where $\beta_1 = \beta_1(y_1, t_1)$, $\beta_2 = \beta_2(x_1, t_1)$, $\gamma_1 = \gamma_1(y_n, T)$, $\gamma_2 = \gamma_1(x_n, T)$ and $i = 1, 2 \ldots n$

Considering the specified flux-averaged concentrations $(u_1 = \beta_1, u_n = \gamma_1)$, $(u_1 = \beta_2, u_n = \gamma_2)$ on the boundary nodes $\boldsymbol{x}_1 = (x_1, y_1, t_1)$ and $\boldsymbol{x}_n = (x_n, y_n, T)$ of the space coordinates x

and y respectively in (24), and partitioning the $(n \times n)$ global matrix equation in (24) leads to $(n-2) \times (n-2)$ submatrix equation for the unknown field nodal variables: $u_2, u_3, \ldots u_{n-1}$ for the flux-averaged concentrations at interior nodes $\boldsymbol{x}_i = (x_i, y_i, z_i, t_i), i = 2, 3, \ldots, n-1$, expressed as:

$$\frac{du_i}{dt} + v_1 \begin{pmatrix} \phi_{2x}(\boldsymbol{x}_2) & \phi_{3x}(\boldsymbol{x}_2) \cdots & \phi_{sx}(\boldsymbol{x}_2) \\ \phi_{2x}(\boldsymbol{x}_3) & \phi_{3x}(\boldsymbol{x}_3) \cdots & \phi_{sx}(\boldsymbol{x}_3) \\ \vdots & \vdots \quad \vdots & \vdots \\ \phi_{2x}(\boldsymbol{x}_s) & \phi_{3x}(\boldsymbol{x}_s) \cdots & \phi_{sx}(\boldsymbol{x}_s) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \\ \vdots \\ u_s \end{pmatrix} + v_1 \begin{pmatrix} \phi_{2y}(\boldsymbol{x}_2) & \phi_{3y}(\boldsymbol{x}_2) \ldots & \phi_{sy}(\boldsymbol{x}_2) \\ \phi_{2y}(\boldsymbol{x}_3) & \phi_{3y}(\boldsymbol{x}_3) \cdots & \phi_{sy}(\boldsymbol{x}_3) \\ \vdots & \vdots \quad \vdots & \vdots \\ \phi_{2y}(\boldsymbol{x}_s) & \phi_{3y}(\boldsymbol{x}_s) \cdots & \phi_{sy}(\boldsymbol{x}_s) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \\ \vdots \\ u_s \end{pmatrix}$$

$$= D_1 \begin{pmatrix} \phi_{2xx}(\boldsymbol{x}_2) & \phi_{3xx}(\boldsymbol{x}_2) \cdots & \phi_{sxx}(\boldsymbol{x}_2) \\ \phi_{2xx}(\boldsymbol{x}_2) & \phi_{3xx}(\boldsymbol{x}_3) \cdots & \phi_{sxx}(\boldsymbol{x}_3) \\ \vdots & \vdots \quad \vdots & \vdots \\ \phi_{2xx}(\boldsymbol{x}_s) & \phi_{3xx}(\boldsymbol{x}_s) \cdots & \phi_{sxx}(\boldsymbol{x}_s) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \\ \vdots \\ u_s \end{pmatrix} + D_2 \begin{pmatrix} \phi_{2yy}(\boldsymbol{x}_2) & \phi_{3yy}(\boldsymbol{x}_2) \ldots & \phi_{syy}(\boldsymbol{x}_2) \\ \phi_{2yy}(\boldsymbol{x}_2) & \phi_{3yy}(\boldsymbol{x}_3) \cdots & \phi_{syy}(\boldsymbol{x}_3) \\ \vdots & \vdots \quad \vdots & \vdots \\ \phi_{2yy}(\boldsymbol{x}_s) & \phi_{3yy}(\boldsymbol{x}_s) \cdots & \phi_{syy}(\boldsymbol{x}_s) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \\ \vdots \\ u_s \end{pmatrix}$$

(25)

where $i = 2, 3, \ldots . s = n-1$.

Using equation (21), the corresponding initial conditions for the unknown field nodal variables: $u_2, u_3, \ldots u_{n-1}$ are expressed as:

$$u(x_\alpha, y_\alpha, 0) = \eta(x_\alpha, y_\alpha) \quad \alpha = 2, 3 \ldots s = n-1 \tag{26}$$

Then, the system of ODEs in equation (25) and the initial conditions in equation (26) are numerically integrated using Matlab ode 45, to obtain the computational values of $u_2, u_3, \ldots u_{n-1}$.

3 Results and Discussions

Here, the new method Meshfree Method of Lines (MFMOL), is applied to solve 2D Advection Diffusion Equations (ADE) for multiphase flows of solute transport in a nonreactive, source-free, homogeneous and isotropic porous media. For ADE used for modelling solute transport, the peclet number, $Pe \gg 1$ corresponds to the advection time scale being greater than the diffusion time scale of the flow and this causes an unstable flow. However, the Peclet number, $Pe \ll 1$ corresponds to the advection time scale being less than

the diffusion time scale of the flow and this gives a stable flow. For all the computations, the space variable discretizations for the problem complex domain $\Omega^2 = [0,1] \times [0,1]$ are done using scattered and unevenly distributed nodal points: $\{(0,0), (0.5, 0.3), (0.7, 0.8), (1,1)\}$.

Example 1: Consider the *diffusion-controlled* solute transport equation (1) with the dispersion coefficients $D_1 = D_2 = 0.5\ m^2 s^{-1}$, the flow average linear velocities $v_1 = v_2 = 0.5\ m$. The initial and boundary conditions are:

$$\eta(x, y, 0) = e^{-x} + e^{-y} \tag{27}$$

$$\beta_1(y,t) = (1 + e^{-y})e^{-4t}, \gamma_1(y,t) = (e^{-1} + e^{-y})e^{-4t} \tag{28}$$

$$\beta_2(x,t) = (e^{-x} + 1)e^{-4t},\ \gamma_2(y,t) = (e^{-x} + e^{-1})e^{-4t} \tag{29}$$

The flow Peclet number: $Pe = \frac{|v|L}{|D|} = 1$, where $L_1 = L_2 = 1$. The existing model's exact solution is given as [13]:

$$u(x,y,t) = (e^{-x} + e^{-y})e^{-4t}, \quad (x,y,t) \in [0,1] \times [0,1] \times [0,0.1] \tag{30}$$

Using the procedures in equations (5) – (26), gives the MFMOL approximation for the system of ODEs:

$$\frac{du_i}{dt} = D_1 \begin{pmatrix} \phi_{2xx}(\boldsymbol{x}_2) & \phi_{3xx}(\boldsymbol{x}_2) \\ \phi_{2xx}(\boldsymbol{x}_3) & \phi_{3xx}(\boldsymbol{x}_3) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix} - v_1 \begin{pmatrix} \phi_{2x}(\boldsymbol{x}_2) & \phi_{3x}(\boldsymbol{x}_2) \\ \phi_{2x}(\boldsymbol{x}_3) & \phi_{3x}(\boldsymbol{x}_3) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix} -$$

$$v_2 \begin{pmatrix} \phi_{2y}(\boldsymbol{x}_2) & \phi_{3y}(\boldsymbol{x}_2) \\ \phi_{2y}(\boldsymbol{x}_3) & \phi_{3y}(\boldsymbol{x}_3) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix}$$

$$+\ D_2 \begin{pmatrix} \phi_{2yy}(\boldsymbol{x}_2) & \phi_{3xx}(\boldsymbol{x}_2) \\ \phi_{2yy}(\boldsymbol{x}_3) & \phi_{3xx}(\boldsymbol{x}_3) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix}, \quad i = 2 \text{ and } 3 \tag{31}$$

subject to the initial conditions:

$u(\boldsymbol{x},0) = [u(\boldsymbol{x}_2,0),\ u(\boldsymbol{x}_3,0)] = [e^{-x_2} + e^{-y_2},\ e^{-x_3} + e^{-y_3}]$, where $\boldsymbol{x} = (x,y)$

(32)

The computational outputs of equations (31) – (32) are given as:

$$\begin{pmatrix} \dot{u_2} \\ \dot{u_3} \end{pmatrix} = \begin{pmatrix} -1.3765 & -2.7527 \\ -0.6783 & -1.3563 \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix}, \quad u(x,0) = \begin{pmatrix} 1.3473 \\ 0.9459 \end{pmatrix} \text{ and } \dot{u_i} = \frac{du_i}{dt}$$

(33)

The second iterate of Variational Iteration Method (VIM) solution is obtained as [32]:

$$u(x,y,t) = (1 + 3t + t^2)(e^{-x} + e^{-y}) \tag{34}$$

Example 2: Consider the *advection-controlled* solute transport equation (1) with the dispersion coefficients $D_1 = D_2 = 0.01\ m^2 s^{-1}$, the flow average linear velocities $v_1 = v_2 = 0.8\ ms^{-1}$, The boundary and initial conditions are:

$$\beta_1(y,t) = \frac{1}{4t+1} e^{\left(\frac{-(0.8t+0.5)^2-(y-0.8t-0.5)^2}{0.01(4t+1)}\right)}, \quad \gamma_1(y,t) = \frac{1}{4t+1} e^{\left(\frac{-(0.5-0.8t)^2-(y-0.8t-0.5)^2}{0.01(4t+1)}\right)}$$

(35)

$$\beta_2(x,t) = \frac{1}{4t+1} e^{\left(\frac{-(x-0.8t-0.5)^2-(0.8t+0.5)^2}{0.01(4t+1)}\right)}, \quad \gamma_2(y,t) = \frac{1}{4t+1} e^{\left(\frac{-(x-0.8t-0.5)^2-(0.5-0.8t)^2}{0.01(4t+1)}\right)}$$

(36)

$$\eta(x,y,0) = e^{-100[(x-0.5)^2+(y-0.5)^2]} \tag{37}$$

The flow Peclet number: $Pe = \frac{|v|L}{|D|} = 80$, where $L_1 = L_2 = 1$. The existing exact solution of the model is given as [33]:

$$u(x,y,t) = \frac{1}{4t+1} e^{\left(\frac{-(x-0.8t+0.5)^2-(y-0.8t-0.5)^2}{0.01(4t+1)}\right)}, \quad (x,y,t) \in [0,1] \times [0,1] \times [0,0.3].$$

(38)

The procedures in equation (5) – (26) gives the MFMOL approximation for the system of ODEs:

$$\frac{du_i}{dt} = D_1 \begin{pmatrix} \phi_{2xx}(\boldsymbol{x}_2) & \phi_{3xx}(\boldsymbol{x}_2) \\ \phi_{2xx}(\boldsymbol{x}_3) & \phi_{3xx}(\boldsymbol{x}_3) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix} - v_1 \begin{pmatrix} \phi_{2x}(\boldsymbol{x}_2) & \phi_{3x}(\boldsymbol{x}_2) \\ \phi_{2x}(\boldsymbol{x}_3) & \phi_{3x}(\boldsymbol{x}_3) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix} -$$

$$v_2 \begin{pmatrix} \phi_{2y}(\boldsymbol{x}_2) & \phi_{3y}(\boldsymbol{x}_2) \\ \phi_{2y}(\boldsymbol{x}_3) & \phi_{3y}(\boldsymbol{x}_3) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix}$$

$$+ \; D_2 \begin{pmatrix} \phi_{2yy}(\boldsymbol{x}_2) & \phi_{3xx}(\boldsymbol{x}_2) \\ \phi_{2yy}(\boldsymbol{x}_3) & \phi_{3xx}(\boldsymbol{x}_3) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix} \qquad i = 2 \quad \text{and} \quad 3$$

(39)

subject to the initial conditions:

$$u(\boldsymbol{x}, 0) = [u(\boldsymbol{x}_2, 0),\ u(\boldsymbol{x}_3, 0)] = \left[e^{-100\left[(x_2-0.5)^2+(y_2-0.5)^2\right]},\ e^{-100\left[(x_3-0.5)^2+(y_3-0.5)^2\right]}\right]$$

(40)

The computational values of equations (39) – (40) are given as:

$$\begin{pmatrix} \dot{u_2} \\ \dot{u_3} \end{pmatrix} = \begin{pmatrix} -0.3301 & -0.6597 \\ -0.6758 & -1.3520 \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix}, \quad u(\boldsymbol{x}, 0) = \begin{pmatrix} 0.018316 \\ 2.26E-6 \end{pmatrix} \quad \text{and} \quad \dot{u_i} = \frac{du_i}{dt}$$

(41)

The first iterate of VIM solution is obtained as [32]

$$u(x, y, t) = [1 + \{D(A^2 + B^2 + 2q) - v(A + B)\}t]e^{-100((x-0.5)^2+(y-0.5)^2)}$$

(42)

where: $v = v_1 = v_2 = 0.8$, and $D = D_1 = D_2 = 0.01$

Example 3: Consider the *diffusion-controlled* solute transport equation (1) with the dispersion coefficients $D_1 = 1.4m^2s^{-1}$, $D_2 = 1.7\ m^2s^{-1}$, the flow average linear velocities $v_1 = v_2 = 1.0\ ms^{-1}$, The initial and boundary conditions are:

$$\eta(x, y, 0) = e^{-c_1 x} + e^{-c_2 y} \tag{43}$$

$$\beta_1(y, t) = (1 + e^{-c_2 y})e^{-10t}, \quad \gamma_1(y, t) = (e^{-c_1} + e^{-c_2 y})e^{-10t} \tag{44}$$

$$\beta_2(x,t) = (e^{-c_1 x} + 1)e^{-10t},\ \ \gamma_2(y,t) = (e^{-c_1 x} + e^{-c_2})e^{-10t} \qquad (45)$$

$$c_1 = \frac{-v_1 - \sqrt{{v_1}^2 + 4bD_1}}{2D_1},\ \ c_2 = \frac{-v_2 - \sqrt{{v_2}^2 + 4bD_2}}{2D_2}, b = -10,\ Pe = \frac{|v|L}{|D|} < 1, L_1 = L_2 = 1$$

(46)

The existing exact solution of the model is given as [13]:

$$u(x,y,t) = (e^{-c_1 x} + e^{-c_2 y})e^{-10t}\ ,\quad (x,y,t) \in [0,1] \times [0,1] \times [0,0.1] \qquad (47)$$

Following the procedures in equations (5) – (26) gives the MFMOL approximation for the system of ODEs:

$$\frac{du_i}{dt} = D_1 \begin{pmatrix} \phi_{2xx}(\boldsymbol{x}_2) & \phi_{3xx}(\boldsymbol{x}_2) \\ \phi_{2xx}(\boldsymbol{x}_3) & \phi_{3xx}(\boldsymbol{x}_3) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix} - v_1 \begin{pmatrix} \phi_{2x}(\boldsymbol{x}_2) & \phi_{3x}(\boldsymbol{x}_2) \\ \phi_{2x}(\boldsymbol{x}_3) & \phi_{3x}(\boldsymbol{x}_3) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix} -$$

$$v_2 \begin{pmatrix} \phi_{2y}(\boldsymbol{x}_2) & \phi_{3y}(\boldsymbol{x}_2) \\ \phi_{2y}(\boldsymbol{x}_3) & \phi_{3y}(\boldsymbol{x}_3) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix}$$

$$+\ D_2 \begin{pmatrix} \phi_{2yy}(\boldsymbol{x}_2) & \phi_{3xx}(\boldsymbol{x}_2) \\ \phi_{2yy}(\boldsymbol{x}_3) & \phi_{3xx}(\boldsymbol{x}_3) \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix} \quad i = 2 \text{ and } 3 \qquad (48)$$

subject to the initial conditions:

$$u(\boldsymbol{x},0) = [u(\boldsymbol{x}_2,0),\ u(\boldsymbol{x}_3,0)] = \ [e^{-c_1 x_2} + e^{-c_2 y_2},\ e^{-c_1 x_3} + e^{-c_2 y_3}] \qquad (49)$$

The computational outputs of equations (48) – (49) give:

$$\begin{pmatrix} \dot{u_2} \\ \dot{u_3} \end{pmatrix} = \begin{pmatrix} -4.02179 & -8.04295 \\ -2.55048 & -5.1006 \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \end{pmatrix},\ \ u(\boldsymbol{x},0) = \begin{pmatrix} 2.4939 \\ 2.9610 \end{pmatrix} \text{ and } \dot{u_i} = \frac{du_i}{dt} \quad 50)$$

The first iterate of the VIM solution is given as [32]:

$$u(x,y,t) = (1 + \{D_1 {c_1}^2 + v_1 c_1\}t)e^{-c_1 x,} + (1 + \{D_2 {c_2}^2 + v_2 c_2\}t)e^{-c_2 y} \qquad (51)$$

Then equations (33), (41), and (50) are numerically integrated via the MATLAB ode45 solver, to obtain the computational values of the unknown nodal values u_2 and u_3.

Tables 1-3 show the computational results of the new method MFMOL, the exact solutions, the Variational Iteration Method (VIM) solution, and the absolute errors for examples 1-3, respectively. Figs 1-20 show graphical displays comparing the exact solutions, MFMOL solutions, and the absolute errors for examples 1-3.

Table 1. Example 1's table for the exact solutions, the MFMOL solutions, and the absolute error computed at $(0.5, 0.3)$ and $(0.7,0.8)$ over $0 \leq t \leq 0.001$ with $\Delta t = 0.0002$

(x, y)	t	$u_{ex}(x, y, t)$	MFMOL: $u_{num}(x, y, t)$	VIM [32]	$e_1: \lvert u_{ex} - u_{num} \rvert$	$e_2: \lvert u_{ex} - VIM \rvert$
(0.5,0.3)	0.0000	1.3473	1.3473	1.3473	0.0E-4	0.0E-4
	0.0002	1.3463	1.3464	1.3482	1.0E-4	1.9E-3
	0.0004	1.3452	1.3455	1.3490	3.0E-4	3.8E-3
	0.0006	1.3441	1.3446	1.3498	5.0E-4	5.7E-3
	0.0008	1.3430	1.3437	1.3506	7.0E-4	7.6E-3
	0.001	1.3420	1.3428	1.3514	8.0E-4	9.4E-3
(0.7,0.8)	0.0000	0.9459	0.9459	0.9459	0.0E-4	0.0E-4
	0.0002	0.9452	0.9455	0.9465	3.0E-4	7.0E-4
	0.0004	0.9444	0.9450	0.9470	6.0E-4	2.6E-3
	0.0006	0.9436	0.9446	0.9476	1.0E-3	4.0E-3
	0.0008	0.9429	0.9441	0.9482	1.2E-3	5.3E-3
	0.001	0.9421	0.9437	0.9488	1.6E-3	6.7E-3

Table 2. Example 2's table for the exact solutions, the MFMOL numerical solutions, and the absolute errors computed at points $(0.5, 0.3)$ and $(0.7,0.8)$ over $0 \leq t \leq 0.1$ with $\Delta t = 0.02$

(x, y)	t	$u_{ex}(x, y, t)$	MFMOL:$u_{num}(x, y, t)$	VIM [32]	$e_1: \lvert u_{ex} - u_{num} \rvert$	$e_2: \lvert u_{ex} - u_{num} \rvert$
(0.5,0.3)	0.00	0.01830	0.01830	0.01830	0.0E-4	0.0E-4
	0.02	0.01200	0.01820	0.01820	6.2E-3	6.2E-3
	0.04	0.00760	0.01810	0.01820	1.1E-2	1.1E-2
	0.06	0.00470	0.01790	0.01810	1.3E-2	1.3E-2

	0.08	0.00280	0.01780	0.01800	1.5E-2	1.5E-2
	0.1	0.00170	0.01770	0.01790	1.6E-2	1.6E-2
(0.7,0.8)	0.00	0.00000	0.00000	0.00000	0.0E-5	0.0E-5
	0.02	0.00002	0.00025	0.00001	2.3E-4	1.0E-5
	0.04	0.00015	0.00051	0.00001	3.6E-4	1.4E-4
	0.06	0.00074	0.00077	0.00002	3.0E-5	7.2E-4
	0.08	0.00270	0.00100	0.00003	1.7E-3	2.7E-3
	0.1	0.00800	0.00130	0.00003	6.7E-3	8.0E-3

Table 3. Example 3's table for comparison of the exact solution and the MFMOL numerical solution, with the absolute error computed at $(0.5, 0.3)$ and $(0.7,0.8)$ over $0 \leq t \leq 0.001$ with $\Delta t = 0.0002$

(x, y)	t	$u_{ex}(x, y, t)$	MFMOL:$u_{num}(x, y, t)$	VIM[32]	e_1: $\lvert u_{ex} - u_{num} \rvert$	e_1: $\lvert u_{ex} - u_{num} \rvert$
(0.5,0.3)	0.0000	2.4940	2.4939	2.4940	1.0E-4	0.0E-4
	0.0002	2.4939	2.4871	2.4939	6.8E-3	0.0E-4
	0.0004	2.4939	2.4804	2.4939	1.4E-2	0.0E-4
	0.0006	2.4938	2.4736	2.4938	2.0E-2	0.0E-4
	0.0008	2.4938	2.4669	2.4938	2.7E-2	0.0E-4
	0.001	2.4937	2.4602	2.4937	3.4E-2	0.0E-4
(0.7,0.8)	0.0000	2.9610	2.9610	2.9610	0.0E-4	0.0E-4
	0.0002	2.9609	2.9567	2.9609	4.2E-3	0.0E-4
	0.0004	2.9609	2.9524	2.9609	8.5E-3	0.0E-4
	0.0006	2.9608	2.9482	2.9608	1.3E-2	0.0E-4
	0.0008	2.9610	2.9439	2.9608	1.7E-2	2.0E-4
	0.001	2.9607	2.9396	2.9607	2.1E-2	0.0E-4

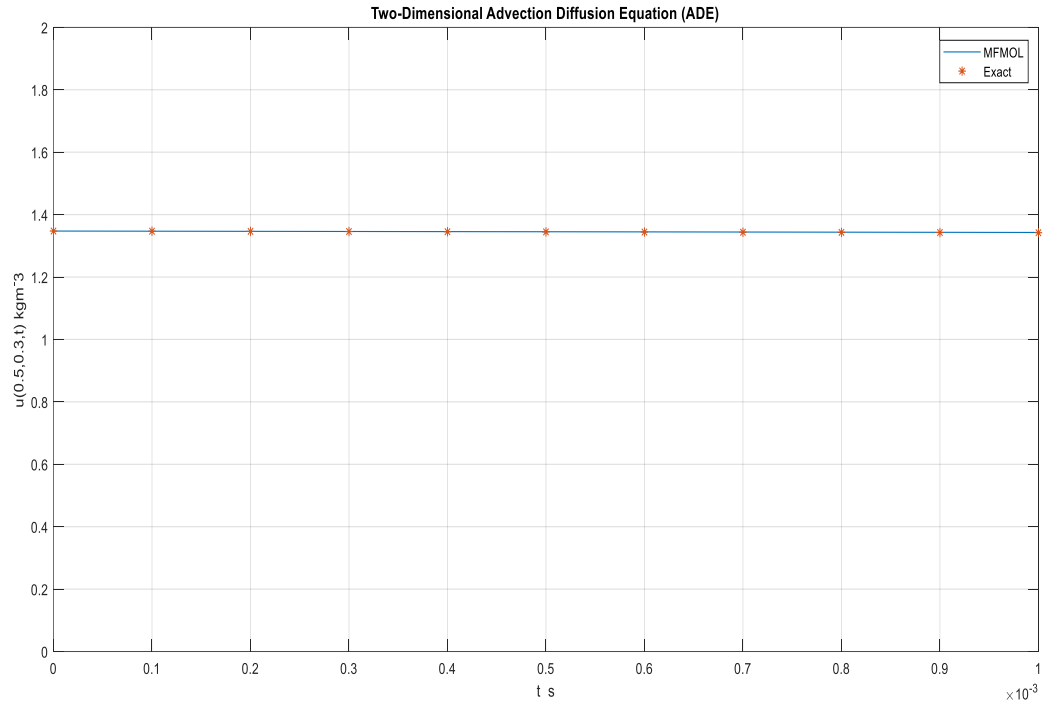


Figure 1. Plot of Example 1, for the flux averaged solute concentration corresponding to fixed point $(x, y) = (0.5, 0.3)$ over $0 \leq t \leq 0.001$ with time steps $\Delta t = 0.0001$

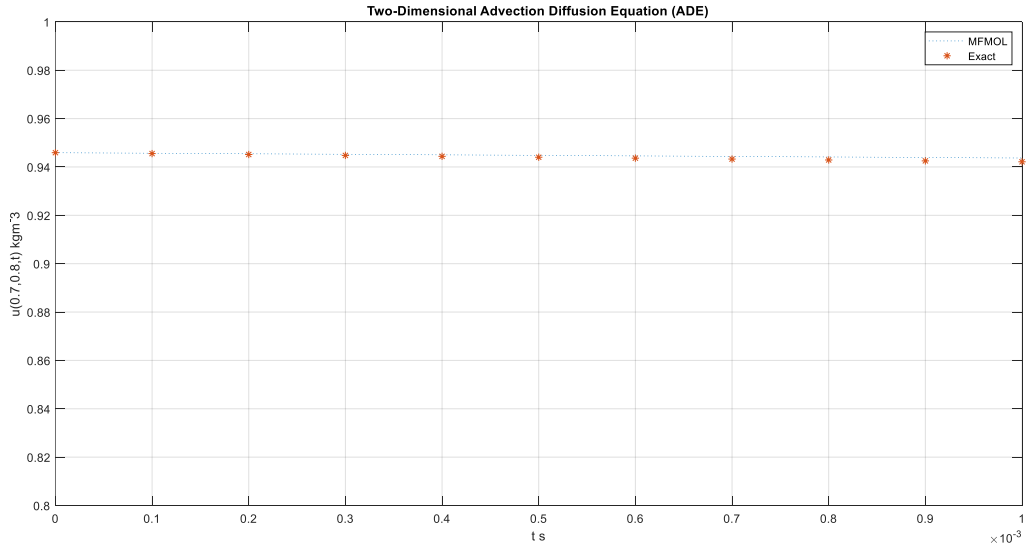


Figure 2. Plot of Example 1, for the flux averaged solute concentration corresponding to fixed point $(x, y) = (0.7, 0.8)$ over $0 \leq t \leq 0.001$ with time steps $\Delta t = 0.0001$

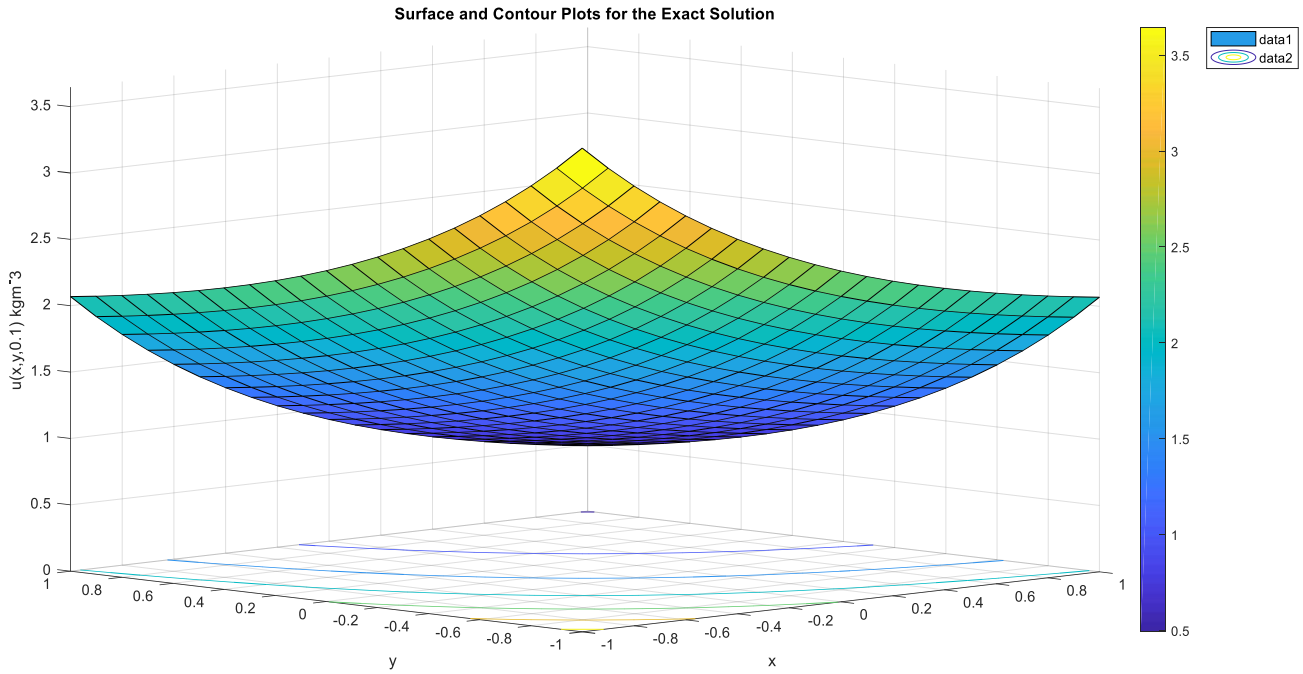


Figure 3. Surface and Contour plots of Example 1 for the exact solutions of the spatial distributions of the flux-averaged concentration $u(x, y, t)$ at $t = 0.1$

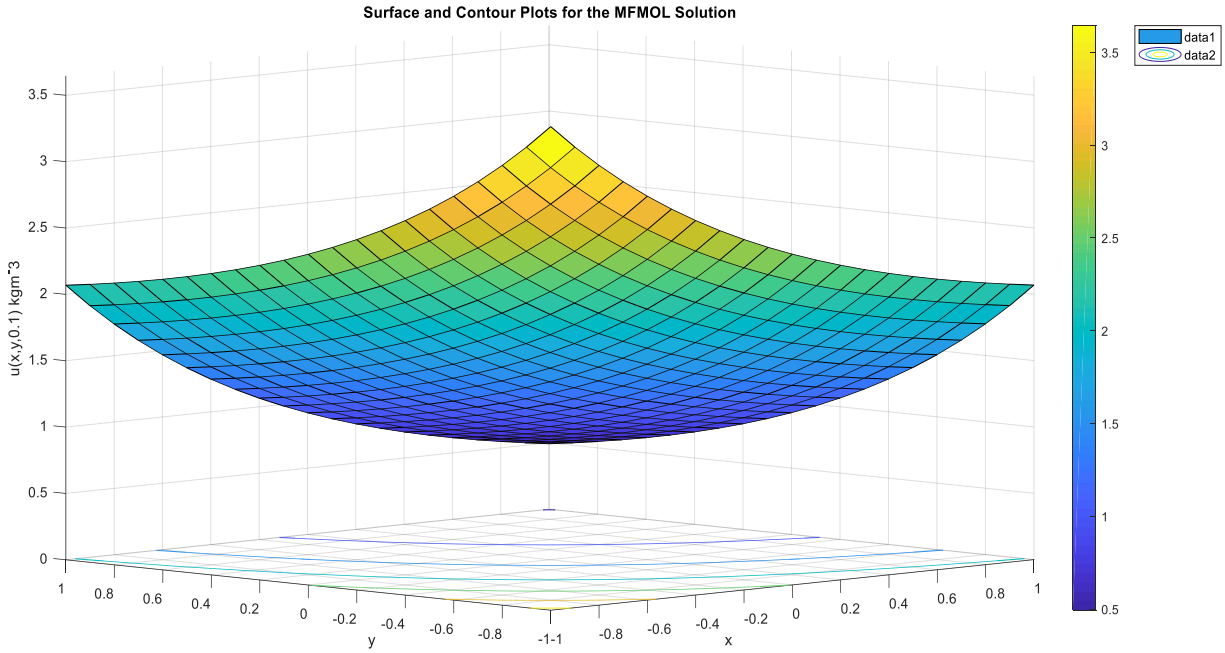


Figure 4. Surface and Contour plots of Example 1 for the MFMOL solutions of the spatial distributions of the flux-averaged concentration $u(x, y, t)$ at $t = 0.1$

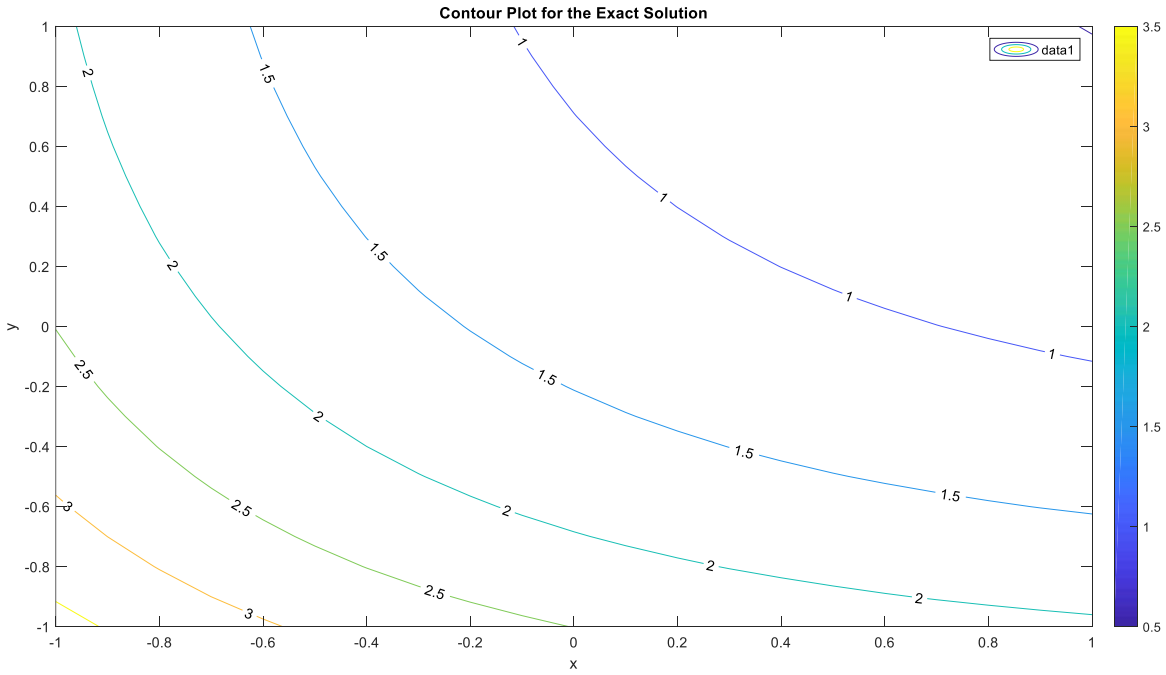


Figure 5. Contour plots of Example 1, for the exact solutions of the spatial distributions of the flux-averaged concentration $u(x, y, t)$ at $t = 0.1$

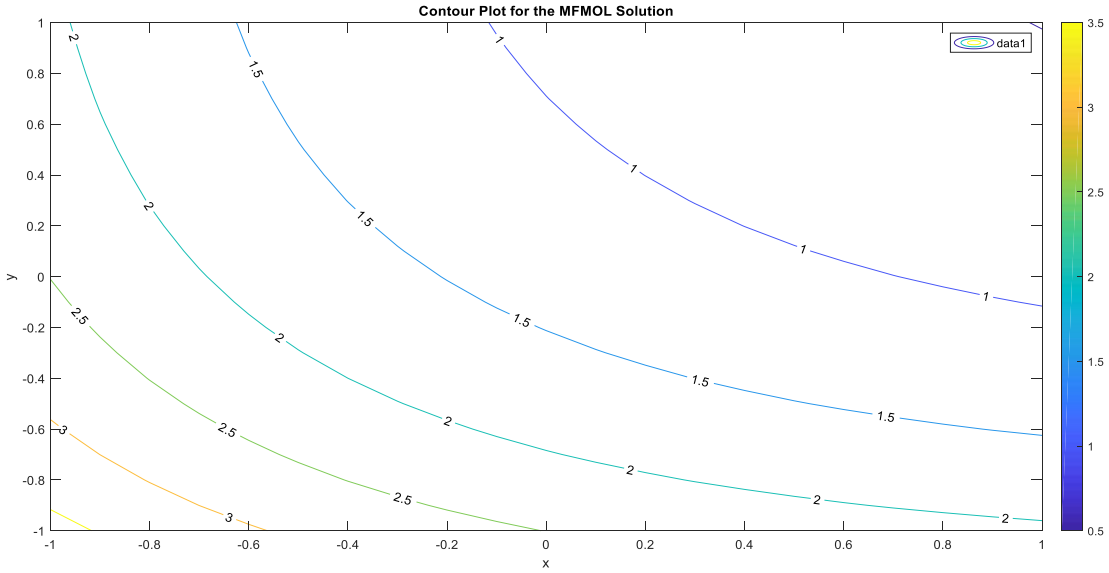


Figure 6. Contour plots of Example 1, for the MFMOL solutions of the spatial distributions of the flux-averaged concentration $u(x, y, t)$ at $t = 0.1$

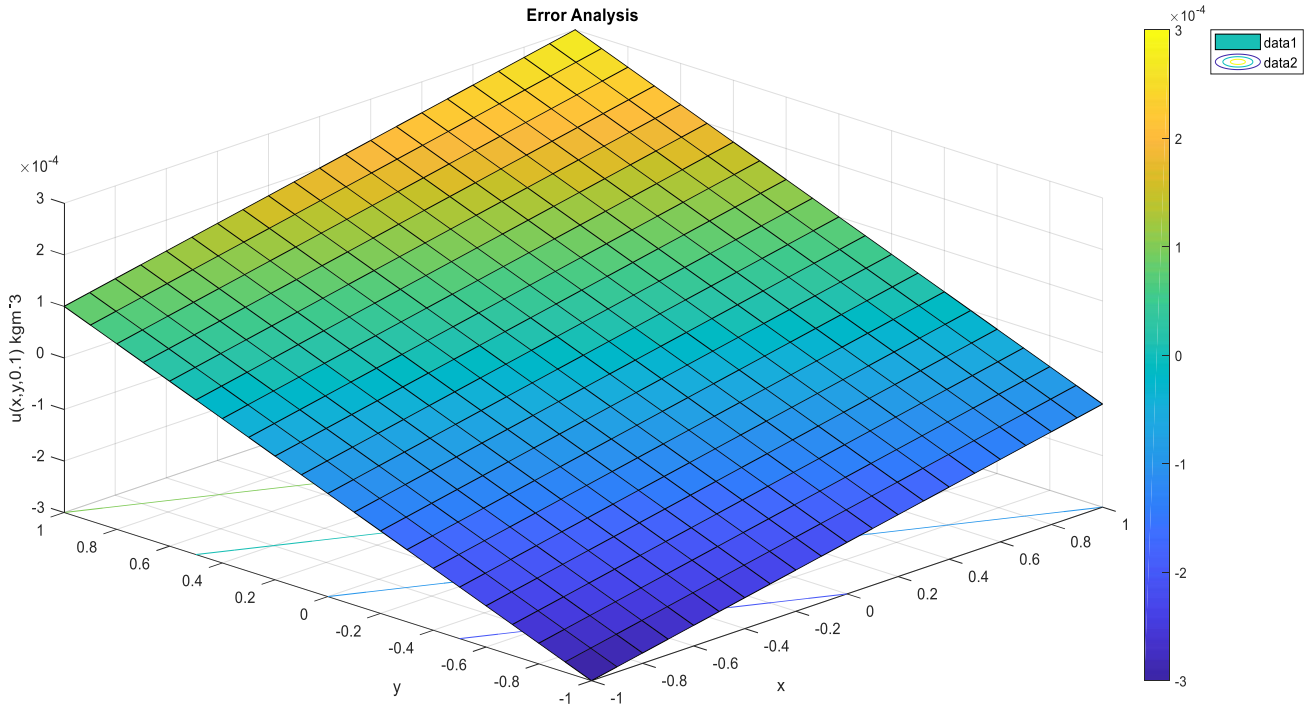


Figure 7. Surface and Contour plots of the Error between the exact and MFMOL solutions of Example 1 for the spatial distributions of the flux-averaged concentration $u(x, y, t)$ at $t = 0.1$

In Table 1, the numerical results of flux-averaged (flowing) concentrations of example 1 using the proposed method MFMOL were compared with the exact solutions and the Variational Iteration Method (VIM) solution for effluent concentrations at varied temporal step size Δt and fixed position (x, y) for diffusion-dominated flow ($Pe = 1$), with initial and boundary conditions, in homogeneous porous medium. The proposed method MFMOL gave good approximations and stability than VIM solutions as shown with the computational values and the absolute errors e_1 and e_2 respectively. This established convergence of the new method MFMOL in simulating flowing concentration for 2D solute transport models in homogeneous and isotropic porous media. The sensitivity of the MFMOL values to spatial nodes (x, y) is observed in Table 1, although negligible, where the point $(0.5, 0.3)$ gave more

accuracy than the point $(0.7, 0.8)$ as shown by absolute errors e_1. Figure 1 and Figure 2 showed the graphical displays of the flux-averaged concentrations of the MFMOL solutions and the exact solutions at fixed spatial points $(0.5, 0.3)$ and $(0.7, 0.8)$ respectively over various temporal step size Δt. Figure 3 and Figure 4 showed the combined surface and contour plots for the exact solution and the MFMOL solution respectively, for spatial distributions $u(x, y, t)$ of flux-averaged solute concentration at fixed $t = 0.1$. Figure 5 and Figure 6 showed the explicit contour plots for the exact solutions and the MFMOL solutions respectively for spatial distributions $u(x, y, t)$ of flux-averaged concentrations at fixed time $t = 0.1$. Figure 7 showed the combined surface and contour plots of the Error between the MFMOL solutions and the exact solutions of the flux-averaged concentration $u(x, y, t)$ at $t = 0.1$.

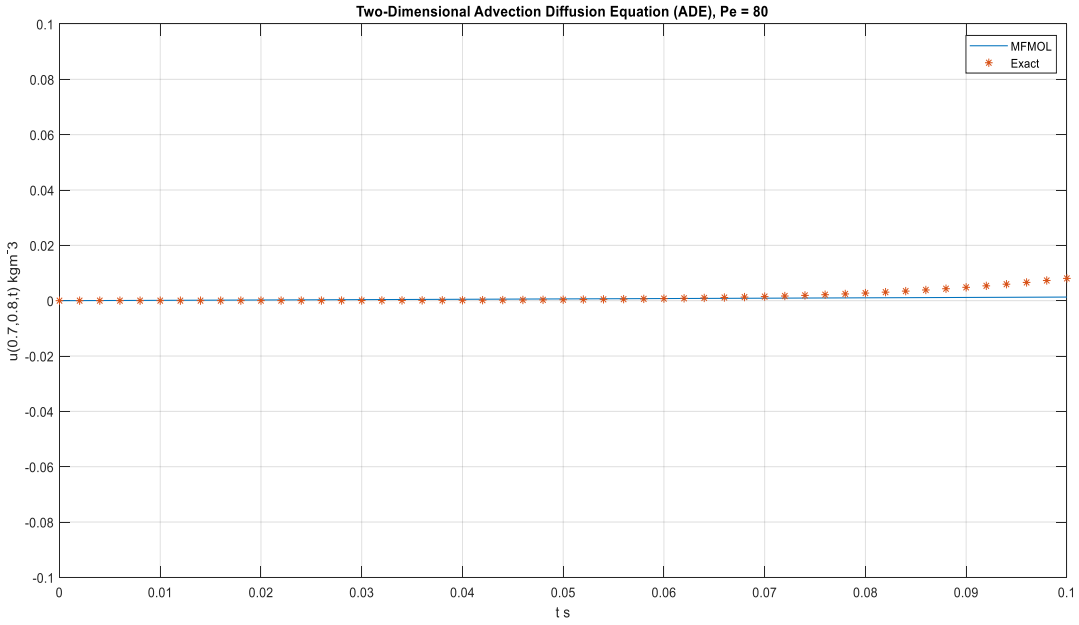


Figure 8. Plot of Example 2 corresponding to $(0.7, 0.8)$ over $0 \leq t \leq 0.1$ with $\Delta t = 0.002$

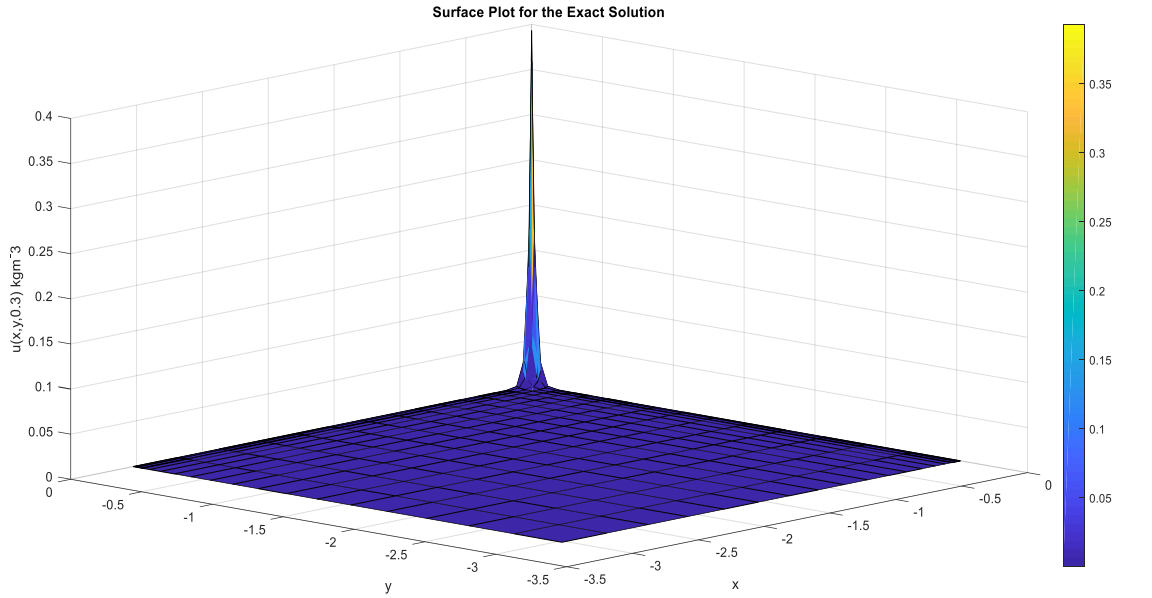


Figure 9. Surface Plots of Example 2 for the exact solutions of the spatial distributions of the flux-averaged solute concentration $u(x, y, t)$ at $t = 0.3$

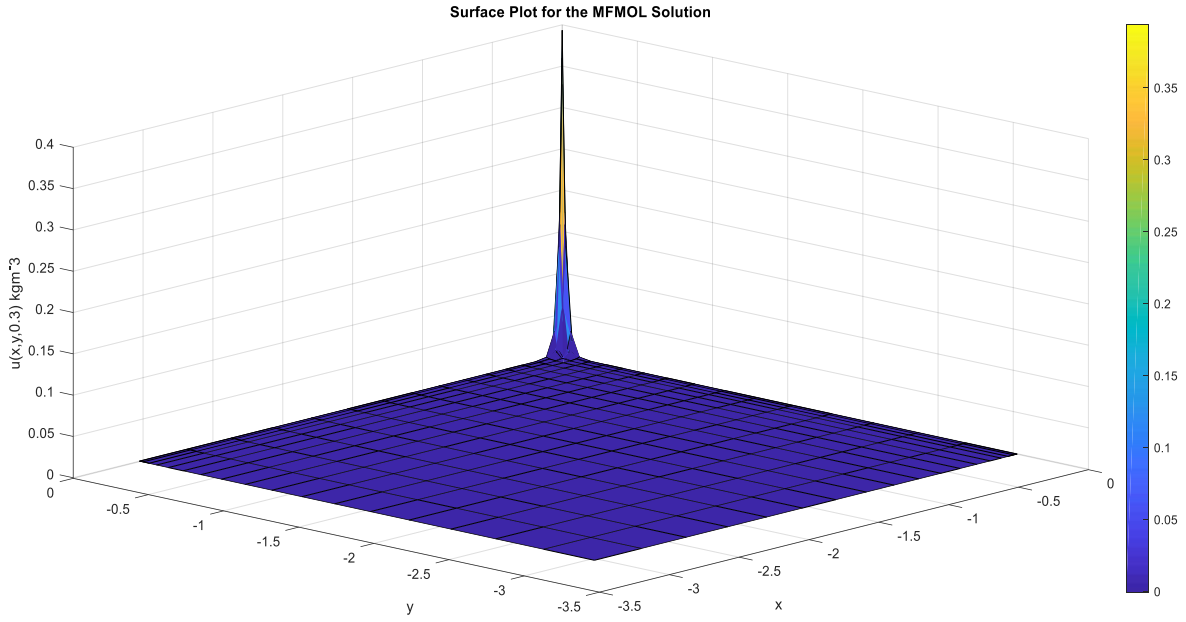


Figure 10. Surface Plots of Example 2 for the MFMOL solutions of the spatial distributions of the flux-averaged solute concentration $u(x, y, t)$ at $t = 0.3$

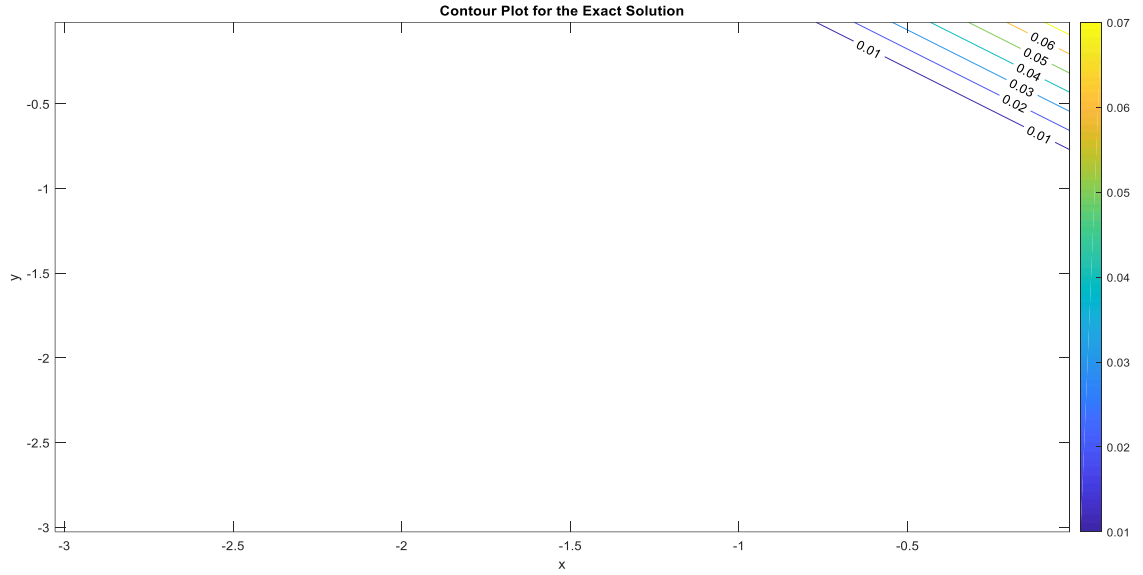


Figure 11. Contour plots of Example 2, for the exact solutions of spatial distributions of the flux-averaged concentration $u(x, y, t)$ at $t = 0.3$

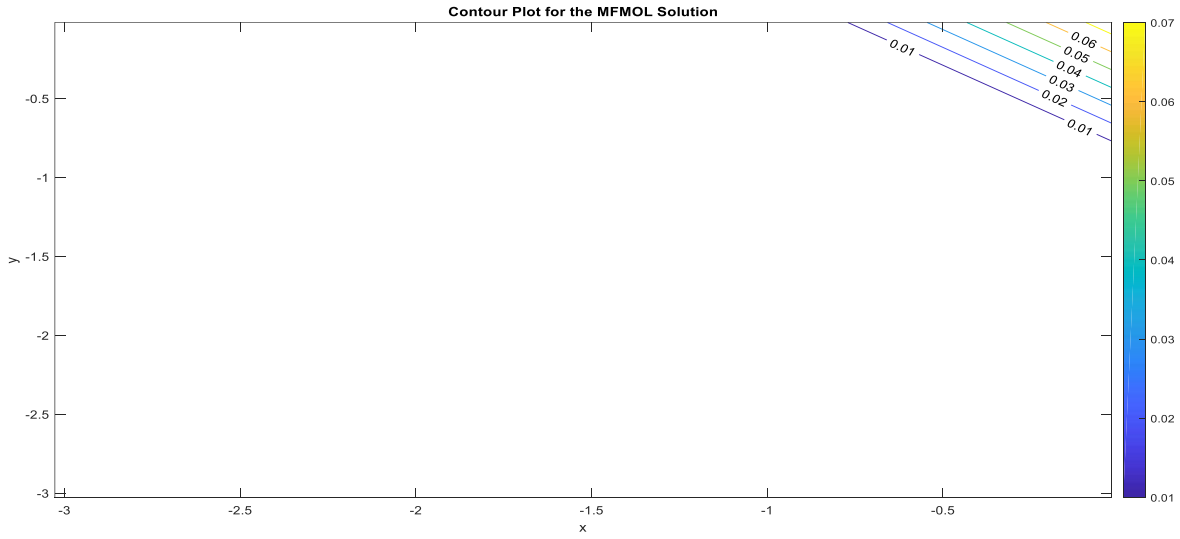


Figure 12. Contour plots of Example 2, for the MFMOL solutions of spatial distributions of the flux-averaged concentration $u(x, y, t)$ at $t = 0.3$

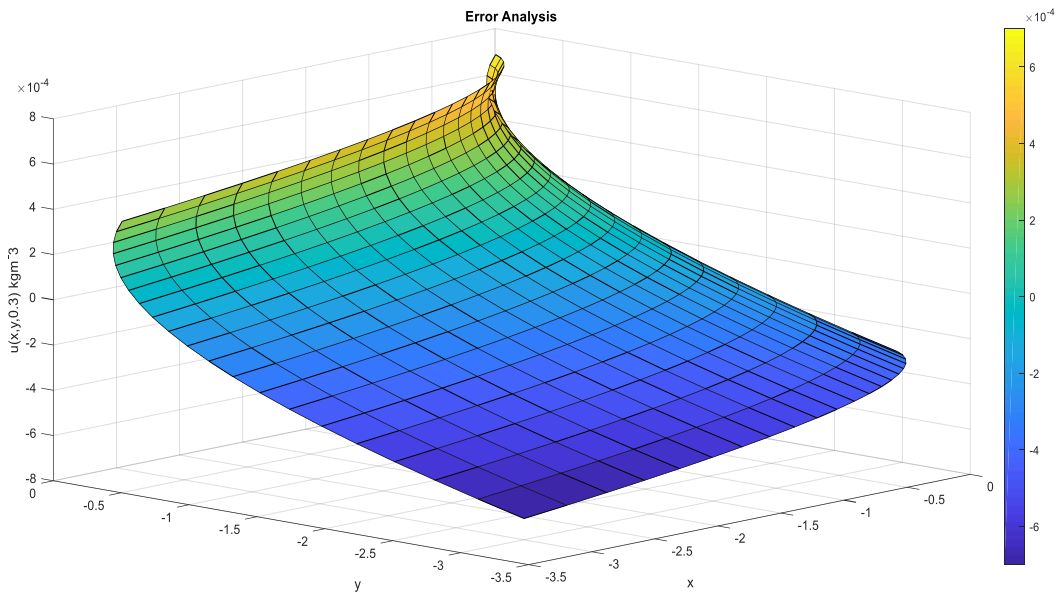


Figure 13. Surface plots of the errors between the exact and MFMOL solutions of Example 2 for the spatial distributions of the flux-averaged concentrations $u(x, y, t)$ at $t = 0.3$

In Table 2, the computational values of the proposed method MFMOL were compared with known exact solutions and the VIM solutions for the flowing concentrations at varied temporal step size Δt and fixed position (x, y) for advection-dominated flow ($Pe = 80$) subject to complex initial and boundary conditions in the complex domain. The proposed method resulted in good approximations and stability with minimal numerical oscillations without the use of a stabilization technique. Its accuracy here is almost the same or slightly higher than the VIM solutions as shown via the computational values and the absolute errors e_1 and e_2 respectively. The absolute errors computed also showed that the MFMOL is convergent for advection-controlled 2D solute transport models where the numerical oscillations or instabilities used to be significant. The sensitivity of the MFMOL values to spatial nodes (x, y) is observed in Table 2, although negligible, where the point $(0.7, 0.8)$ gave more accuracy than the point $(0.5, 0.3)$ as shown by the absolute errors e_1. Also, higher time steps $\Delta t = 0.02$ were used for computational optimal accuracy instead of lower time steps often

used for better accuracy in transient flow models. The use of lower time steps is analogous to mesh refinement for optimal accuracy in mesh-based methods. Fig. 8 showed the graphical display of the flux-averaged concentrations of the MFMOL solutions and the exact solutions at fixed spatial points $(0.5, 0.3)$ over various temporal step size Δt. Figs 9 and 10 showed the surface plots of the exact and the MFMOL solutions respectively for spatial distribution $u(x, y, t)$ of the flux-averaged concentration fixed at $t = 0.3$. Figs. 11 and 12 showed the contour plots for the exact solutions and the MFMOL solutions respectively, for spatial distribution $u(x, y, t)$ of flux-averaged concentration fixed at $t = 0.3$. Fig 13 showed the surface plots of the error between the exact and the MFMOL solutions of the spatial distribution $u(x, y, t)$ for the flux-averaged concentration fixed at $t = 0.1$.

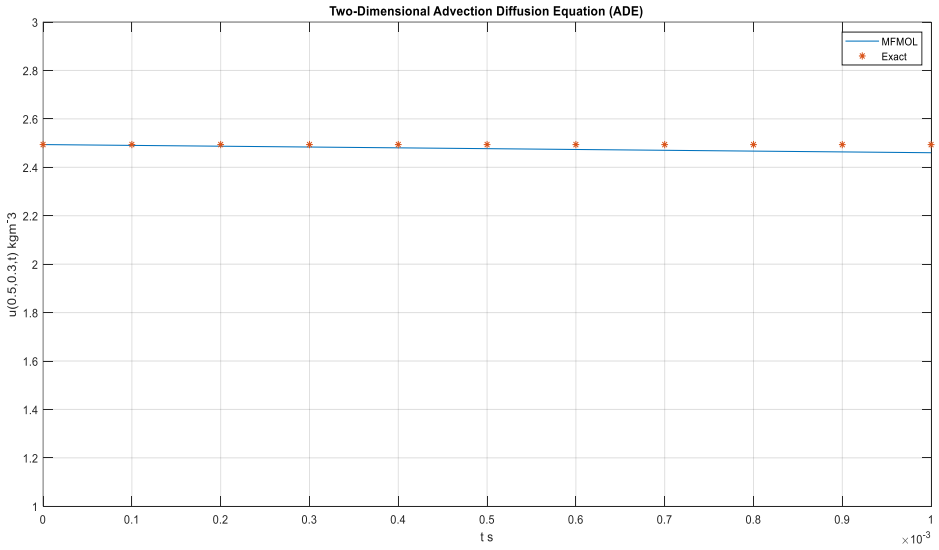


Figure 14. Plot of Example 3 corresponding to $(0.5, 0.3)$ over $0 \leq t \leq 0.001$ with $\Delta t = 0.0001$

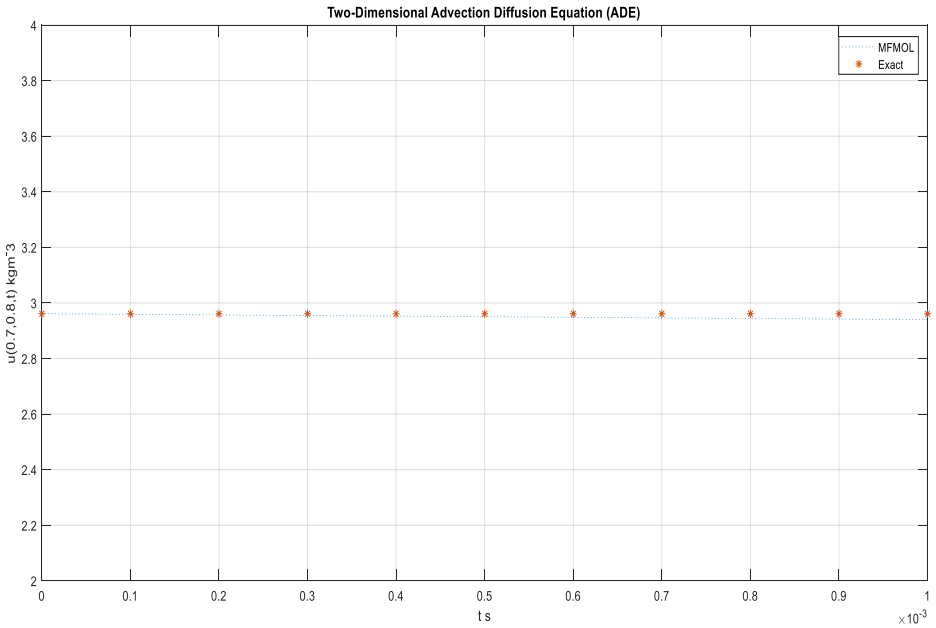


Figure 15. Plot of Example 3 corresponding to $(0.7, 0.8)$ over $0 \leq t \leq 0.001$ wi th$\Delta t = 0.0001$

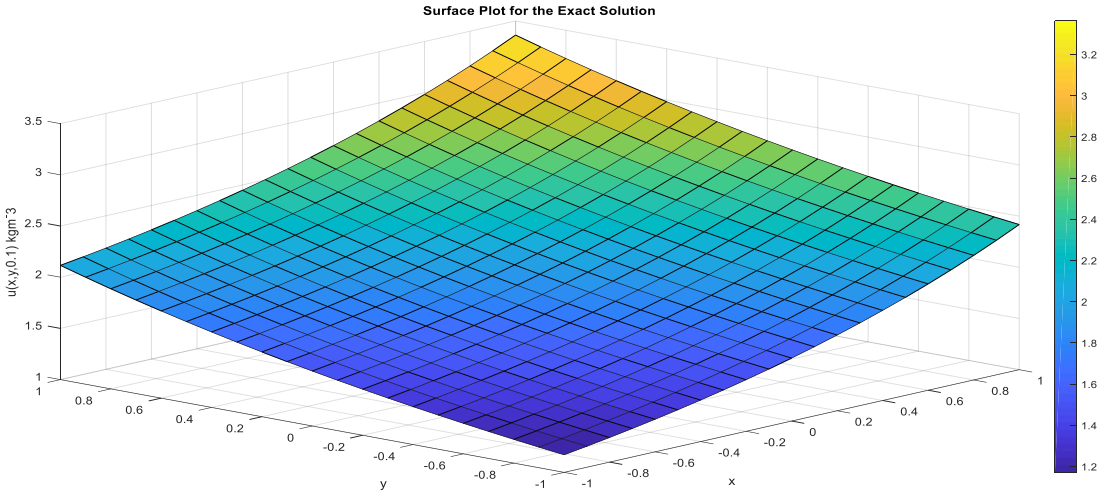


Figure 16. Surface plot of example 3 for the exact solution of the spatial distributions of the flux-averaged solute concentration $u(x, y, t)$ at $t = 0.1$

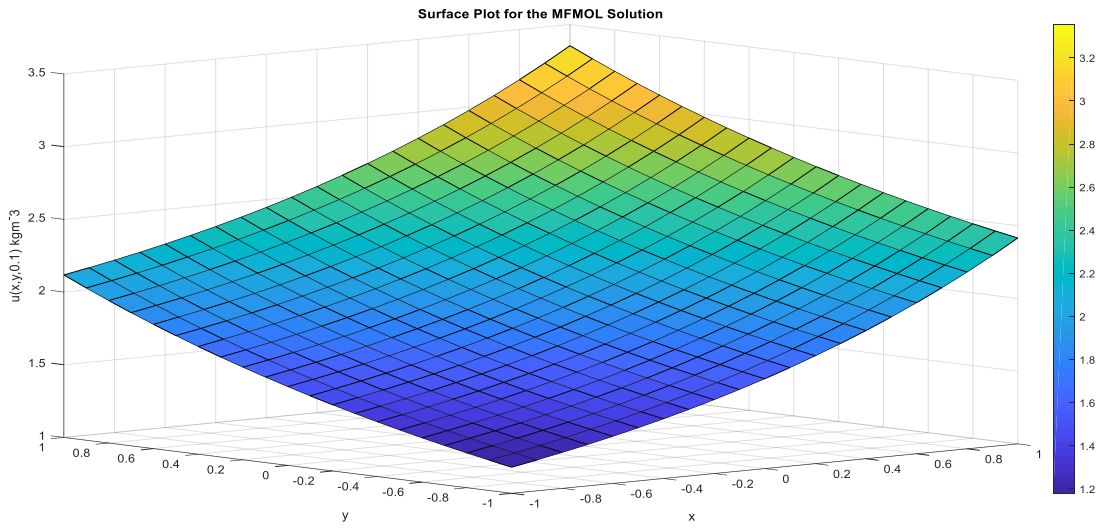


Figure 17. Surface plot of example 3 of the MFMOL solution of the spatial distributions of the flux-averaged solute concentration $u(x, y, t)$ at $t = 0.1$

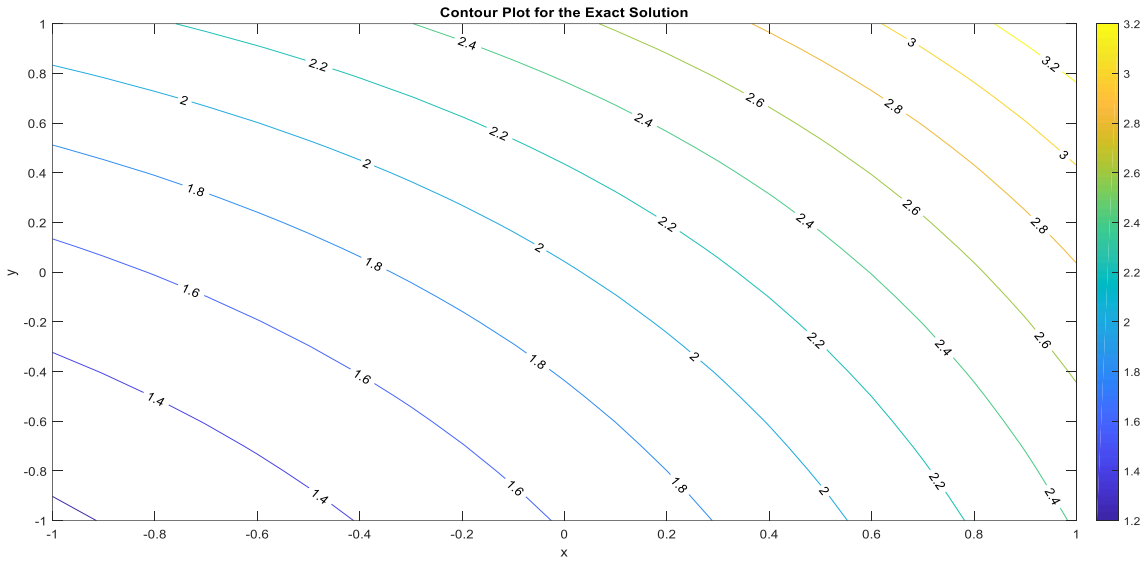


Figure 18. Contour plot of example 3, for the exact solutions of the spatial distributions of the flux-averaged concentration $u(x, y, t)$ at $t = 0.1$

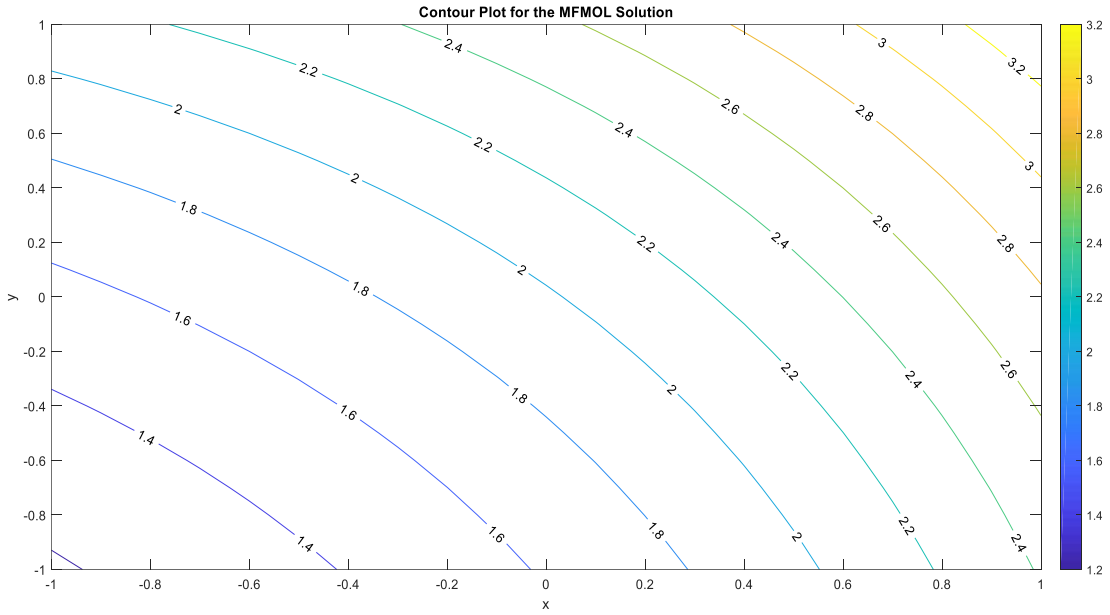


Figure 19. Contour plot of example 3, for the MFMOL solutions of the spatial distributions of the flux-averaged concentration $u(x, y, t)$ at $t = 0.1$

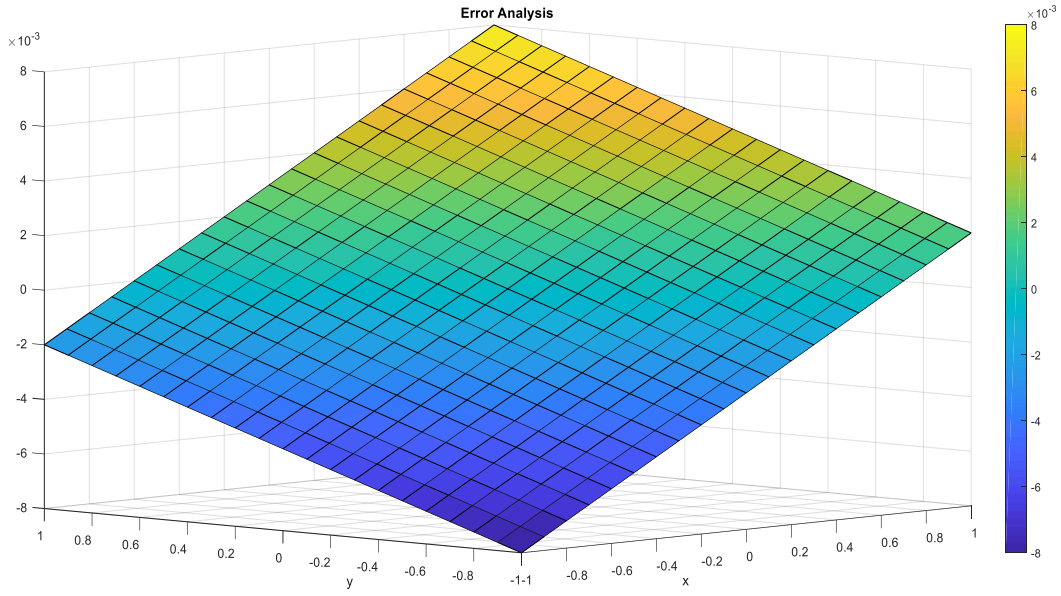


Figure 20. Surface plots of the error between the exact and MFMOL solutions of Example 3 for the spatial distributions of the flux-averaged concentration $u(x, y, t)$ at $t = 0.1$

In Table 3, the approximate values of the proposed MFMOL were compared with existing exact solutions and the VIM solutions for the effluent concentration at varied temporal step size Δt and constant position (x, y) for the case of diffusion-dominated flow $(Pe < 1)$ with the complex initial and boundary conditions in the complex domain of porous structure. It is observed that the proposed method gave a good approximation, as validated by the exact solutions and VIM solutions. Here, the VIM solutions are similar to the exact solutions. Also, the sensitivity of the solutions is independent of both spatial positions (x, y) and the lower time steps Δt; rather, it depends on the nature of the subsidiary conditions. The absolute errors computed also showed that the MFMOL is convergent for diffusion-controlled 2D solute transport models. Figs. 14 and 15 showed the graphical displays of the flux-averaged concentrations of the MFMOL solutions and the exact solutions at fixed spatial points $(0.5, 0.3)$ and $(0.7, 0.8)$, respectively, over various temporal step sizes Δt. Figs. 16 and 17 showed the surface plots of the exact and the MFMOL solutions for spatial distribution $u(x, y, t)$ of the flux-averaged concentrations respectively at $t = 0.1$. Figs. 18 and 19 showed the contour plots for the exact solutions and the MFMOL solutions for spatial distribution $u(x, y, t)$ of flux-averaged concentration respectively at $t = 0.1$. Fig. 20 showed the surface plots of the error between the exact and the MFMOL solutions of the spatial distribution $u(x, y, t)$ for the flux-averaged concentration fixed at $t = 0.1$.

The Variational Iteration Method (VIM) is a powerful computational method for various problems in the applied sciences and engineering, due to its stability and fast convergence to closed-form solutions or approximate solutions in series form [34], [35]. However, it has limited capacity in solving boundary value problems as its zeroth approximation depends only on the initial conditions, and its solution may not satisfy the boundary conditions. This setback is overcome by the proposed method MFMOL, which utilises the Kronecker delta function property to easily impose boundary conditions, making the present method superior to VIM in solving a wide range of boundary value problems.

4 Conclusion

In this paper, a novel Meshfree Method of Lines (MFMOL) was proposed in strong form formulations for solving 2D Advection Diffusion Equations (ADEs) modelling flux-averaged (flowing) concentrations of solute transport in nonreactive, source-free, homogeneous and isotropic porous media. The method was applied to both advective ($Pe > 1$) and diffusive ($Pe < 1$) problems without any stabilization technique for the convective terms of the models, and the results agreed with both existing exact solutions and the Variational Iteration Method (VIM) solutions, with negligible numerical instabilities as shown in Tables 1-3 and Figs. 1–20. This shows the accuracy and efficiency of the new method, not only in simulating 2D solute transport in homogeneous complex domains without a stabilization technique, but also in overcoming the various challenges ranging from low-quality meshes, numerical instabilities, discontinuities, massive computational efforts, high costs, to substantial setup time in generating quality meshes, which are encountered by mesh-based methods due to mesh interpolation for the complex domains. Regarding the easy imposition of boundary conditions, which VIM may not always satisfy in solving solute transport models, the present method shows superior performance over VIM by utilizing the Kronecker delta function property to impose boundary conditions for numerous models easily

This research acknowledges some limitations for optimal computational accuracy of the proposed method. These include the geometrical location of star nodal points in the domain of the problem and its boundary, the time steps and the complexities of the initial and boundary conditions. The sensitivity of the limit factors to the accuracy of the solution is trivial with negligible effects on computational errors.

Despite these limitations, various features of the proposed method involved in generating acceptable computational results have established its solid foundation for handling 2D models with constant parameters in homogeneous and isotropic porous domains. Considering the effectiveness of the proposed method for 2D models, future research is to

explore its capacity for solving 3D models, coupled transport-reaction systems and various time-dependent models such as series RC and LC circuits in electricity and magnetism [36], Magnetohydrodynamic models (MHD) [37], and others. Additionally, with the series of transformation [38], [39], appropriate discretisation of variable-dependent parameters in the models [40] or other approaches, future work will examine the significant potential of the proposed method in solving transport models in heterogeneous and anisotropic porous media for practical applications such as groundwater flow, reservoir flow, shallow water equation, and others.

Acknowledgement

The work presented in this paper was supported by the Federal Government of the Republic of Nigeria through PTDF/ED/ISS/PHD/JAK/2045/21.

References

[1] M. Zhinjuan, C. Xiaofei, and M. Lidong, "A Hybrid Interpolating Meshless Method for 3D Advection Diffusion Problems," *Mathematics,* MDPI, 2022.

[2] Z. Yongou, D. Sina, L. Wei, and C. Yingbin, "Performance of the radial point interpolation method (RPIM) with implicit time integration scheme for transient wave propagation dynamics," *Computers & Mathematics with Application*. Elsevier. Vol. 114, pp. 95-111, 2022.

[3] E. Oñate and M. Manzan, "A general procedures for deriving stabilized space-time finite elements method for advective-diffusive problems," *Int. J. Numer. Meth. Fluids*. Vol. 31, pp. 203-221, 2015.

[4] J. Carrer, C. Curha and W. J. Mansur, "The boundary element method applied to the solution of two-dimensional diffusion-advection problems for non-isotropic materials," *J. Braz. Soc. Mech. Sci*. Vol. 39, pp. 4533-4545, 2017.

[5] E. Barbieri and N. Petrinic, "Three-dimensional crack propagation with distance-based discontinuous kernels in meshfree methods,". *Comput. Mech*. Vol. 53, pp. 325-342, 2014.

[6] P. N. Vinh, R. Timon, B. Stephane, and D. Mac, "Meshless methods: A review and computer implementation aspects,". *Mathematics and Computers in Simulation*. Elsevier. Vol. 79. pp. 763-813, 2008.

[7] Q. Wu, P. P. Peng and Y. M. Cheng, "The Interpolating element-free Galerkin method for elastic large deformation problems,". *Sci. China*. Vol. 64, pp. 364-374, 2021.

[8] H. Sirajul, H. Arshad and U. Maryam, "RBFs Meshless Method of Lines for the Numerical Solution of Time-Dependent Nonlinear Coupled Partial Differential Equations," *Applied Mathematics*, Scientific Research, 2001.

[9] T. Belytschko, Y. Y. Lu, and L. Gu, "Element-free Galerkin Methods. *International Journal for Numerical Methods in Engineering,*" vol. 37, no. 2, pp. 229–256, 1994.

[10] M. Zhijuan, C. Xiaofei and M. Lidong, A Hybrid Interpolating Meshless Method for 3D Advection-Diffusion Problems. *Mathematics,* MDPI, 2022.

[11] B. Nayroles, G. Touzot and P. Villon, "Generalizing the finite element method: Diffuse Approximation and diffuse elements." *Comput. Mech.* Vol. 10, no. 5, pp 307-318, 1992.

[12] G.R. Liu and Y.T. Gu, "A point interpolation method for two-dimensional solids," *Int. J. Numerical Methods Eng*.," Vol. 50, pp. 937-951, 2001.

[13] C. Heng and Z. Guodong, "Analyzing 3D Advection-Diffusion Problems by Using theImproved Element-Free Galerkin Method," *Mathematical Problems in Engineering*, Hindawi, 2020.

[14] M. Mategaonkar and T.T. Eldho, "Two dimensional contaminant transport modelling using meshfree point collocation method (PCM)," *Engineering Analysis with Boundary Elements*. Elsevier, 2012.

[15] B. Nagina, I. A. T. Syed, and H. Sijarul, "Meshless Method of Lines for Numerical Solution of Kawahara Type Equations," *Applied Mathematics*. Scientific Research, 2011.

[16] S. N. Atluri and T. Zhu, "A new meshless local Petrov–Galerkin (MLPG) approach in Computational mechanics," *Comput. Mech*., Vol. 22, pp. 117–127, 1998.

[17] E. Oñate, S. Idelsohn, O. C. Zienkiewicz, and R. L. Taylor, "A finite point method in computational mechanics applications to convective transport and fluid flow," *Int. J.*

Numerical Methods Eng., Vol. 39, pp. 3839–3866, 1996.

[18] G.R. Liu, "*Meshfree Methods: Moving beyond the Finite Element Method,"* Taylor and Francis Group, LLC, New York. 2010.

[19] M. B. Liu, G. R. Liu, and Z. Zong, "An overview on smoothed particle Hydrodynamics," *Int. J. Comput. Methods*, Vol. 5, no 1. pp. 135–188, 2008.

[20] [20] W. K. Liu, S. Jun and Y. Zhang, "Reproducing kernel particle methods," *Int. J. Numerical*
Methods Fluids, Vol. 20, pp. 1081–1106, 1995.

[21] J. G. Wang and G. R. Liu, "A point interpolation meshless method based on radial basis Functions," *International Journal for Numerical Methods in Engineering*, 2002.

[22] M. J. D. Powell, "*The theory of radial basis function approximation in 1990, in Advances in Numerical Analysis,"* F. W. Light, ed., Oxford University Press, Oxford, pp. 303–322,1992.

[23] H. Wendland, "Error estimates for interpolation by compactly supported radial basis functions of minimal degree," *J. Approximation Theory*. Vol. 93. pp 258–396. 1998

[24] E. J. Kansa, "A scattered data approximation scheme with application to computational Fluid dynamics I & II, *Comput. Math. Appl*. Vol. 19, pp. 127–161, 1990.

[25] Z. Yandaizi, W. Tielin and Z. Jesse, "Development of gas-solid fluidization: Particulate and aggregative," *Powder Technology*. Elsevier. Vol. 421, 2023.

[26] S. Das and T. I. Eldho, "A Meshless Weak-Strong Form Method for the Simulation of Coupled Flow and Contaminant Transport in an Unconfined Aquifer," *Transport in Porous Media*. Springer. Vol. 143, pp. 703 – 737, 2022.

[27] E. O$\bar{n}$ate, S. Idelson, O. C. Zienkiewicz, R. L. Taylor R. L. and C. Saco, "A Stabilized finite point method for analysis of fluid mechanics problems," *Comp. Methods Appl. Mech and Eng*, Elsevier. Vol. 139. pp. 315 – 346, 1996.

[28] E. O$\bar{n}$ate, "Finite Increment Calculus (FIC): a framework for deriving enhanced Computational methods in mechanics," *Advanced Modelling and Simulation in Engineering Science*, CIMNE, Barcelona, Spain, 2016.

[29] A. Aatish and T. I. Eldho, "Simulation of flows and Transport process – Scope of Meshlesh Methods: Recent Advances in Computational Mechanics," Lecture Notes in Mechanical Engineering. Springer. Singapore, 2021.

[30] Q. Xinqiang, H. Gang and P. Gaosheng, "Finite Point Method of Nonlinear Convective Diffusion Equation," *Filomat*. Vol. 5, no 34, pp. 1517 – 1533, 2020.

[31] A. R. Appadu and H. H. Gidey, "Time-Splitting Procedures for the Numerical Solution of the 2D Advection-Diffusion Equation," *Mathematical Problems in Engineering*, Hindawi Publishing Corporation. Article 1D 634657, 2013.

[32] J. A. Kazeem, "A Meshfree Method for the Solutions of Multiphase Flow Problems in Porous Media," Unpublished doctoral thesis, University of Abuja, F.C.T., Nigeria, 2025.

[33] A. M. S. Muhannad and F. J. Borhan, "Numerical Solutions Based on Finite Difference Technique for Two-Dimensional Advection-Diffusion Equation," *British Journal of Mathematics and Computer Science*. Article no: BJMCS.25464, 2016.

[34] W. Abdul Majid, "*Linear and Nonlinear Integral Equation: Methods and Applications*," Springer-Verlag, New York, (2011).

[35] N. Okiotor, F. Ogunfiditimi, and M. O. Durojaye, "On the computation of the lagrange multiplier for the variational iteration method (VIM) for solving differential equations," *Journal of Advances in Mathematics and Computer Science*, Vol. *35,* no 3, pp. 74-92, 2020.

[36] H. P. Albertus, "The Lagrangian and Hamiltonian for RLC Circuit: Simple Case," *International Journal of Applied Sciences and Smart Technologies.* Vol. 2, no 2, pp. 169–178, 2020.

[37] D. R. Kirubaharan, A. D. Subhashini, N. V. N. Babu, and G. Murali, "Quantitative Analysis of Magnetohydrodynamic Sustained Convective Flow via Vertical Plate,". *International Journal of Applied Sciences and Smart Technologies.* Vol. 6, no 2, pp. 407– 416, 2024.

[38] M. K. Singh, P. Singh and V. P. Singh, "Analytical Solution for Two-Dimensional Solute Transport in Finite Aquifer with Time-Dependent Source Concentration," *Journal of Engineering Mechanics*. Vol 136, no 10, 2010.

[39] P. Singh, "One Dimensional Solute Transport Originating from a Exponentially Decay Type Point Source along Unsteady Flow through Heterogeneous Medium," *Journal of Water Resource and Protection*. Vol. 3, pp. 590-597, 2011.

[40] G. D. Hutomo, J. Kusuma, A. Ribal, A. G. Mahie and N. Aris, "Numerical solution of 2-dadvection-diffusion equation with variable coefficient using du-fort frankel method," *J. Phys: Conf. Ser.1180 012009,* 2019.

This page intentionally left