

Article

Numerical Inverse Laplace Transform Methods for Advection-Diffusion Problems

Kamran ^{1,*}, Farman Ali Shah ¹, Wael Hosny Fouad Aly ², Hasan Aksoy ³, Fahad M. Alotaibi ⁴
and Ibrahim Mahariq ²¹ Department of Mathematics, Islamia College Peshawar, Peshawar 25120, Khyber Pakhtoon Khwa, Pakistan² College of Engineering and Technology, American University of the Middle East, Egaila 54200, Kuwait³ Department of Electrical and Electronic Engineering, University of Turkish Aeronautical Association, Ankara 06790, Turkey⁴ Department of Information Systems, Faculty of Computing and Information Technology (FCIT), King Abdulaziz University, Jeddah 34025, Saudi Arabia

* Correspondence: kamran.maths@icp.edu.pk

Abstract: Partial differential equations arising in engineering and other sciences describe nature adequately in terms of symmetry properties. This article develops a numerical method based on the Laplace transform and the numerical inverse Laplace transform for numerical modeling of diffusion problems. This method transforms the time-dependent problem to a corresponding time-independent inhomogeneous problem by employing the Laplace transform. Then a local radial basis functions method is employed to solve the transformed problem in the Laplace domain. The main feature of the local radial basis functions method is the collocation on overlapping sub-domains of influence instead of on the whole domain, which reduces the size of the collocation matrix; hence, the problem of ill-conditioning in global radial basis functions is resolved. The Laplace transform is used in comparison with a finite difference technique to deal with the time derivative and avoid the effect of the time step on numerical stability and accuracy. However, using the Laplace transform sometimes leads to a solution in the Laplace domain that cannot be converted back into the real domain by analytic methods. Therefore, in such a case, the Laplace transform is inverted numerically. In this investigation, two inversion techniques are utilized; (i) the contour integration method, and (ii) the Stehfest method. Three test problems are used to evaluate the proposed numerical method. The numerical results demonstrate that the proposed method is computationally efficient and highly accurate.

Keywords: Laplace transform; diffusion problems; local radial basis functions; contour integration method; trapezoidal rule; Stehfest's method



Citation: Kamran; Shah, F.A.; Aly, W.H.F.; Aksoy, H.; Alotaibi, F.M.; Mahariq, I. Numerical Inverse Laplace Transform Methods for Advection-Diffusion Problems. *Symmetry* **2022**, *14*, 2544. <https://doi.org/10.3390/sym14122544>

Academic Editors: Ji-Huan He and Muhammad Nadeem

Received: 6 November 2022

Accepted: 27 November 2022

Published: 1 December 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Advection-diffusion equations (ADEs) have an important role in many physical systems, i.e., vorticity, heat, energy, mass and fluid flow [1]. Symmetry is a fundamental property of nature and its phenomena; in physics, it ensures the existence of conservation laws, and, in mathematics, it means a transformation from one solution to another [2]. ADEs are able to adequately describe physical and biological processes and have symmetry properties [3]. Furthermore, many physical phenomena relating to the transformation of energy during advection and diffusion processes can be described by ADEs [4]. It is a 1D PDE which demonstrates the advection and diffusion of mass, vorticity, heat and energy [5], etc. ADEs can be utilized to describe heat transfer in draining film [6], the dispersion of traces in porous media [7], the transfer of pollutants in the atmosphere [8], and water transmission in soil [9]. In this investigation, we consider an ADE of the form [10,11]:

$$\frac{\partial C(x,t)}{\partial t} + \bar{\theta} \frac{\partial C(x,t)}{\partial x} - \bar{\alpha} \frac{\partial^2 C(x,t)}{\partial x^2} = h(x,t), \quad x \in \Omega, \quad t > 0, \quad (1)$$

with initial condition

$$C(x, 0) = C_0(x), \quad x \in \Omega,$$

and Dirichlet boundary conditions

$$\mathcal{L}_B C(x, t) = g(x, t), \quad x \in \partial\Omega, \quad (2)$$

where Ω is the domain and $\partial\Omega$ is its boundary, $\mathcal{L} = \frac{\partial}{\partial x} + \frac{\partial^2}{\partial x^2}$ and \mathcal{L}_B are the linear differential and boundary operators; $\bar{\theta}$ and $\bar{\alpha}$ are the advection and diffusion coefficients, respectively, where $\bar{\theta}$ is an arbitrary constant and $\bar{\alpha}$ is a positive constant. Whereas, $C(x, t)$ is the concentration and $h(x, t)$, $g(x, t)$ are given smooth functions.

ADEs have been studied by many authors, and various approximate and analytic techniques have been developed for the solution of 1D ADEs [12]. Kumar et al. [13,14] studied the analytic solution of 1D ADEs with variable coefficients. Zoppou and Knight [15] investigated the analytic solution of an ADE with variable coefficients. In [16], the analytic solution of 2D ADE, arising in cytosolic calcium concentration distribution, was examined. Further information on other analytic methods for ADEs can be found in [17–19] and the references therein.

However, in many situations, the analytic solution of ADEs is difficult to obtain. Numerical techniques are then used to approximate the solution of ADEs. The numerical investigation of the solutions of ADEs has been considered by many researchers, and numerous methods have been proposed. These include the stable explicit finite difference method (FDM) [20], the implicit FDM [21], the compact FDM [10], the cubic trigonometric B-splines method [11], the finite element method (FEM) [22], and the boundary element method [23]. The numerical solutions of ADEs are difficult to obtain due to: (i) the first- and second-order derivatives with respect to space variables. Depending on the values of $\bar{\alpha}$ and $\bar{\theta}$, the equation becomes parabolic for diffusion-dominated and hyperbolic for advection-dominated processes. Traditionally used FDMs face problems when oscillations and smoothing of the wave front are introduced [24,25]; (ii) since all the methods depend on a mesh, it is necessary to construct a fine mesh for optimal results. However, mesh generation is the most time-consuming part of the solution process and, for complex geometries, problems can occur with implementation.

Due to the complexity of mesh generation, remarkable work has recently been undertaken on the development of mesh-free or mesh-less methods. Numerous mesh-free methods for approximating the solutions of PDEs in different fields of engineering and other fields have been developed. The major advantages of mesh-less methods are: (i) they do not require a mesh, which can be challenging in 3D-space cases; (ii) they are more suitable than mesh-based methods in cases of large deformations or moving discontinuities. One of the common features of all mesh-less methods is their ability to construct a functional interpolation or approximation entirely from information about a set of dispersed nodes between which there is no predetermined connectivity or relationship. Various mesh-less methods have been developed to date. These include, the Galerkin mesh-free methods [26], the boundary knot method [27], the singular boundary method [28], and the local point interpolation method [29].

Another group of mesh-less methods that are based on radial basis functions (RBFs), represents one of the best tools for obtaining the numerical solutions to various real world problems [30–34]. The main characteristics of RBFs are their smoothness, spectral convergence and ease of implementation. RBF-based mesh-less methods are divided into two categories: global mesh-less methods (GRBFMs) and local mesh-less methods (LRBFMs). In GRBFMs, the solution is approximated using the linear combinations of RBFs; these methods involve full system matrices that are often ill-conditioned. Due to the involvement of full system matrices the approximation of large-scale problems becomes difficult for some RBF methods. The condition number of the system matrices increases as the number of interpolation nodes increases. Another factor is the use of infinitely smooth basis functions with arbitrary shape parameters [35]. The accuracy of the approximation

and the conditioning of the system matrix depends on the value of the shape parameter. For GRBFMs, how to select an optimal shape parameter is still an open question [36].

In 2003, Shu et al. [37] proposed LRBFMs to solve 2D Navier–Stokes equations. In LRBFMs, the collocation is made in overlapping sub-domains, which reduces the size of the system matrix, and then these small system matrices are solved for each node. Subsequently, LRBFMs have been applied to numerous problems in engineering and other science subjects, such as compressible flow problems [38], incompressible flow problems [39], arbitrarily shaped membranes [40], boundary value problems [41], and diffusion problems [42]. The main drawback of LRBFMs is that this method does not work for elliptic problems in a straightforward way. In addition, these LRBFMs are employed to approximate the solutions of time-dependent PDEs coupled with a time-stepping scheme. The drawback of the time-stepping technique is that it does not always lead to a stable solution. The finite difference time-stepping scheme is stable if the errors remain constant or decay during computation. In addition, in the finite difference time-stepping scheme, the accuracy is achieved at a very small time step and, hence, this scheme encounters an exponential increase in computing costs. To overcome this drawback of the time-stepping technique, the Laplace transform (LT) is used.

Some remarkable work has been published in the literature on the combination of LT with different space-discretization methods to avoid time-instability problems. Examples of this approach include the boundary particle method [43], Kansa’s method [44], the LRBFM on unit sphere method [45], the FDM [46], the spectral method [47], and the FEM [48,49]. In this study, we propose LRBFMs coupled with LT for the approximation of the solution of 1D linear ADEs. For the numerical inversion, we use two well-known methods: (i) the contour integration method; and (ii) Stehfest’s method.

2. Implementation of the Method

In our numerical scheme, there are three main steps: (a) implementation of the Laplace transform, (b) implementation of a local RBF method, and (c) numerical inversion of the Laplace transform.

2.1. (a): Laplace Transform (LT)

The LT transforms the time-dependent advection diffusion equation from a time domain to an inhomogeneous time-independent problem to an LT domain.

Let $C(x, t)$, $t > 0$ be a piecewise continuous, and let $\widehat{C}(x, s)$ be its LT, which is defined as

$$\widehat{C}(x, s) = \mathcal{L}\{C(x, t)\} = \int_0^{\infty} e^{-st} C(x, t) dt, \quad (3)$$

where s denotes the LT parameter. The LT of $\frac{dC(x, t)}{dt}$ is given as

$$\mathcal{L}\left\{\frac{dC(x, t)}{dt}\right\} = s\widehat{C}(x, s) - C_0 \quad (4)$$

Employing the LT to (1) and (2), we get

$$s\widehat{C}(x, s) - C_0 - \mathcal{L}\widehat{C}(x, s) = \widehat{h}(x, s), \quad (5)$$

and

$$\mathcal{B}\widehat{C}(x, s) = \widehat{g}(x, s). \quad (6)$$

Simplifying (5) and (6), we get

$$(sI - \mathcal{L})\widehat{C}(x, s) = \widehat{G}_1(x, s), \quad (7)$$

$$\mathcal{L}_B\widehat{C}(x, s) = \widehat{g}(x, s), \quad (8)$$

where

$$\widehat{G}_1(x, s) = C_0 + \widehat{h}(x, s),$$

where I is the identity operator, \mathcal{L} is the linear differential operator and \mathcal{L}_B is the boundary differential operator. To solve the system (7) and (8), first, we discretize the operators \mathcal{L} and \mathcal{L}_B using LRBFM, then the system (7) and (8) is solved for each s in the LT domain. Finally, the solution of the problem can be obtained via inverse LT.

2.2. (b): Local RBF Method

Here, we propose a local RBF method for approximating the 1D ADEs in Laplace space. This method provides a linear system that is sparse and well-conditioned. For the set $\{\widehat{C}(x_i) : i = 1, 2, 3, \dots, N\}$, where $\{x_i : i = 1, 2, 3, \dots, N\} \subset \Omega$, the local RBF approximation of $\widehat{C}(x)$ is of the form,

$$\widehat{C}(x_i) = \sum_{x_j \in \Omega_i} \alpha_j^i \varphi(\|x_i - x_j\|), \tag{9}$$

where $\alpha^i = \{\alpha_j^i\}_{j=1}^n$, $\varphi(r)$, $r > 0$ is the radial kernel. Ω_i is the sub-domain which contains the node x_i and its neighboring n nodes around it. Thus, we get N number linear systems of order $n \times n$ as

$$\begin{pmatrix} \widehat{C}(x_1^i) \\ \widehat{C}(x_2^i) \\ \vdots \\ \widehat{C}(x_n^i) \end{pmatrix} = \begin{pmatrix} \varphi_{11} & \varphi_{12} & \cdot & \cdot & \cdot & \varphi_{1n} \\ \varphi_{21} & \varphi_{22} & \cdot & \cdot & \cdot & \varphi_{2n} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \varphi_{n1} & \varphi_{n2} & \cdot & \cdot & \cdot & \varphi_{nn} \end{pmatrix} \begin{pmatrix} \alpha_1^i \\ \alpha_2^i \\ \cdot \\ \cdot \\ \alpha_n^i \end{pmatrix}, \quad i = 1, 2, 3, \dots, N, \tag{10}$$

which can be written as,

$$\widehat{C}^i = \Phi^i \alpha^i, \quad i = 1, 2, 3, \dots, N, \tag{11}$$

the matrix Φ^i has entries $d_{kj}^i = \varphi(\|x_k^i - x_j^i\|)$, where $x_k^i, x_j^i \in \Omega_i$; each of the above systems is then solved for the $\alpha^i = \{\alpha_j^i\}_{j=1}^n$. The index i denotes that the nodes belong to the local domain Ω_i of each node x_i . In addition, for \mathcal{L} we have,

$$\mathcal{L}\widehat{C}(x_i) = \sum_{x_j \in \Omega_i} \alpha_j^i \mathcal{L}\varphi(\|x_i - x_j\|), \tag{12}$$

From Equation (12), we have

$$\mathcal{L}\widehat{C}(x_i) = \mathbf{v}^i \cdot \alpha^i, \tag{13}$$

where α^i is a column vector of order n and \mathbf{v}^i is a row vector of order n having elements of the form

$$\mathbf{v}^i = \mathcal{L}\varphi(\|x_i - x_j^i\|), \quad x_j^i \in \Omega_i, \tag{14}$$

From Equation (11), we have

$$\alpha^i = (\Phi^i)^{-1} \widehat{C}^i, \tag{15}$$

From (13) and (15), we have

$$\begin{aligned} \mathcal{L}\widehat{C}(x_i) &= \mathbf{v}^i (\Phi^i)^{-1} \widehat{C}^i \\ &= \Theta^i \widehat{C}^i, \end{aligned} \tag{16}$$

where,

$$\Theta^i = \mathbf{v}^i (\Phi^i)^{-1}, \tag{17}$$

thus, the operator \mathcal{L} has an approximation via local radial basis functions at each node x_i as

$$\mathcal{L}\widehat{C} \equiv \mathbf{D}\widehat{C}, \tag{18}$$

where D denotes the sparse matrix of order N which approximates the linear differential operator \mathcal{L} . A similar procedure can be applied to the boundary operator \mathcal{L}_B as

$$\mathcal{L}_B \hat{C} \equiv \mathbf{B}\hat{C}, \quad (19)$$

substituting Equations (18) and (19) in Equations (7) and (8), we get

$$(sI - \mathbf{D})\hat{C}(x, s) = \hat{G}_1(x, s), \quad (20)$$

$$\mathbf{B}\hat{C}(x, s) = \hat{g}(x, s), \quad (21)$$

Solving the system (20) and (21) for each point s , we will obtain the approximate solution $\hat{C}(x, s)$. Finally, we obtain the solution $C(x, t)$ of the problem defined in Equations (1) and (2) using the numerical inverse LT.

2.3. (c): Inverse Laplace Transform

This section is about numerical inverse LT methods. In this investigation, we use two numerical inverse LT methods: (i) the contour integration method; and (ii) Stehfest's method.

(i) Contour Integration Method

In the contour integration method, the solution $C(x, t)$ is obtained via the inverse LT given as

$$C(x, t) = \frac{1}{2\pi i} \int_{\zeta-i\infty}^{\zeta+i\infty} e^{st} \hat{C}(x, s) ds = \frac{1}{2\pi i} \int_{\Gamma} e^{st} \hat{C}(x, s) ds, \quad t > 0, \quad (22)$$

where Γ is an appropriately selected contour of integration in a complex plane. The solution of the problem (1) and (2) mainly depends on Γ . For optimal solution of (22), an optimal contour parabolic or hyperbolic may be utilized. Various contour integration methods are developed for the approximation of the inverse Laplace transform. However, in this study, we utilize two recently proposed contours in [47,48]. The parabolic contour is given as

$$s(v) = \eta(1 + iv)^2 \quad (\Gamma_p), \quad (23)$$

where η is an unknown parameter. In [48], the authors proposed a contour given as

$$s(v) = \chi + \varepsilon - \chi \sin(\zeta - iv), \quad \text{for } -\infty < v < \infty, \quad (\Gamma_h) \quad (24)$$

where $\chi > 0$, $\frac{\pi}{2} < \alpha_1 < \pi$, $0 < \zeta < \alpha_1 - \frac{\pi}{2}$, $\varepsilon \geq 0$.

Using Equations (23) and (24) in Equation (22), we have

$$C(x, t) = \frac{1}{2\pi i} \int_{-\infty}^{\infty} e^{s(v)t} \hat{C}(x, s(v)) s'(v) dv \quad (25)$$

Equation (25) is solved via the trapezoidal rule, with step k as

$$C_{App}(x, t) = \frac{1}{2\pi i} \sum_{j=-M_p}^{M_p} k e^{s(v_j)t} \hat{C}(x, s(v_j)) s'(v_j), \quad v_j = jk. \quad (26)$$

2.4. Error Analysis

2.4.1. Parabolic Contour

Here, we discuss the error analysis of our scheme. We consider the integral

$$I = \int_{-\infty}^{\infty} G(\vartheta) d\vartheta,$$

and its infinite and finite approximations

$$I_k = k \sum_{j=-\infty}^{\infty} G(jk), \quad I_{k;M_p} = k \sum_{j=-M_p}^{M_p} G(jk).$$

The quantities $E_T = |I_{k;M_p} - I_k|$ and $E_d = |I - I_k|$ are called the truncation and discretization errors. The proof of the discretization error of the proposed numerical scheme is based on the following theorem:

Theorem 1 ([47]). *Let $s = \xi + i\rho$, where $\xi, \rho \in \mathbb{R}$. Let $G(s)$ be analytic in the strip $-a_1 < \rho < a_2$, for some $a_1, a_2 > 0$, with $G(s) \rightarrow 0$, as $|s| \rightarrow \infty$. Let for $Q_+, Q_-, G(s)$ satisfies*

$$\int_{-\infty}^{\infty} |G(\xi + i\rho)| d\xi \leq Q_+, \tag{27}$$

$$\int_{-\infty}^{\infty} |G(\xi - i\rho)| d\xi \leq Q_-, \tag{28}$$

$\forall 0 < r < a_1, 0 < z < a_2$. Then

$$|I_k - I| \leq \frac{M_p(+)}{\exp(\frac{2\pi a_1}{k}) - 1} + \frac{M_p(-)}{\exp(\frac{2\pi a_2}{k}) - 1}.$$

Remark 1. *In the literature, G is typically a real valued function, $a_1 = a_2, Q_+ = Q_- = Q$, say, then we obtain an estimate given as*

$$E_d = |I_k - I| \leq \frac{2Q}{\exp(\frac{2\pi a_1}{k}) - 1}.$$

For the best approximation of E_d in [47], the authors obtained the best parameters. They obtained the optimal values of η and k by asymptotically equalizing the discretization and the truncation errors. The optimal values of the parameters given by

$$k = \frac{1}{M_p(1 + 8\omega)^{1/2}}, \quad \varrho = \frac{\pi M_p}{4T(8\omega + 1)^{1/2}}, \quad \omega = \frac{T}{t_0},$$

thus, we have the following estimate

$$\Gamma_p(est) = |C_{App}(x, t) - C(x, t)| = O\left(e^{-\left(2\pi M_p \frac{1}{\sqrt{8\omega+1}}\right)}\right), \tag{29}$$

For more details, readers are referred to [47]. In our numerical experiments, M_p is defined as $m = \frac{2M_p}{k}$, where k and m correspond to the quadrature step and the nodes, respectively.

2.4.2. Hyperbolic Contour

For the contour Γ_h , the authors in [48] obtained the optimal parameters, with the error estimate given as

$$\Gamma_h(est) = |C_{App}(x, t) - C(x, t)| = O\left(\ell(\sigma_r M_h) e^{-\omega M_h}\right), \tag{30}$$

where $\omega = \frac{\bar{r}}{b}(1 - \vartheta)$, $\bar{r} = 2r\pi$, $0 < \vartheta < 1$, $\tau = t_0/T$, $\ell(x) = \max(\log(1/x), 1)$, $\sigma_r = (\tau \sin(\theta - r)) \frac{\vartheta \bar{r}}{\mathcal{U}}$, where \mathcal{U} satisfies $\cosh(\mathcal{U})(\vartheta \tau \sin(\theta)) = 1$, $k = \mathcal{U}/M_h$, $0 < t_0 \leq t \leq T$, and $\chi = \vartheta \bar{r} M_h / (\mathcal{U} T)$.

2.4.3. (ii) Stehfest’s Method

In this section, we utilize Stehfest’s method for inverting the LT numerically. Stehfest’s method was developed in the 1960s. It is one of the most popular methods for numerical inversion of the LT. This method is fast and gives good results for smooth functions. The solution $C(x, t)$ can be approximated via Stehfest’s method as

$$C_{App}(x, t) = \frac{\ln 2}{t} \sum_{i=1}^{M_s} w_i \widehat{C}(x, s), \tag{31}$$

where w_i are given by

$$w_i = (-1)^{\frac{M_s}{2} + i} \sum_{h=\lfloor \frac{i+1}{2} \rfloor}^{\min(i, \frac{M_s}{2})} \frac{h^{\frac{M_s}{2}} (2h!)}{(\frac{M_s}{2} - h)! h! (h - 1)! (i - h)! (2h - i)!}. \tag{32}$$

Here, M_s , referred to as Stehfest’s number, should be even. Solving (20) and (21) for the parameters $s = \frac{\ln 2}{t} i, i = 1, 2, \dots, M_s$ in the Laplace domain. Using (31), the original solution can be obtained.

The Gaver–Stehfest algorithm has some attractive properties: (i) the approximations of any function using this algorithm are linear in values of the transform; (ii) it requires the values of the transform function for real s only; (iii) the computation of the coefficients is very easy; and (iv) the approximations using this algorithm are exact for constant functions. In the literature, use of this algorithm is reported by many authors [50,51], where it has been demonstrated that the method converges very rapidly for non-oscillatory functions.

2.4.4. Error Analysis

In this section, we present an error analysis of our proposed numerical scheme. In step (a), we use the LT which is analytic and in this step no error occurs. Errors in our method occur in step (b) and (c). In step (b), we use the local RBs method, which has an error estimate $O(\sigma^{\frac{1}{\varepsilon Q_d}})$, $0 < \sigma < 1$, where Q_d and ε denote the fill distance and the shape parameter [52]. In step (c), Stehfest’s numerical inversion for the LT is used. An error analysis of this method is described in [53,54], in which the authors report the effect of parameters on the numerical efficiency and accuracy. They concluded that “If β significant digits are required, suppose $M_s = \lceil 2.2\beta \rceil$ positive integer. Set the precision of the system at $v = \lceil 1.1M \rceil$. Then given the transform $\widehat{C}(x, s)$ and the argument t , calculate $C_{App}(x, t)$ in (31)”. On the basis of these conclusions, the following error estimation is obtained:

Remark 2 ([43]). *If the error of the input data is $10^{-v+1} \leq \frac{\|\widehat{C} - C_{App}\|}{\|C_{App}\|} \leq 10^{-v}$, with M_s a positive even integer via $v = \lceil 1.1M_s \rceil$. Then the error is $10^{-\beta+1} \leq \frac{\|\widehat{C} - C_{App}\|}{\|C_{App}\|} \leq 10^{-\beta}$, where $M_s = \lceil 1.1\beta \rceil$.*

The key steps of the proposed numerical scheme are presented in the following Algorithm 1:

Algorithm 1: The necessary steps of the proposed numerical scheme

- 1: **Input:** The computational domain, the time derivative in $[0, 1]$, the final time, the initial shape parameter and the other parameter of the given model, the inhomogeneous function, and initial boundary conditions.
- 2: **Step 1:** Reduce the problem defined in Equations (1) and (2) to an equivalent time-independent problem defined in Equations (7) and (8) by employing the Laplace transform.
- 3: **Step 2:** Discretize the linear differential operator \mathcal{L} and boundary operator \mathcal{L}_B using Equations (18) and (19).
- 4: **Step 3:** Solve the system of Equations (20) and (21) in parallel for each point s .
- 5: **Step 4:** Compute the approximate solution using (26) or (31).
- 6: **Output:** The approximate solution is $C_{App}(x, t)$.

3. Stability

In this section, we discuss the stability of the system (20) and (21) by writing it in discrete form as

$$\widehat{\mathbf{G}}\mathbf{C} = \mathbf{p}, \quad (33)$$

where \mathbf{G} is obtained using the local radial basis functions method. The stability constant for the discrete system (33) is defined as

$$\sigma = \sup_{\widehat{\mathbf{C}} \neq 0} \frac{\|\widehat{\mathbf{C}}\|}{\|\widehat{\mathbf{G}}\widehat{\mathbf{C}}\|}, \quad (34)$$

where the value of the constant σ is finite for any discrete norm $\|\cdot\|$ defined on R^N . Equation (34) implies

$$\|\mathbf{G}\|^{-1} \leq \frac{\|\widehat{\mathbf{C}}\|}{\|\widehat{\mathbf{G}}\widehat{\mathbf{C}}\|} \leq \sigma. \quad (35)$$

For the pseudoinverse \mathbf{G}^\dagger of \mathbf{G} , we have

$$\|\mathbf{G}^\dagger\| = \sup_{\mathbf{v} \neq 0} \frac{\|\mathbf{G}^\dagger \mathbf{v}\|}{\|\mathbf{v}\|}. \quad (36)$$

Therefore,

$$\|\mathbf{G}^\dagger\| \geq \sup_{\mathbf{v}=\widehat{\mathbf{G}}\widehat{\mathbf{C}} \neq 0} \frac{\|\mathbf{G}^\dagger \widehat{\mathbf{G}}\widehat{\mathbf{C}}\|}{\|\widehat{\mathbf{G}}\widehat{\mathbf{C}}\|} = \sup_{\widehat{\mathbf{C}} \neq 0} \frac{\|\widehat{\mathbf{C}}\|}{\|\widehat{\mathbf{G}}\widehat{\mathbf{C}}\|} = \sigma. \quad (37)$$

Equations (35) and (37) give the upper and lower bounds for σ . For the system (33), the pseudoinverse may be difficult to calculate, but it ensures the stability. We can utilize the command `condst` in MATLAB for estimating $\|\mathbf{G}^{-1}\|_\infty$, thus

$$\sigma = \frac{\text{condst}(\mathbf{G}')}{\|\mathbf{G}\|_\infty} \quad (38)$$

4. Numerical Examples

To validate the proposed numerical methods, we consider three test problems. The MQ RBFs defined by $\varphi = \sqrt{r^2 + \vartheta^2}$ are used in this paper. For the optimal shape parameter, the uncertainty principal due to [55] is utilized. We performed our experiments in MATLAB R2018a on a Windows 10 (64 bit) PC equipped with an Intel(R) Core(TM) i5-3317U CPU @ 1.70 GHz and with 4 GB of RAM. The performance of our method is tested using the L_∞ error norm which is denoted and defined by

$$Error_{max} = \|C(x_i, t) - C_{App}(x_i, t)\|_\infty,$$

where $C_{App}(x, t)$ and $C(x, t)$ are the numerical and analytical solutions, respectively.

4.1. Problem 1

Here, we consider a 1D diffusion equation of the form

$$\frac{\partial C(x, t)}{\partial t} - \frac{\partial^2 C(x, t)}{\partial x^2} = \exp(x)(2t - t^2), \quad t > 0, \quad 0 < x < 1, \tag{39}$$

with the initial and boundary data

$$C(x, 0) = 0,$$

and

$$\begin{aligned} C(0, t) &= t^2, \\ C(1, t) &= \exp(1)t^2. \end{aligned}$$

The analytical solution is

$$C(x, t) = \exp(x)t^2.$$

The results obtained along the parabolic contour are shown in Table 1, and along the hyperbolic contour are shown in Table 2. Similarly, the results obtained using Stehfest’s method are presented in Table 3. For the contours, we obtained the quadrature nodes using the MATLAB command $v = -M_h : k : M_h$. In our experiments, different parameter values are used which are $\zeta = 0.15410, \chi = 2, c = 0.1, r = 0.13870, \vartheta = 0.10$, and $t \in [0.5, 5]$. Various nodes in the local and global domains are selected for the numerical experiments. Figure 1a–d depicts the numerical and exact solutions of problem 1, computed using the hyperbolic contour with parameter values $M_h = 50, N = 30, n = 12$. Figure 2a,b displays the exact and numerical space-time solutions of problem 1, computed along the hyperbolic contour with parameter values $M_h = 50, N = 40, n = 12$. Figure 3a shows a comparison between the errors of problem 1 obtained using the three inversion techniques. It is observed that the performance of the two contour integration methods is better than for Stehfest’s method. Figure 3b shows the plot of error versus the nodes M_p along the parabolic contour. Similarly Figure 4a shows the plot of error versus M_h using the hyperbolic contour. A comparison between the observed and theoretical error is shown in Figure 4b; a good agreement between the error and the error estimate is observed. Figure 4c shows a plot of error versus M_s using Stehfest’s method. In Figure 5a, the plot shows error versus t using the hyperbolic path. Figure 5b depicts a plot of error versus t using Stehfest’s method. From all the figures and tables, a very high accuracy and stability is evident. However, it is observed that the results along the hyperbolic path are more accurate than that of the results along the parabolic path and using Stehfest’s method. The results demonstrate that the method is capable of approximating the solution of this type of equation efficiently.

Table 1. The $Error_{max}$ using the proposed scheme along the path Γ_p , at $t = 1$.

M_p	N	n	$Error_{max}$	ϑ	C. Time (s)
30	70	10	1.18×10^{-2}	6.4	0.353447
40			2.55×10^{-4}	6.4	0.416162
50			2.46×10^{-4}	6.4	0.538895
60	20	8	3.60×10^{-5}	1.4	0.271687
	30		5.01×10^{-5}	2.1	0.324185
	40		6.04×10^{-5}	2.9	0.386367

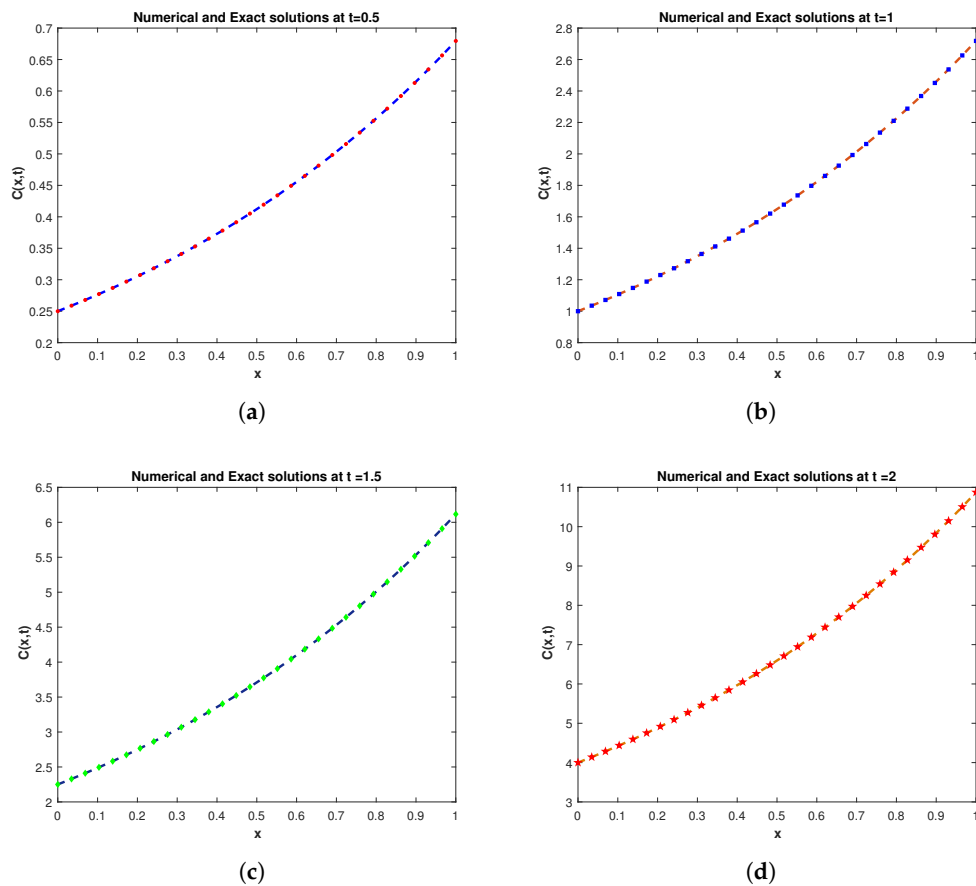


Figure 1. Numerical and exact solutions of problem 1 at different time levels with $M_h = 50$, $N = 30$, $n = 12$.

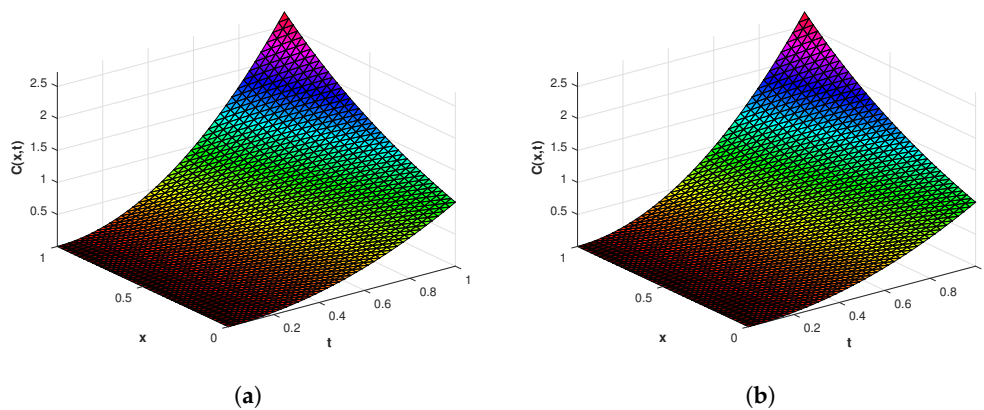


Figure 2. (a) Numerical and (b) exact space-time solutions of problem 1 at $t = 1$ with $M_h = 50$, $N = 40$, $n = 12$.

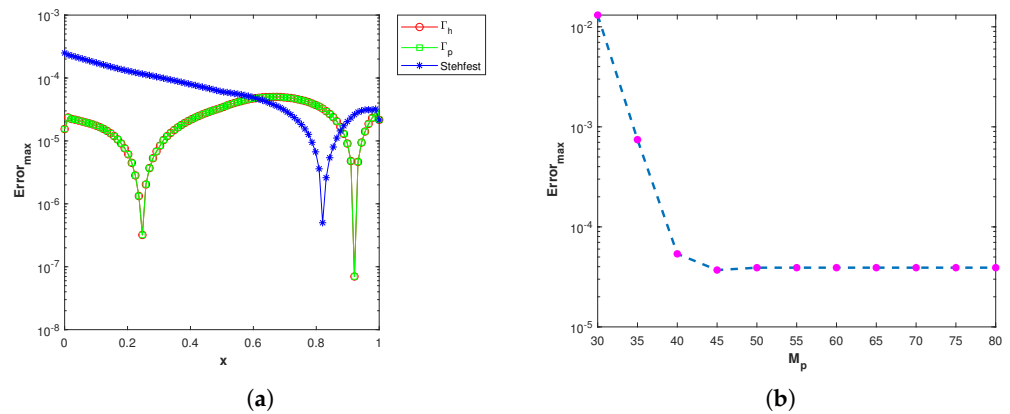


Figure 3. (a) Comparison of absolute errors of problem 1 using the three inversion methods with $M_h = 80$, $M_p = 60$, $M_s = 18$, $n = 15$, and $N = 90$. (b) $Error_{max}$ versus M_p with $P_e = 2$, $N = 90$, and $n = 6$.

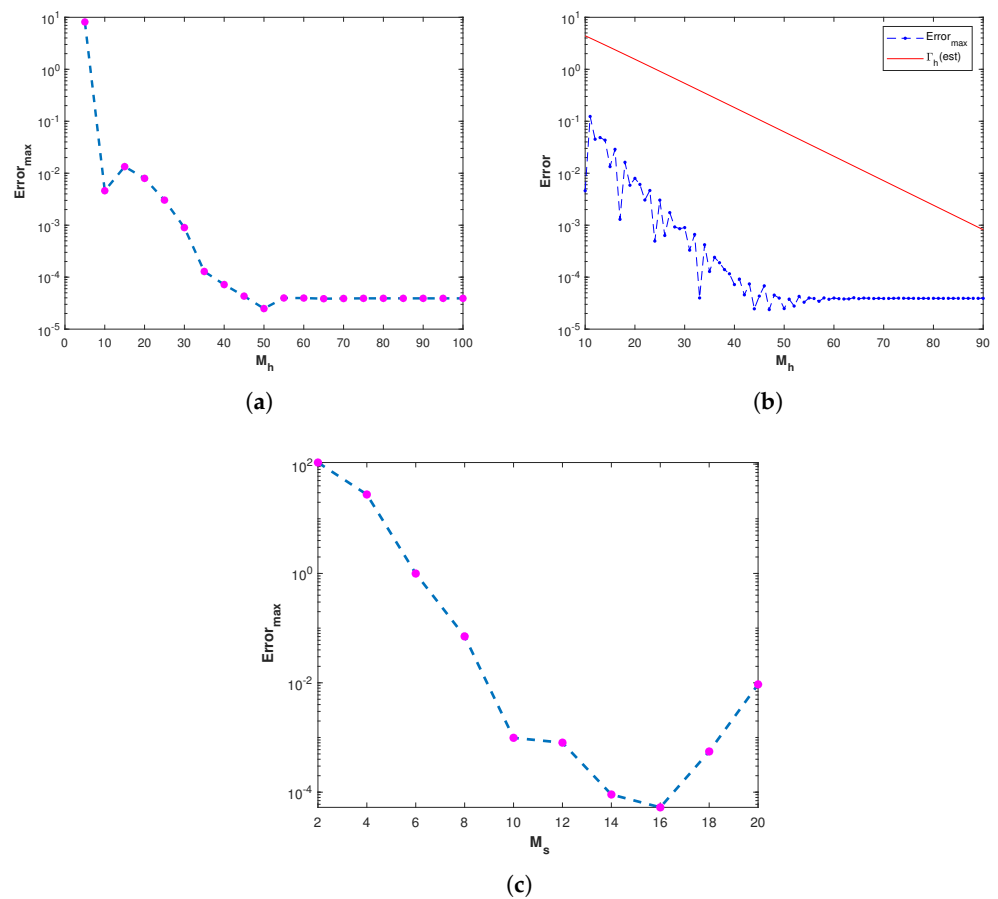


Figure 4. (a) $Error_{max}$ versus M_h with $n = 6$, and $N = 90$. (b) The absolute error versus error estimate for different values of M_h are shown, corresponding to Problem 1 with $N = 90$, $n = 6$. It is observed that the error agrees with the error estimate. (c) $Error_{max}$ versus M_s with $n = 6$, and $N = 90$, corresponding to problem 1.

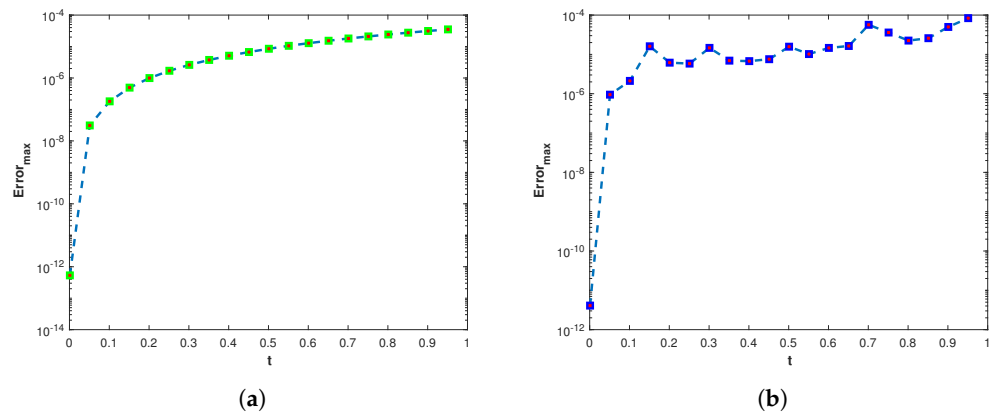


Figure 5. (a) $Error_{max}$ versus t with $n = 6$, $N = 90$ and $M_h = 90$. (b) $Error_{max}$ versus t with $n = 6$, $N = 90$ and $M_s = 16$, corresponding to problem 1.

Table 2. The $Error_{max}$ using the proposed scheme along the path Γ_h at $t = 1$.

M_h	N	n	$Error_{max}$	ϑ	C. Time (s)
30	20	10	5.49×10^{-4}	1.7	0.202415
40			1.67×10^{-4}	1.7	0.225109
50			1.45×10^{-4}	1.7	0.308366
60			1.53×10^{-4}	1.7	0.436627
70	30	12	7.50×10^{-5}	3.0	0.843163
	50		7.42×10^{-5}	5.1	2.020373
	80		9.54×10^{-5}	8.3	4.864506

Table 3. The $Error_{max}$ using the proposed scheme at $t = 1$.

M_s	N	n	$Error_{max}$	ϑ	CPU (s)
8	70	10	6.22×10^{-2}	6.4	0.346949
10			9.56×10^{-4}	6.4	0.334023
12			7.99×10^{-4}	6.4	0.281686
14			2.16×10^{-4}	6.4	0.288764
16	50	12	1.61×10^{-4}	5.1	0.225202
	60		1.81×10^{-4}	6.2	0.244202
	70		5.04×10^{-4}	7.3	0.336965
	80		3.47×10^{-4}	8.3	0.346112
	90		6.01×10^{-4}	9.4	0.418864

4.2. Problem 2

Here, a 1D equation of the form [10] is considered

$$\frac{\partial C(x, t)}{\partial t} + \bar{\theta} \frac{\partial C(x, t)}{\partial x} - \bar{\alpha} \frac{\partial^2 C(x, t)}{\partial x^2} = 0, \text{ where } 0 < x < 1, t > 0, \tag{40}$$

with the initial and boundary data

$$C(x, 0) = \bar{a} \exp(-\bar{c}x),$$

and

$$C(0, t) = \bar{a} \exp((\bar{b}t)),$$

$$C(1, t) = \bar{a} \exp((\bar{b}t - \bar{c})),$$

the analytical solution of the problem is

$$C(x, t) = \bar{a} \exp((\bar{b}t - \bar{c}\bar{\zeta})),$$

where

$$\bar{c} = \frac{-\bar{\beta} + \sqrt{\bar{\beta}^2 + 4\bar{\alpha}\bar{b}}}{2\bar{\alpha}}.$$

The ratio of the two constants $\bar{\theta}$ and $\bar{\alpha}$ is denoted by P_e , called the Péclet number, i.e., $P_e = |\frac{\bar{\theta}}{\bar{\alpha}}|$. For a large value of P_e , the convection term is dominant and, for a small value of P_e , the diffusion term is dominant [10]. The results using the proposed numerical scheme along the parabolic contour are shown in Table 4, and the results obtained using Stehfest's method are presented in Table 5. The same set of parameter values for the contours is used in this problem. Figure 6a,b depicts the numerical solutions of problem 2 computed using the parabolic contour at different time levels, with $P_e = 10$ and $P_e = 20$ and other parameter values $M_p = 70$, $N = 30$, $n = 10$. Figure 7a shows a comparison between the errors of problem 2 obtained using the three inversion techniques with various values of P_e . It is observed that the performance of the two contour integration methods is better than for Stehfest's method. Figure 7b shows the plot of error versus the nodes M_p along the parabolic contour. Figure 8a shows the plot of error versus M_h using the hyperbolic contour. A comparison between the observed and theoretical error is shown in Figure 8b; a good agreement between error and error estimate is observed. Figure 8c shows the plot of error versus M_s using Stehfest's method. In Figure 9a, the plot shows error versus t using the hyperbolic path. Figure 9b shows a plot of error versus t using Stehfest's method. From all the figures and tables, a very high accuracy and stability is evident. However, it is observed that the results along the hyperbolic path are more accurate than those for the results along the parabolic path and using Stehfest's method. The results demonstrate that the method is capable of approximating the solution of this type of equation efficiently.

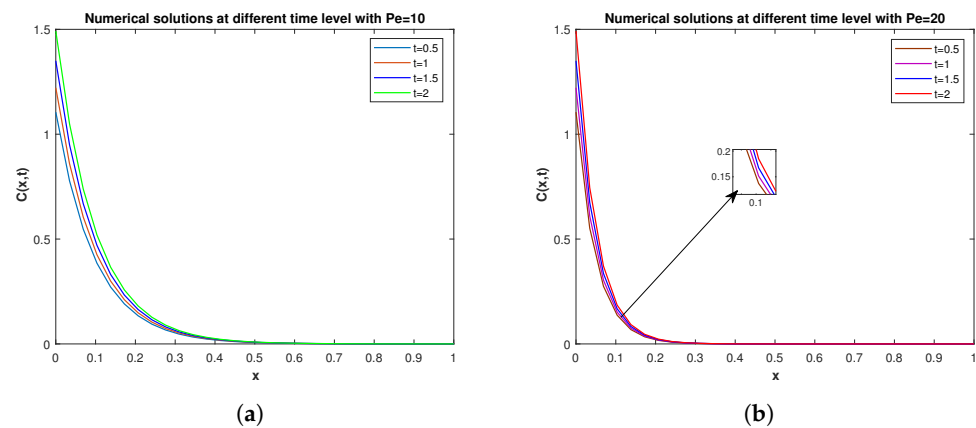


Figure 6. Numerical solutions of problem 2 at different time levels with $M_p = 70$, $N = 30$, $n = 10$.

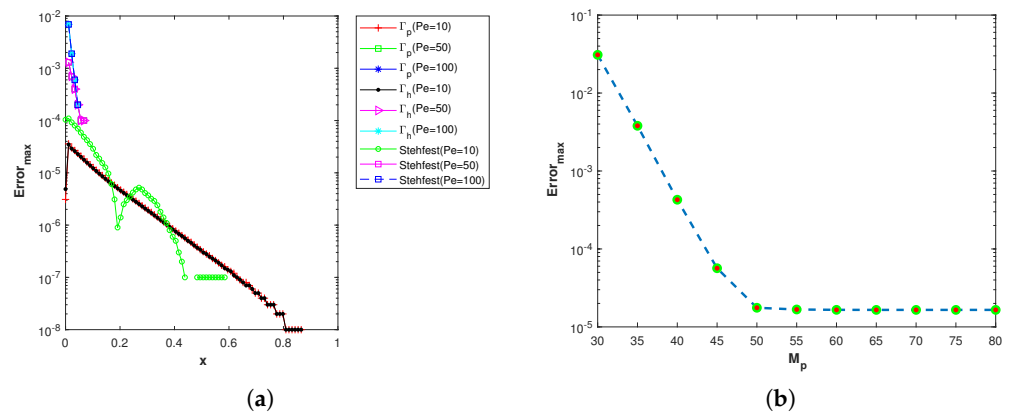


Figure 7. (a) Comparison of absolute errors of problem 2 for various values of Pe , with $M_h = 90$, $M_p = 70$, $M_s = 18$, $n = 15$, and $N = 90$. (b) $Error_{max}$ versus M_p with $Pe = 2$, $N = 90$, and $n = 6$.

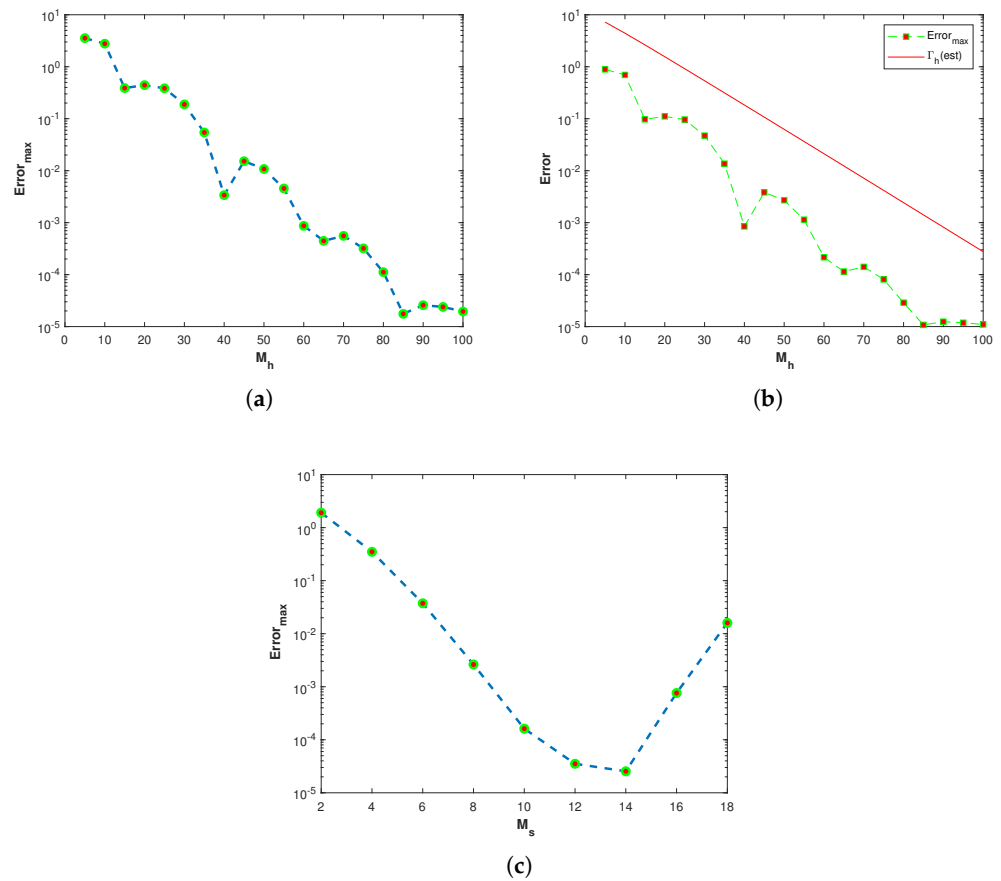


Figure 8. (a) $Error_{max}$ versus M_h with $Pe = 2$, $n = 6$, and $N = 90$. (b) Absolute error versus error estimate for different values of M_h are shown, corresponding to Problem 2 with $N = 100$, $n = 10$, $Pe = 10$. It is observed that the error agrees with the error estimate. (c) $Error_{max}$ versus M_s with $Pe = 2$, $n = 6$, and $N = 90$, corresponding to problem 2.

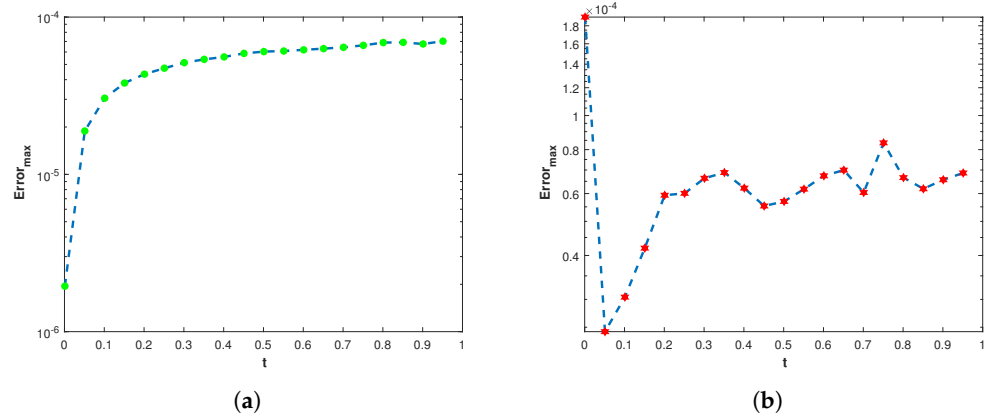


Figure 9. (a) $Error_{max}$ versus t with $P_e = 10$, $n = 6$, $N = 90$ and $M_h = 90$. (b) $Error_{max}$ versus t with $P_e = 10$, $n = 6$, $N = 90$ and $M_s = 16$, corresponding to problem 2.

Table 4. The $Error_{max}$ using the proposed scheme along the path Γ_p at $t = 1$.

M_p	N	n	$Error_{max}$	ϑ	C. Time (s)
40	90	10	1.12×10^{-4}	8.3	0.450731
50			3.54×10^{-5}	8.3	0.671841
60			3.53×10^{-5}	8.3	1.206206
70			3.53×10^{-5}	8.3	1.150397
80	60	10	7.61×10^{-5}	5.5	0.839738
	70		3.61×10^{-5}	6.4	0.997628
	100		1.07×10^{-5}	9.2	1.612720
[56]			4.41×10^{-3}		

Table 5. The $Error_{max}$ using the proposed scheme at $t = 1$.

M_s	N	n	$Error_{max}$	ϑ	CPU (s)
8	70	10	1.10×10^{-3}	6.4	0.317848
10			1.08×10^{-4}	6.4	0.332286
12			3.25×10^{-5}	6.4	0.323966
14			3.64×10^{-5}	6.4	0.331252
16	50	12	1.53×10^{-4}	5.1	0.268543
	60		1.34×10^{-4}	6.2	0.321235
	70		8.73×10^{-5}	7.3	0.365170
	80		7.73×10^{-5}	8.3	0.443169
	90		4.96×10^{-5}	9.4	0.533850
	100		2.72×10^{-5}	9.4	0.558243
[56]			4.41×10^{-3}		

4.3. Problem 3

Here, a 1D equation of the form [10] is considered

$$\frac{\partial C(x, t)}{\partial t} + \bar{\theta} \frac{\partial C(x, t)}{\partial x} - \bar{\alpha} \frac{\partial^2 C(x, t)}{\partial x^2} = 0, \quad 0 < x < 2, t > 0, \tag{41}$$

with the initial and boundary data

$$C(x, 0) = \sin(x),$$

and

$$C(0, t) = \exp(-\bar{\alpha}t) \sin(-\bar{\theta}t),$$

$$C(2, t) = \exp(-\bar{\alpha}t) \sin(2 - \bar{\theta}t),$$

the analytical solution is given as

$$C(x, t) = \exp(-\bar{\alpha}t) \sin(x - \bar{\theta}t).$$

The results obtained for various values of P_e along the hyperbolic contour are shown in Table 6. The results obtained for P_e using Stehfest's numerical inversion method are presented in Table 7. Here, we use the same set of parameters used for problem 1 and problem 2. Figure 10a–f depict the numerical and exact solutions of problem 3, computed using Stehfest's method, with $M_s = 16$, $N = 100$, $n = 12$, and $P_e = 2$, at different time levels. Figure 11a,b shows the exact and numerical space-time solutions of problem 3 using Stehfest's method at $t = 1$ with $M_s = 16$, $N = 30$, $n = 12$, and $P_e = 2$. Figure 12a shows a comparison between the errors of problem 3 obtained using the three inversion techniques with various values of P_e . It is observed that the performance of the two contour integration methods is better than for Stehfest's method. Figure 12b shows a plot of error versus the nodes M_p along the parabolic contour. Figure 13a shows a plot of error versus M_h using the hyperbolic contour. A comparison between the observed and theoretical error is shown in Figure 13b; a good agreement between the error and the error estimate is observed. Figure 13c shows the plot of error versus M_s using Stehfest's method. In Figure 14a, the plot shows error versus t using the hyperbolic path. Figure 14b depicts a plot of error versus t using Stehfest's method. From all the figures and tables, a very high accuracy and stability is evident. However, it is observed that the results along the hyperbolic path are more accurate than the results along the parabolic path and using Stehfest's method.

Table 6. $Error_{max}$ of the proposed method for problem 3 using Γ_h at $t = 1$ for various Péclet numbers.

M_h	N	n	$P_e = 100$	$P_e = 1000$	$P_e = 10,000$	$P_e = 20,000$
30	40	8	9.83×10^{-2}	9.83×10^{-2}	9.83×10^{-2}	9.83×10^{-2}
40			2.20×10^{-3}	2.20×10^{-3}	2.20×10^{-3}	2.20×10^{-3}
50			5.70×10^{-3}	5.70×10^{-3}	5.70×10^{-3}	5.70×10^{-3}
60			4.37×10^{-4}	4.37×10^{-4}	4.37×10^{-4}	4.38×10^{-4}
70			2.96×10^{-4}	2.96×10^{-4}	2.96×10^{-4}	2.96×10^{-4}
90			1.79×10^{-5}	1.76×10^{-5}	1.79×10^{-5}	1.80×10^{-5}
100			1.25×10^{-5}	1.24×10^{-5}	1.29×10^{-5}	1.29×10^{-5}
90	20	10	1.67×10^{-4}	1.95×10^{-4}	2.02×10^{-4}	2.02×10^{-4}
	30		2.59×10^{-5}	4.09×10^{-5}	5.12×10^{-5}	5.20×10^{-5}
	40		4.38×10^{-5}	4.62×10^{-5}	5.16×10^{-5}	5.20×10^{-5}
	50		5.93×10^{-5}	6.40×10^{-5}	6.41×10^{-5}	6.40×10^{-5}
[10]				2.87×10^{-7}	2.68×10^{-4}	7.40×10^{-4}

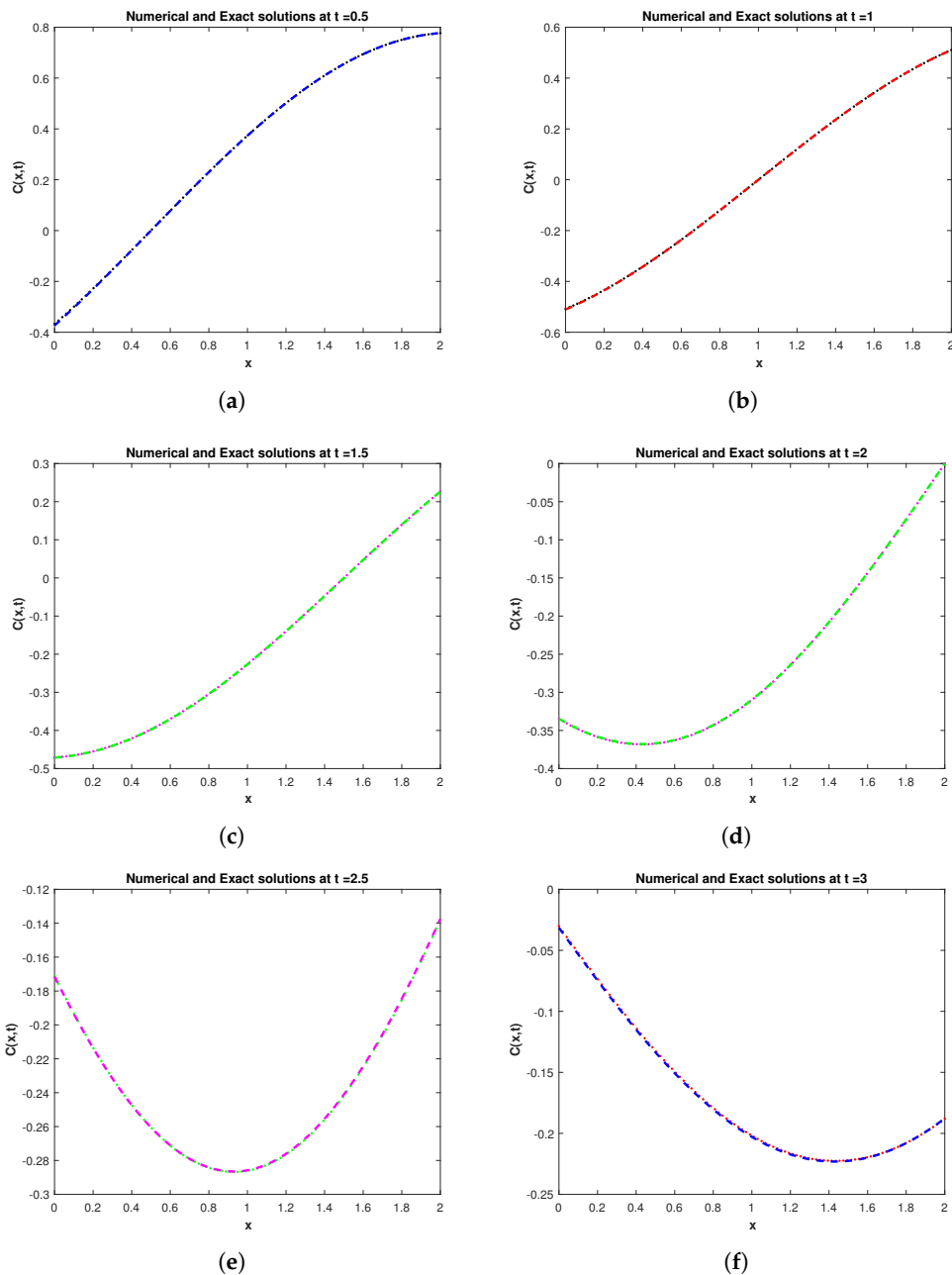


Figure 10. Numerical and exact solutions of problem 3 at different time levels with $M_s = 16$, $N = 100$, $n = 12$, and $Pe = 2$.

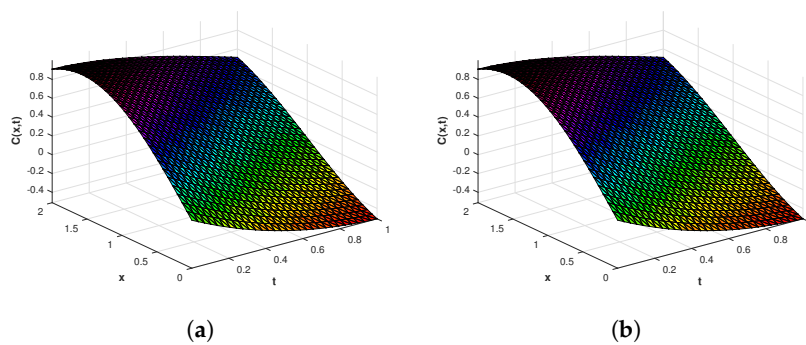


Figure 11. (a) Numerical and (b) exact space-time solutions of problem 3 at $t = 1$ with $M_s = 16$, $N = 30$, $n = 12$, and $Pe = 2$.

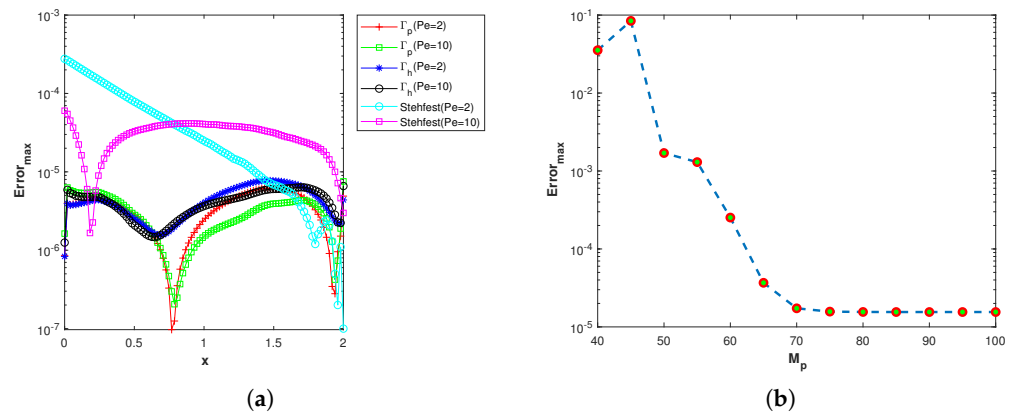


Figure 12. (a) Comparison of absolute errors of problem 3 for various values of Pe , with $M_h = 90$, $M_p = 70$, $M_s = 16$, $n = 15$, and $N = 100$. (b) Error versus M_p with $Pe = 2$, $N = 90$, and $n = 12$.

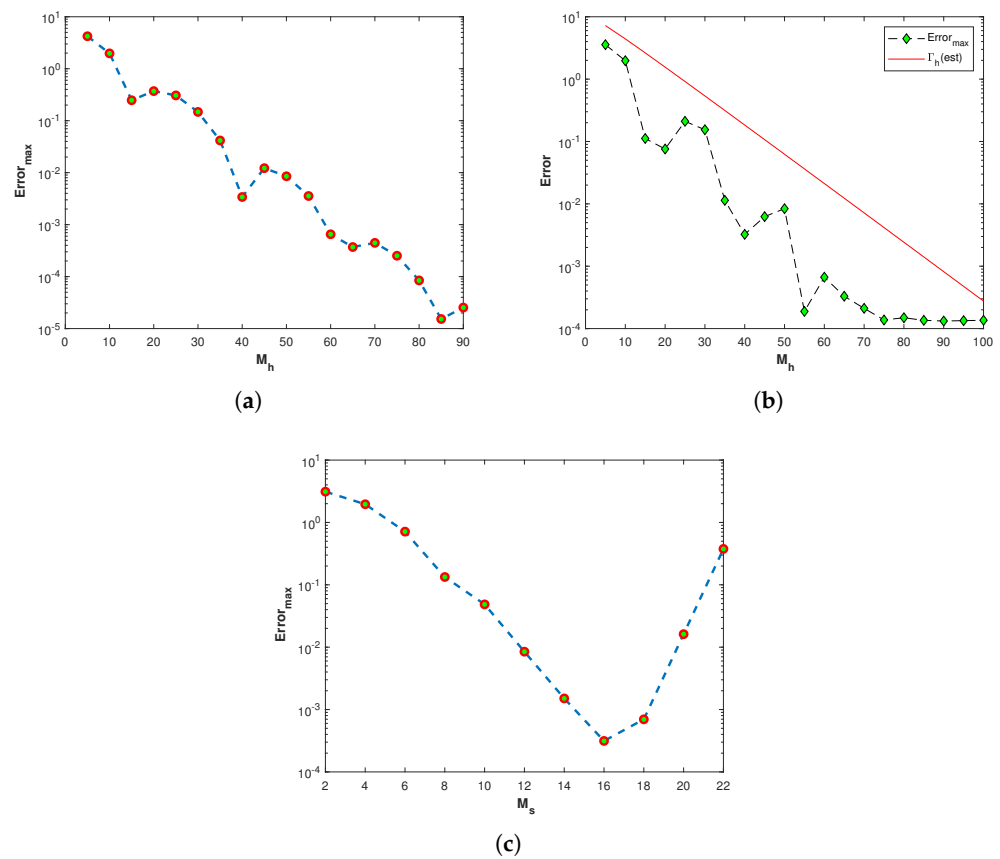


Figure 13. (a) Error versus M_h with $Pe = 2$, $n = 12$, and $N = 90$. (b) The absolute error versus error estimate for different values of M_h are shown, corresponding to Problem 3 with $N = 100$, $n = 5$, $Pe = 10$. It is observed that the error agrees with the error estimate. (c) Error versus M_s with $Pe = 2$, $n = 12$, and $N = 90$, corresponding to problem 3.

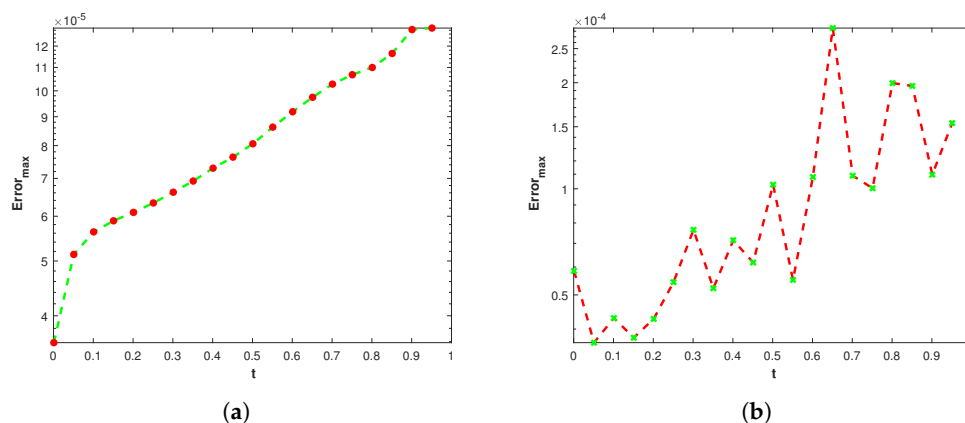


Figure 14. (a) Error versus t with $Pe = 10$, $n = 5$, $N = 100$ and $M_h = 90$. (b) Error versus t with $Pe = 10$, $n = 5$, $N = 100$ and $M_s = 18$, corresponding to problem 3.

Table 7. $Error_{max}$ using the proposed scheme at $t = 1$ and $Pe = 100$.

M_s	N	n	$Error_{max}$	ϑ	CPU (s)
8	70	10	1.60×10^{-1}	6.4	0.334523
10			6.29×10^{-2}	6.4	0.426634
12			8.80×10^{-3}	6.4	0.347591
14			2.40×10^{-3}	6.4	0.479109
16	50	12	2.20×10^{-4}	5.1	0.361524
	60		2.61×10^{-4}	6.2	0.322293
	70		4.20×10^{-4}	7.3	0.376275
	80		3.90×10^{-4}	8.3	0.447432
	90		3.44×10^{-4}	9.4	0.718522
	100		4.98×10^{-4}	9.4	0.672432

5. Conclusions

In this paper, a numerical method based on the Laplace transform and local radial basis functions was developed for numerical simulation diffusion equations. The method employed the Laplace transform and inverse Laplace transform methods to cope with the time derivative. This method first uses the Laplace transform to convert the given problem into an inhomogeneous problem in the Laplace domain. Then it utilizes the local radial basis functions for discretization of the spatial derivatives. Finally, it uses the numerical inverse Laplace transform methods to convert the solution obtained in the Laplace domain into a time domain. The present method is almost exact in time, having no stability restrictions, which is a common issue in finite-difference time-stepping methods. In these methods, for optimal accuracy, we need a very small time step. The local radial basis functions makes this method very interesting and easy to implement. An error analysis and the results of computation demonstrate that the present method is an efficient and competitive method for the solution of diffusion problems, with the benefits that it is a simple, mesh-less, easy to implement and highly accurate numerical method.

Author Contributions: Conceptualization, K. and F.A.S.; methodology, K.; software, K. and F.M.A.; validation, K., I.M. and H.A.; formal analysis, F.M.A.; investigation, H.A.; resources, W.H.F.A.; data curation, I.M. and F.M.A.; writing—K. and F.A.S.; writing—K., H.A. and F.A.S.; visualization, W.H.F.A.; supervision, I.M. and K.; project administration, W.H.F.A.; funding acquisition, W.H.F.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Acknowledgments: The authors Wael Hosny Fouad Aly and Ibrahim Mahariq would like to thank the College of Engineering and Technology, American University of the Middle East, Kuwait for support. The author Fahad M. Alotaibi would like to thank the Department of Information Systems, Faculty of Computing and Information Technology (FCIT), King Abdulaziz University, Jeddah 34025, Saudi Arabia.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dehghan, M. Weighted finite difference techniques for the one-dimensional advection-diffusion equation. *Appl. Math. Comput.* **2004**, *147*, 307–319. [[CrossRef](#)]
2. Gazizov, R.K.; Kasatkin, A.A.; Lukashchuk, S.Y. Symmetry properties of fractional diffusion equations. *Phys. Scr.* **2009**, *2009*, 014016. [[CrossRef](#)]
3. Sun, Y.; Jayaraman, A.S.; Chirikjian, G.S. Lie group solutions of advection-diffusion equations. *Phys. Fluids* **2021**, *33*, 046604. [[CrossRef](#)]
4. Mittal, R.C.; Jain, R.K. Redefined cubic B-spline collocation method for solving convection diffusion equations. *Appl. Math. Model.* **2012**, *36*, 5555–5573. [[CrossRef](#)]
5. Dehghan, M. Numerical solution of the three-dimensional advection-diffusion equation. *Appl. Math. Comput.* **2004**, *150*, 5–19. [[CrossRef](#)]
6. Isenberg, J.; Gutfinger, C. Heat transfer to a draining film. *Int. J. Heat Transf.* **1972**, *16*, 505–512. [[CrossRef](#)]
7. Fattah, Q.N.; Hoopes, J.A. Dispersion in anisotropic homogeneous porous media. *J. Hydraul. Eng.* **1985**, *111*, 810–827. [[CrossRef](#)]
8. Zlatev, Z.; Berkowicz, R.; Prahm, L.P. Implementation of a variable stepsize variable formula method in the time-integration part of a code for treatment of long-range transport of air pollutants. *J. Comput. Phys.* **1984**, *55*, 278–301. [[CrossRef](#)]
9. Parlange, J.Y. Water transport in soils. *Annu. Rev. Fluid Mech.* **1980**, *12*, 77–102. [[CrossRef](#)]
10. Mohebbi, A.; Dehghan, M. High-order compact solution of the one-dimensional heat and advection-diffusion equations. *Appl. Math. Model.* **2010**, *34*, 3071–3084. [[CrossRef](#)]
11. Nazir, T.; Abbas, M.; Ismail, A.I.M.; Majid, A.A.; Rashid, A. The numerical solution of advection-diffusion problems using new cubic trigonometric B-splines approach. *Appl. Math. Model.* **2016**, *40*, 4586–4611. [[CrossRef](#)]
12. Ismail, H.N.; Elbarbary, E.M.; Salem, G.S. Restrictive Taylor’s approximation for solving convection-diffusion equation. *Appl. Math. Comput.* **2004**, *147*, 355–363. [[CrossRef](#)]
13. Kumar, A.; Jaiswal, D.K.; Kumar, N. Analytical solutions of one-dimensional advection-diffusion equation with variable coefficients in a finite domain. *J. Earth Syst. Sci.* **2009**, *118*, 539–549. [[CrossRef](#)]
14. Kumar, A.; Jaiswal, D.K.; Kumar, N. Analytical solutions to one-dimensional advection-diffusion equation with variable coefficients in semi-infinite media. *J. Hydrol.* **2010**, *380*, 330–337. [[CrossRef](#)]
15. Zoppou, C.; Knight, J.H. Analytical solution of a spatially variable coefficient advection-diffusion equation in up to three dimensions. *Appl. Math. Model.* **1999**, *23*, 667–685. [[CrossRef](#)]
16. Jha, B.K.; Adlakha, N.; Mehta, M.N. Analytic solution of two dimensional advection diffusion equation arising in cytosolic calcium concentration distribution. *Int. Math. Forum* **2012**, *7*, 135–144.
17. Guerrero, J.P.; Pimentel, L.C.G.; Skaggs, T.H.; Van Genuchten, M.T. Analytical solution of the advection-diffusion transport equation using a change-of-variable and integral transform technique. *Int. J. Heat Mass Transf.* **2009**, *52*, 3297–3304. [[CrossRef](#)]
18. Guerrero, J.P.; Pimentel, L.C.G.; Skaggs, T.H. Analytical solution for the advection-dispersion transport equation in layered media. *Int. J. Heat Mass Transf.* **2013**, *56*, 274–282. [[CrossRef](#)]
19. Jaiswal, D.K.; Atul, K.; Yadav, R.R. Analytical solution to the one-dimensional advection-diffusion equation with temporally dependent coefficients. *J. Water Resour. Prot.* **2011**, *2011*, 3781.
20. Karahan, H. Unconditional stable explicit finite difference technique for the advection-diffusion equation using spreadsheets. *Adv. Eng. Softw.* **2007**, *38*, 80–86. [[CrossRef](#)]
21. Karahan, H. Implicit finite difference techniques for the advection-diffusion equation using spreadsheets. *Adv. Eng. Softw.* **2006**, *37*, 601–608. [[CrossRef](#)]
22. Nguyen, H.; Reynen, J. A space-time least-square finite element scheme for advection-diffusion equations. *Comput. Methods Appl. Mech. Eng.* **1984**, *42*, 331–342. [[CrossRef](#)]
23. Cunha, C.L.N.; Carrer, J.A.M.; Oliveira, M.F.; Costa, V.L. A study concerning the solution of advection-diffusion problems by the Boundary Element Method. *Eng. Anal. Bound. Elem.* **2016**, *65*, 79–94. [[CrossRef](#)]
24. Brebbia, C.; Skerget, P. Time dependent diffusion convection problems using boundary elements. In *Numerical Methods in Laminar and Turbulent Flow, Proceedings of the Third International Conference, Seattle, WA, USA, 8–11 August 1983*; Pineridge Press: Swansea, UK, 1983; pp. 715–739.
25. Boztosun, I.; Charafi, A. An analysis of the linear advection-diffusion equation using mesh-free and mesh-dependent methods. *Eng. Anal. Bound. Elem.* **2002**, *26*, 889–895. [[CrossRef](#)]
26. Chen, J.S.; Hillman, M.; Rüter, M. An arbitrary order variationally consistent integration for Galerkin meshfree methods. *Int. J. Numer. Methods Eng.* **2013**, *95*, 387–418. [[CrossRef](#)]

27. Chen, W. Symmetric boundary knot method. *Eng. Anal. Bound. Elem.* **2002**, *26*, 489–494. [[CrossRef](#)]
28. Fu, Z.; Chen, W.; Wen, P.; Zhang, C. Singular boundary method for wave propagation analysis in periodic structures. *J. Sound Vib.* **2018**, *425*, 170–188. [[CrossRef](#)]
29. Gu, Y.; Liu, G.R. A local point interpolation method for static and dynamic analysis of thin beams. *Comput. Methods Appl. Mech. Eng.* **2001**, *190*, 5515–5528. [[CrossRef](#)]
30. 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. *Comput. Math. Appl.* **1990**, *19*, 147–161. [[CrossRef](#)]
31. Hon, Y.C.; Schaback, R.; Zhou, X. An adaptive greedy algorithm for solving large RBF collocation problems. *Numer. Algorithms* **2003**, *32*, 13–25. [[CrossRef](#)]
32. Šarler, B. From global to local radial basis function collocation method for transport phenomena. In *Advances in Meshfree Techniques*; Springer: Dordrecht, The Netherlands, 2007; pp. 257–282.
33. Fasshauer, G.E.; McCourt, M.J. Stable evaluation of Gaussian radial basis function interpolants. *SIAM J. Sci. Comput.* **2012**, *34*, A737–A762. [[CrossRef](#)]
34. Kansa, E.J. Multiquadrics-A scattered data approximation scheme with applications to computational fluid-dynamics-I surface approximations and partial derivative estimates. *Comput. Math. Appl.* **1990**, *19*, 127–145. [[CrossRef](#)]
35. Waters, J.; Pepper, D.W. Global versus localized RBF meshless methods for solving incompressible fluid flow with heat transfer. *Numerical Heat Transfer. Part B Fundam.* **2015**, *68*, 185–203. [[CrossRef](#)]
36. Fasshauer, G.E.; Zhang, J.G. On choosing “optimal” shape parameters for RBF approximation. *Numer. Algorithms* **2007**, *45*, 345–368. [[CrossRef](#)]
37. Shu, C.; Ding, H.; Yeo, K.S. Local radial basis function-based differential quadrature method and its application to solve two-dimensional incompressible Navier-Stokes equations. *Comput. Methods Appl. Mech. Eng.* **2003**, *192*, 941–954. [[CrossRef](#)]
38. Shu, C.; Ding, H.; Chen, H.Q.; Wang, T.G. An upwind local RBF-DQ method for simulation of inviscid compressible flows. *Comput. Methods Appl. Mech. Eng.* **2005**, *194*, 2001–2017. [[CrossRef](#)]
39. Ding, H.; Shu, C.; Yeo, K.S.; Xu, D. Numerical computation of three-dimensional incompressible viscous flows in the primitive variable form by local multiquadric differential quadrature method. *Comput. Methods Appl. Mech. Eng.* **2006**, *195*, 516–533. [[CrossRef](#)]
40. Wu, W.X.; Shu, C.; Wang, C.M. Vibration analysis of arbitrarily shaped membranes using local radial basis function-based differential quadrature method. *J. Sound Vib.* **2007**, *306*, 252–270. [[CrossRef](#)]
41. Lee, C.K.; Liu, X.; Fan, S.C. Local multiquadric approximation for solving boundary value problems. *Comput. Mech.* **2003**, *30*, 396–409. [[CrossRef](#)]
42. Šarler, B.; Vertnik, R. Meshfree explicit local radial basis function collocation method for diffusion problems. *Comput. Math. Appl.* **2006**, *51*, 1269–1282. [[CrossRef](#)]
43. Fu, Z.J.; Chen, W.; Yang, H.T. Boundary particle method for Laplace transformed time fractional diffusion equations. *J. Comput. Phys.* **2013**, *235*, 52–66. [[CrossRef](#)]
44. Moridis, G.J.; Kansa, E. The Laplace Transform Multiquadric Method: A Highly Accurate Scheme for the Numerical Solution of Partial Differential Equations. *Int. J. Sci. Comput. Model.* **1993**, submitted.
45. Le Gia, Q.T.; McLean, W. Solving the heat equation on the unit sphere via Laplace transforms and radial basis functions. *Adv. Comput. Math.* **2014**, *40*, 353–375. [[CrossRef](#)]
46. Jacobs, B.A. High-order compact finite difference and Laplace transform method for the solution of time-fractional heat equations with dirichlet and neumann boundary conditions. *Numer. Methods Partial Differ. Equ.* **2016**, *32*, 1184–1199. [[CrossRef](#)]
47. Weideman, J.A.C.; Trefethen, L. Parabolic and hyperbolic contours for computing the Bromwich integral. *Math. Comput.* **2007**, *76*, 1341–1356. [[CrossRef](#)]
48. McLean, W.; Thomée, V. Numerical solution via Laplace transforms of a fractional order evolution equation. *J. Integral Equ. Appl.* **2010**, *22*, 57–94. [[CrossRef](#)]
49. López-Fernández, M.; Palencia, C. On the numerical inversion of the Laplace transform of certain holomorphic mappings. *Appl. Numer. Math.* **2004**, *51*, 289–303. [[CrossRef](#)]
50. Kuznetsov, A. On the Convergence of the Gaver–Stehfest Algorithm. *SIAM J. Numer. Anal.* **2013**, *51*, 2984–2998. [[CrossRef](#)]
51. Davies, B.; Martin, B. Numerical inversion of the Laplace transform: A survey and comparison of methods. *J. Comput. Phys.* **1979**, *33*, 1–32. [[CrossRef](#)]
52. Sarra, S.A.; Kansa, E.J. Multiquadric radial basis function approximation methods for the numerical solution of partial differential equations. *Adv. Comput. Mech.* **2009**, *2*, 220.
53. Abate, J.; Valkó, P.P. Multi-precision Laplace transform inversion. *Int. J. Numer. Methods Eng.* **2004**, *60*, 979–993. [[CrossRef](#)]
54. Abate, J.; Whitt, W. A unified framework for numerically inverting Laplace transforms. *INFORMS J. Comput.* **2006**, *18*, 408–421. [[CrossRef](#)]
55. Schaback, R. Error estimates and condition numbers for radial basis function interpolation. *Adv. Comput. Math.* **1995**, *3*, 251–264. [[CrossRef](#)]
56. Zerroukat, M.; Djidjeli, K.; Charafi, A. Explicit and implicit meshless methods for linear advection-diffusion-type partial differential equations. *Int. J. Numer. Methods Eng.* **2000**, *48*, 19–35. [[CrossRef](#)]