Asia Pac. J. Math. 2025 **12**:44

QUARTER-SWEEP FINITE DIFFERENCE APPROXIMATION WITH THOMAS ALGORITHM FOR SOLVING NONLINEAR ADVECTION-DIFFUSION EQUATION

JACKEL VUI LUNG CHEW[1,*], JUMAT SULAIMAN[2], ANDANG SUNARTO[3]

[1]Faculty of Computing and Informatics, Universiti Malaysia Sabah Labuan International Campus, 87000 Sg. Pagar, Labuan, Malaysia

[2]Faculty of Science and Natural Resources, Universiti Malaysia Sabah, 88400 Kota Kinabalu, Sabah, Malaysia

[3]Tadris Matematika, Universitas Islam Negeri (UIN) Fatmawati Sukarno, 38211, Bengkulu, Indonesia

[*]Corresponding author: jackelchew93@ums.edu.my

Received Mar. 22, 2025

Abstract. Solving nonlinear advection-diffusion equations efficiently is one of the challenging tasks in computational mathematics. These equations are commonly used to model transport phenomena such as fluid flow, heat transfer, and pollutant dispersion. Finite difference methods are widely applied for solving these equations. However, their application in multi-dimensional mathematical problems involves high computational complexity. To address this issue, this paper investigates the computational efficiency of the quarter-sweep finite difference approximation combined with the Thomas algorithm. The proposed numerical method utilises the quarter-sweep strategy to significantly reduce the number of computations required per iteration while maintaining numerical accuracy. Through extensive numerical experiments, the computational performance of the proposed method is carefully assessed by comparing it with the standard implicit finite difference method. The experimental results show that the proposed numerical method achieves higher computational efficiency while maintaining comparable numerical accuracy when it is compared to the standard implicit finite difference method. The reduction in computational load makes the proposed method particularly beneficial for large-scale simulations. The contribution of this research is the integration of the quarter-sweep strategy with the Thomas algorithm which offers an alternative numerical solution strategy for solving nonlinear advection-diffusion equations. This research has potential implications in fields such as fluid dynamics, environmental modelling, and engineering applications. Despite its advantages, this research is limited to one-dimensional problems. Future work will focus on extending the numerical solution strategy to higher-dimensional problems. The findings of this research contribute to the ongoing efforts in developing efficient and scalable numerical methods for solving nonlinear partial differential equations.

2020 Mathematics Subject Classification. 65M06; 65M22.

nonlinear advection-diffusion equation; quarter-sweep; finite difference method; Thomas algorithm; numerical solution; computational efficiency.

DOI: 10.28924/APJM/12-44

1. Introduction

The nonlinear advection-diffusion equation (NADE) is a commonly used parabolic partial differential equation to model the transport of substances such as solutes and pollutants in fluid flow. It combines advection process, which describes the movement of particles due to fluid flow, and diffusion process, which describes the random spreading of particles from regions of higher to lower concentration. This equation has broad applications in fluid dynamics, heat transfer, environmental science, and pollutant dispersion modelling [1–5].

Solving the NADE is a challenging task for several reasons. One of the main challenges is the nonlinear nature of the equation. This nature can develop shock solutions that complicate numerical approaches [6]. Additionally, many existing methods that solve the NADE can maintain the theoretical findings but require significant computational resources [7]. Guaranteeing numerical stability is also critical because inappropriate time steps or parameter values can lead to inaccurate or divergent solutions [3,5,6].

Based on our review on the finite difference methods for NADE. We found that several studies used high-order compact finite difference methods to improve the accuracy of NADE solutions. For example, Mohebbi and Dehghan [8] proposed a fourth-order compact finite difference method for solving the one-dimensional advection–diffusion equations. Their method achieved fourth-order accuracy in both time and space while solving the problems. Then, Gurarslan [9] developed a sixth-order compact finite difference method and demonstrated higher accuracy in solving advection-diffusion equations. To reduce computational times, Gurarslan [9] used the combination of the method of lines and the fourth-order Runge-Kutta method. On the other hand, the implicit finite difference method such as Crank-Nicolson method is commonly combined with spline-based techniques to handle the temporal and spatial components of advection-diffusion equations. The Crank-Nicolson method can guarantee unconditional stability, while B-spline collocation methods can achieve greater accuracy in approximating spatial derivatives [7,10].

Although finite difference methods can produce highly accurate solutions, they are computationally intensive, and this problem becomes more severe in multi-dimensional applications. Motivated by literature that often focuses on improving the accuracy of NADE solutions, but reports less on computational efficiency, this paper proposes an efficient numerical method for solving the NADE. The proposed method combines the quarter-sweep computational strategy with the Thomas algorithm. Previous research has highlighted the efficiency of the quarter-sweep strategy in reducing computational load without compromising solution accuracy [11–15].

Therefore, we integrate the quarter-sweep strategy, for the first time, with the Thomas algorithm to efficiently solve the large systems of equations that arise from the implicit finite difference discretization of the NADE. This paper is organized as follows. Section 2 describes the mathematical problem that we aim to solve. Section 3 discusses the formulation of the proposed numerical method to solve the mathematical problem. Section 4 provides the convergence analysis of the formulated approximation equation. Section 5 presents the numerical experiments set up together with results and discussions. Finally, the conclusion and future work are stated in Section 6.

2. Mathematical Problem

The general form of an advection diffusion equation can be written as [16]:

$$\frac{\partial v}{\partial t} = \alpha \frac{\partial^2 v}{\partial x^2} - \beta \frac{\partial v}{\partial x} + f, \quad x \in (0, L), \quad t \in (0, T), \tag{1}$$

where v stands for the vorticity, α is the advection velocity or phase speed function, β is the viscosity or diffusion function, and f is the known source function with variables x and t. Based on Equation 1, α and β can be assigned positive numbers to get linear advection-diffusion equations. When β is a nonlinear form, $\beta = \beta(v)$, Equation 1 becomes the main equation that we aim to solve, which is the nonlinear advection-diffusion equation (NADE). Additionally, the boundary and initial conditions must be specified for well-posedness. Thus, this paper focuses on the following boundary and initial conditions, $v(0, t) = v_0$, $v(L, t) = v_L$, and $v(x, 0) = v^0$ for the domain $x \in (0, L)$.

3. Proposed Numerical Method

The quarter-sweep strategy is a computational strategy that updates only a subset of the grid points in each iteration, which reduces the computational complexity compared to full-sweep and half-sweep methods [11–15]. For the NADE form as in Equation 1, the solution domain is discretized using finite differences, and the equation is approximated at these grid points. As a result, each grid point can be approximated using $v(ph, nk) = v_p^n$.

To derive the quarter-sweep finite difference operators, let us consider the Taylor expansions as follows.

$$v_p^{n-1} = v_p^n - k \frac{\partial}{\partial t} v_p^n + \frac{k^2}{2} \frac{\partial^2}{\partial t^2} v_p^n + \cdots, \tag{2}$$

$$v_{p+4}^n = v_p^n + 4h \frac{\partial}{\partial x} v_p^n + 16h^2 \frac{\partial^2}{\partial x^2} v_p^n + \cdots, \tag{3}$$

and

$$v_{p-4}^n = v_p^n - 4h \frac{\partial}{\partial x} v_p^n + 16h^2 \frac{\partial^2}{\partial x^2} v_p^n + \cdots. \tag{4}$$

Combining and rearranging terms based on Equations 2, 3, and 4, the quarter-sweep finite difference operators can be derived into

$$\frac{\partial v}{\partial t} = \frac{v_p^n - v_p^{n-1}}{k}, \tag{5}$$

$$\frac{\partial v}{\partial x} = \frac{v_{p+4}^n - v_{p-4}^n}{8h}, \tag{6}$$

and

$$\frac{\partial^2 v}{\partial x^2} = \frac{v_{p+4}^n - 2v_p^n + v_{p-4}^n}{16h^2}, \tag{7}$$

where $p = 4, 8, \ldots, M-4$, $n = 1, 2, \ldots, N$, $h = L/M$, and $k = T/N$, such that $M, N \in \mathbb{Z}^+$. Substituting Equations 5, 6, and 7 into Equation 1 yields a quarter-sweep finite difference approximation to NADE, given by,

$$G(v_p^n) = v_p^n - c_1\left(v_{p+4}^n - 2v_p^n + v_{p-4}^n\right)$$

$$+c_2\left(v_{p+4}^n - v_{p-4}^n\right) - v_p^{n-1} + f, \tag{8}$$

where $c_1 = \frac{\alpha k}{16h^2}$ and $c_2 = \frac{\beta k}{8h}$.

The computation of NADE solutions based on Equation 8 can be facilitated using the computational grid shown in Figure 1.

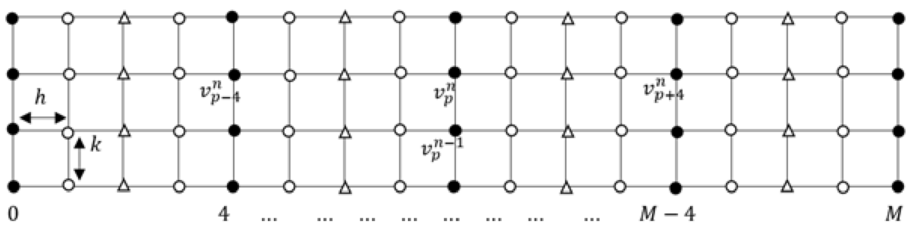

FIGURE 1. Computational grid for a quarter-sweep finite difference approximation.

As illustrated in Figure 1, the approximate solutions of Equation 1 are obtained iteratively by sequentially computing three distinct groups of points: black dots, white dots, and triangles. The computation begins with iterations of the black dots. Once the values at the black dots are determined, the values at the triangle points and white dots are obtained using linear interpolations. To compute the group of black dots, as shown in Figure 1, a system of nonlinear equations corresponding to Equation 8, which applied to the domain $x \in (0, L)$, $t \in (0, T)$, can be derived as,

$$F(\Omega) = 0, \tag{9}$$

where

$$F(.) = \left(G_4^n, \ldots, G_{M-4}^n\right)^T,$$

and

$$\Omega = \left(v_4^n, \ldots, v_{M-4}^n\right).$$

Then, using the second-order Newton method, a system of linear equations can be derived from Equation 9 as,

$$J_F(\Omega) \cdot \Delta\Omega = -F(\Omega), \tag{10}$$

where

$$J_F(\Omega) = \begin{bmatrix} \frac{\partial G_4^n}{\partial v_4^n} & \frac{\partial G_4^n}{\partial v_8^n} & \cdots & \frac{\partial G_4^n}{\partial v_{M-4}^n} \\ \frac{\partial G_8^n}{\partial v_4^n} & \frac{\partial G_8^n}{\partial v_8^n} & \cdots & \frac{\partial G_8^n}{\partial v_{M-4}^n} \\ \vdots & \vdots & & \vdots \\ \frac{\partial G_{M-4}^n}{\partial v_4^n} & \frac{\partial G_{M-4}^n}{\partial v_8^n} & \cdots & \frac{\partial G_{M-4}^n}{\partial v_{M-4}^n} \end{bmatrix},$$

and

$$\Delta\Omega = \Omega^{(i)} - \Omega^{(i-1)}, \quad i = 1, 2, \ldots.$$

Since the computation of Equation 10 involves many grid points, the computational complexity increases and requires more resources. This issue becomes significant when solving higher nonlinearity and multi-dimensional problems. Therefore, to reduce computational complexity, we propose the Thomas algorithm for the iteration process of the solution. Thomas algorithm for solving Equation 10 can be developed as follows. Suppose Equation 10 is tridiagonal with equations of the form

$$a_p\,\Delta\Omega_{p-4} + b_p\,\Delta\Omega_p + d_p\,\Delta\Omega_{p+4} = G_p, \tag{11}$$

for $p = 4, 8, \ldots, M-4$, where

$$a_p = \frac{\partial G}{\partial v_{p-4}^n} = -c_1 - c_2 + c_2'(v_{p+4}^n - v_{p-4}^n),$$

$$b_p = \frac{\partial G}{\partial v_p^n} = 1 + 2c_1,$$

and

$$d_p = \frac{\partial G}{\partial v_{p+4}^n} = -c_1 + c_2 + c_2'(v_{p+4}^n - v_{p-4}^n).$$

In our algorithm, the forward elimination phase involves

$$b_p' = \begin{cases} b_p, & p = 4, \\ b_p - \frac{a_p c_{p-4}}{b_{p-4}'}, & p = 8, \ldots, M-4, \end{cases} \tag{12}$$

and

$$G_p' = \begin{cases} \frac{G_p}{b_p'}, & p = 4, \\ G_p - \frac{a_p G_{p-4}'}{b_p'}, & p = 8, \ldots, M-4. \end{cases} \tag{13}$$

After the forward elimination, we perform back substitution to solve for all $\Delta\Omega$ using

$$\Delta\Omega' = \begin{cases} G_p', & p = M-4, \\ G_p' - \frac{c_p \Delta\Omega_{p+4}'}{b_p'}, & p = M-8, \ldots, 8, 4. \end{cases} \tag{14}$$

After each iteration, the solution vector is updated using

$$\Omega^{(i)} = \Omega^{(i-1)} + \Delta\Omega, \quad i = 1, 2, \ldots. \tag{15}$$

This process is repeated until the convergence criteria are satisfied, which is

$$\|\Omega^{(i)} - \Omega^{(i-1)}\| < \epsilon = 1.0^{-10}. \tag{16}$$

4. Convergence Analysis

Before the proposed method is used for the numerical experiment, we conducted a convergence analysis of the quarter-sweep finite difference approximation (Equation 8) using the Lax Equivalence Theorem. According to this theorem, a numerical scheme is convergent if it is consistent and stable. Therefore, we provide the proofs of consistency, stability and convergence of our method as follows.

Theorem 1. The quarter-sweep finite difference approximation to NADE is consistent if the truncation error approaches zero as both the space and time step sizes approach zero.

Proof. Suppose that the quarter-sweep finite difference approximation to NADE has the form,

$$v_p^n - c_1(v_{p+4}^n - 2v_p^n + v_{p-4}^n) + c_2(v_{p+4}^n - v_{p-4}^n) - v_p^{n-1} + f = 0, \tag{17}$$

with Taylor series expansions given by

$$v_p^{n-1} = v_p^n - k\frac{\partial v_p^n}{\partial t} + \frac{k^2}{2}\frac{\partial^2 v_p^n}{\partial t^2} + O(k^3), \tag{18}$$

$$v_{p+4}^n = v_p^n + 4h\frac{\partial v_p^n}{\partial x} + 16h^2\frac{\partial^2 v_p^n}{\partial x^2} + O(h^3), \tag{19}$$

and

$$v_{p-4}^n = v_p^n - 4h\frac{\partial v_p^n}{\partial x} + 16h^2\frac{\partial^2 v_p^n}{\partial x^2} + O(h^3). \tag{20}$$

Substituting Equations 18, 19, and 20 into Equation 17 gives

$$\begin{aligned} v_p^n - c_1&\left[\left(v_p^n + 4h\frac{\partial v_p^n}{\partial x} + 16h^2\frac{\partial^2 v_p^n}{\partial x^2}\right) - 2v_p^n + \left(v_p^n - 4h\frac{\partial v_p^n}{\partial x} + 16h^2\frac{\partial^2 v_p^n}{\partial x^2}\right)\right] \\ +c_2&\left[\left(v_p^n + 4h\frac{\partial v_p^n}{\partial x} + 16h^2\frac{\partial^2 v_p^n}{\partial x^2}\right) - \left(v_p^n - 4h\frac{\partial v_p^n}{\partial x} + 16h^2\frac{\partial^2 v_p^n}{\partial x^2}\right)\right] \\ &- \left(v_p^n - k\frac{\partial v_p^n}{\partial t} + \frac{k^2}{2}\frac{\partial^2 v_p^n}{\partial t^2}\right) + f = 0 \end{aligned} \tag{21}$$

and after simplifying Equation 21, we have

$$k\frac{\partial v_p^n}{\partial t} - 32c_1h^2\frac{\partial^2 v_p^n}{\partial x^2} + 8c_2h\frac{\partial v_p^n}{\partial x} + O(h^3, k^2) = 0. \tag{22}$$

Equation 22 shows that as the space step size h and the time step size k approach zero, the truncation error approaches zero. Thus, the quarter-sweep finite difference approximation to NADE is proven consistent.

Theorem 2. The quarter-sweep finite difference approximation to NADE is unconditionally stable for any space step size h, time step size k, value of source function f, and wave number w.

Proof. Suppose the quarter-sweep finite difference approximation to NADE is given by

$$v_p^n - c_1(v_{p+4}^n - 2v_p^n + v_{p-4}^n) + c_2(v_{p+4}^n - v_{p-4}^n) - v_p^{n-1} + f = 0, \tag{23}$$

and by von Neumann analysis, we have

$$v_p^n = \xi^n e^{iwph}, \tag{24}$$

where ξ^n is the amplification factor at time level n, $i = \sqrt{-1}$, and w is the wave number.

Substituting Equation 24 into Equation 23, we have

$$\begin{aligned} &\xi^n e^{iwph} - c_1(\xi^n e^{iw(p+4)h} - 2\xi^n e^{iwph} + \xi^n e^{iw(p-4)h}) \\ &+c_2(\xi^n e^{iw(p+4)h} - \xi^n e^{iw(p-4)h}) - \xi^{n-1} e^{iwph} + f = 0. \end{aligned} \tag{25}$$

Factoring out $\xi^n e^{iwph}$ from Equation 25, we obtain

$$1 - c_1(e^{4iwh} - 2 + e^{-4iwh}) + c_2(e^{4iwh} - e^{-4iwh}) - \xi^{-1} + f = 0. \tag{26}$$

Using the identities $e^{4iwh} + e^{-4iwh} = 2\cos(4wh)$ and $e^{4iwh} - e^{-4iwh} = 2i\sin(4wh)$, rearrange Equation 26 to yield

$$\xi = \frac{1}{1 - 2c_1(\cos(4wh) - 1) + 2ic_2\sin(4wh) + f}. \tag{27}$$

Since the von Neumann stability criterion is

$$|\xi| \leq 1, \tag{28}$$

the quarter-sweep finite difference approximation to NADE is thus unconditionally stable.

Given Theorems 1 and 2, the Lax Equivalence Theorem assures convergence of our quarter-sweep finite difference approximation to NADE. This theorem asserts that if a linear finite-difference scheme for a well-posed initial-value problem is consistent and stable, it is guaranteed to be convergent. Consequently, as both h and k approach zero, the numerical solution v_p^n converges to the exact solution.

5. Numerical Experiments

To assess the performance of the proposed method, which can be labelled as the quarter-sweep Thomas algorithm (QST), we conducted numerical experiments on two test problems with known exact solutions. The solution domain was discretized into five different sizes of grid (8192, 16384, 32768, 65536, 131072), and the time steps were 0.01, 0.001, 0.0001, 0.00001, and 0.000001. The purpose of experimenting the proposed method using varied sizes of grid and time step is to numerically display the accuracy, stability, and convergence of the numerical solution.

Then, we compared the results of the proposed method with the standard implicit finite difference method with Thomas algorithm (ST). The performance metrics used in this research are the number of iterations per time level (Iteration), computational time in seconds (Time) and maximum absolute error (MAE). The following are the two test problems of NADE used in our experiment [16].

Test Problem 1.

$$\frac{\partial v}{\partial t} = \frac{\partial^2 v}{\partial x^2} - x\frac{\partial v}{\partial x} + F(x,t), \quad x \in (0,1), \quad t > 0, \tag{29}$$

with an exact solution

$$v(x,t) = \tanh\left(\frac{xe^{-t}}{4}\right). \tag{30}$$

Test Problem 2.

$$\frac{\partial v}{\partial t} = \frac{\partial^2 v}{\partial x^2} - \frac{1}{2}v^2\frac{\partial v}{\partial x} + F(v), \quad x \in (0,1), \quad t \in (0,1), \tag{31}$$

with an exact solution

$$v(x,t) = \sqrt{\frac{1}{2} - \frac{1}{2}\tanh\left(\frac{t}{4} + \frac{x}{2}\right)}. \tag{32}$$

The results of the extensive numerical experiment are tabulated in Tables 1 and 2. Then, the graphs of approximate solutions by the proposed method against the exact solutions are presented in Figures 2 and 3.

Table 1. Performance comparison after solving test problem 1.

M	k	Iteration	Time (ST)	Time (QST)	MAE
8192	10^{-3}	3	0.12	0.12	6.419e-3
	10^{-4}	3	1.20	0.97	6.419e-3
	10^{-5}	3	11.66	9.37	6.419e-3
	10^{-6}	3	108.78	90.34	6.419e-3
	10^{-7}	3	1083.43	900.33	6.419e-3
16384	10^{-3}	3	0.22	0.17	6.419e-3
	10^{-4}	3	2.27	1.74	6.419e-3
	10^{-5}	3	20.77	16.96	6.419e-3
	10^{-6}	3	203.44	164.53	6.419e-3
	10^{-7}	3	2032.28	1631.91	6.419e-3
32768	10^{-3}	5	0.61	0.33	6.419e-3
	10^{-4}	3	4.03	3.25	6.419e-3
	10^{-5}	3	40.30	32.60	6.419e-3
	10^{-6}	3	398.32	318.26	6.419e-3
	10^{-7}	3	3971.90	3169.03	6.419e-3
65536	10^{-3}	5	1.26	0.67	6.419e-3
	10^{-4}	5	12.17	6.73	6.419e-3
	10^{-5}	3	81.53	66.84	6.419e-3
	10^{-6}	3	818.26	657.63	6.419e-3
	10^{-7}	3	8073.38	6537.69	6.419e-3
131072	10^{-3}	5	2.48	2.10	6.419e-3
	10^{-4}	5	24.63	15.10	6.419e-3
	10^{-5}	3	168.66	150.49	6.419e-3
	10^{-6}	3	1661.46	1500.96	6.419e-3
	10^{-7}	3	17478.68	15096.43	6.419e-3

TABLE 2. Performance comparison after solving test problem 2.

M	k	Iteration	Time (ST)	Time (QST)	MAE
8192	10^{-3}	7	0.40	0.33	1.708e-2
	10^{-4}	5	2.95	2.54	1.708e-2
	10^{-5}	5	29.63	25.27	1.708e-2
	10^{-6}	5	281.42	238.04	1.708e-2
	10^{-7}	5	2885.42	2419.84	1.708e-2
16384	10^{-3}	7	0.77	0.65	1.708e-2
	10^{-4}	5	5.81	5.05	1.708e-2
	10^{-5}	5	57.47	47.96	1.708e-2
	10^{-6}	5	553.07	472.56	1.708e-2
	10^{-7}	5	5505.96	4725.93	1.708e-2
32768	10^{-3}	7	1.54	1.31	1.708e-2
	10^{-4}	5	11.55	10.00	1.708e-2
	10^{-5}	5	111.18	98.21	1.708e-2
	10^{-6}	5	1092.40	945.84	1.708e-2
	10^{-7}	5	11237.55	9807.42	1.708e-2
65536	10^{-3}	7	3.04	2.61	1.708e-2
	10^{-4}	5	23.04	21.13	1.708e-2
	10^{-5}	5	237.47	189.83	1.708e-2
	10^{-6}	5	2184.33	1884.34	1.708e-2
	10^{-7}	5	21815.16	18519.79	1.708e-2
131072	10^{-3}	7	5.99	5.52	1.708e-2
	10^{-4}	5	47.31	39.42	1.708e-2
	10^{-5}	5	470.47	385.95	1.708e-2
	10^{-6}	5	4314.82	3871.37	1.708e-2
	10^{-7}	5	45371.53	38419.17	1.708e-2

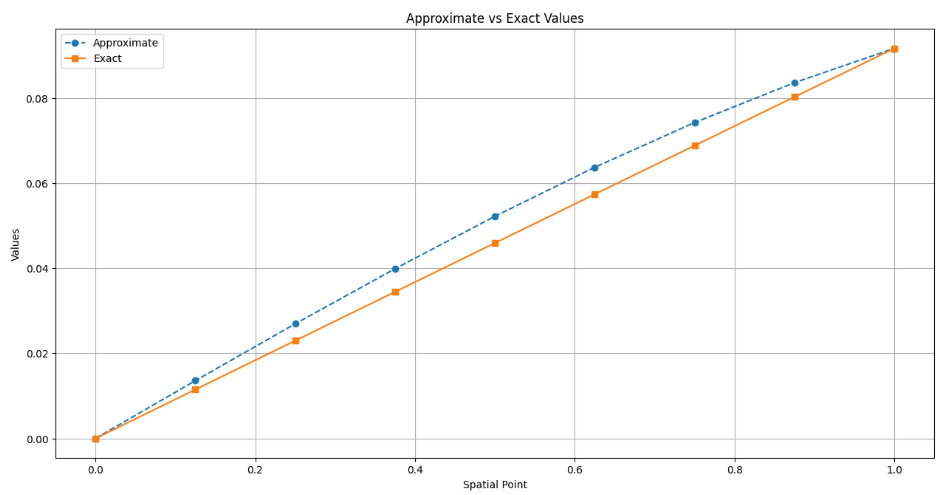

FIGURE 2. Comparison of approximate and exact solutions of test problem 1.

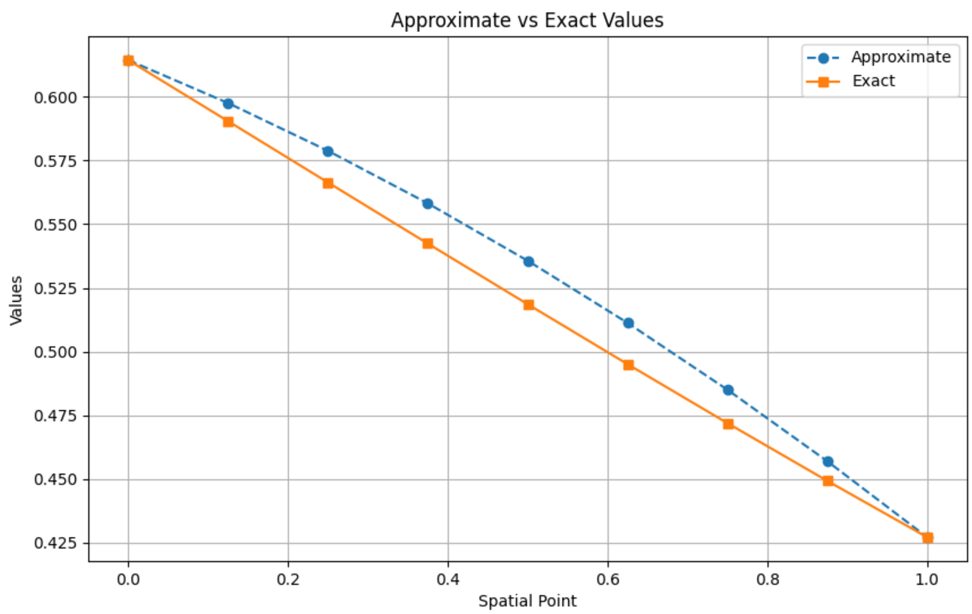

FIGURE 3. Comparison of approximate and exact solutions of test problem 2.

Based on Tables 1 and 2, it can be observed that the MAE values are consistent across all grid sizes and time steps for both ST and QST methods. This result indicates that both methods achieve the same level of accuracy regardless of grid size or time step. Additionally, this consistency in MAE suggests that the accuracy of the numerical solutions is not significantly impacted by either method or the grid and time step sizes within the tested ranges.

Furthermore, the number of iterations required by both methods to solve the system of equations at each time level remains relatively stable across different grid sizes and time steps. The number of iterations required to solve test problem 1 is mostly 3 per time level, with some instances at 5. Meanwhile, test problem 2 mostly required 5 iterations per time level, with a few cases needing 7. This stability in the number of iterations per time level indicates that both methods converge quickly, with the QST method slightly reducing the iteration count compared to the ST method.

In terms of computational time, the QST method consistently required less computation time than the ST method across all configurations. The difference in computation time becomes more significant when the grid size is larger, and the time step is smaller. In other words, this performance improvement scales with larger grid sizes and smaller time steps, with QST consistently achieving faster results by reducing computational load. Overall, the QST method demonstrates clear efficiency benefits in reducing computational time across all configurations without compromising accuracy. This efficiency becomes more pronounced as the grid size increases and the time steps become smaller.

6. Conclusion

This paper presented the quarter-sweep finite difference approximation combined with the Thomas algorithm as an efficient numerical method for solving NADEs. The proposed quarter-sweep Thomas method significantly reduced computational time when solving large systems of equations, especially when the grid size or resolution is high. Despite the reduced computation time, the proposed method maintained the same level of accuracy across various configurations. The performance advantage of the proposed method increased as the grid and time resolutions increased, indicating its scalability for more complex problems. This research suggests that the proposed method would be highly advantageous in multi-dimensional or high-resolution applications where computational efficiency is critical. Future work will focus on extending this method to higher dimensions and exploring its applicability to more complex nonlinear equations.

Acknowledgements. The authors wish to acknowledge the financial support provided by Universiti Malaysia Sabah for the publication of this research. This research is part of the project under the research grant DKP0058.

Authors' Contributions. All authors have read and approved the final version of the manuscript. The authors contributed equally to this work.

Conflicts of Interest. The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] Y.E. Aghdam, H. Mesgrani, M. Javidi, O. Nikan, A Computational Approach for the Space-Time Fractional Advection–Diffusion Equation Arising in Contaminant Transport through Porous Media, Eng. Comput. 37 (2021), 3615–3627. https://doi.org/10.1007/s00366-020-01021-y.

[2] V.R. Hosseini, M. Koushki, W.-N. Zou, The Meshless Approach for Solving 2D Variable-Order Time-Fractional Advection–Diffusion Equation Arising in Anomalous Transport, Eng. Comput. 38 (2022), 2289–2307. https://doi.org/10.1007/s00366-021-01379-7.

[3] S. Kwak, S. Ham, J. Kim, Dispersion and Phase Exchange Process of Chemically Reactive Solute Through Circular Tube, J. Appl. Fluid Mech. 17 (2024), 726–741. https://doi.org/10.47176/jafm.17.4.2305.

[4] P. Klankaew, N. Pochai, A Numerical Groundwater Quality Assessment Model Using the Cubic Spline Method, IAENG Int. J. Appl. Math. 54 (2024), 111–116.

[5] A.F. Aljohani, A. Alsisi, S. Althobaiti, et al. Groundwater Pollution Equation: Lie's Symmetry Analysis and Numerical Consideration, Partial Differ. Equ. Appl. Math. 11 (2024), 100861. https://doi.org/10.1016/j.padiff.2024.100861.

[6] C.N. Angstmann, B.I. Henry, B.A. Jacobs, A.V. McGann, Numeric Solution of Advection–Diffusion Equations by a Discrete Time Random Walk Scheme, Numer. Methods Partial Differ. Equ. 36 (2020), 680–704. https://doi.org/10.1002/num.22448.

[7] S.R. Jena, A. Senapati, One-Dimensional Heat and Advection-Diffusion Equation Based on Improvised Cubic B-Spline Collocation, Finite Element Method and Crank-Nicolson Technique, Int. Commun. Heat Mass Transfer 147 (2023), 106958. https://doi.org/10.1016/j.icheatmasstransfer.2023.106958.

[8] A. Mohebbi, M. Dehghan, High-Order Compact Solution of the One-Dimensional Heat and Advection–Diffusion Equations, Appl. Math. Model. 34 (2010), 3071–3084. https://doi.org/10.1016/j.apm.2010.01.013.

[9] G. Gurarslan, Sixth-Order Combined Compact Finite Difference Scheme for the Numerical Solution of One-Dimensional Advection-Diffusion Equation with Variable Parameters, Mathematics 9 (2021), 1027. https://doi.org/10.3390/math9091027.

[10] T. Nazir, M. Abbas, A.I.M. Ismail, A.A. Majid, A. Rashid, The Numerical Solution of Advection–Diffusion Problems Using New Cubic Trigonometric B-Splines Approach, Appl. Math. Model. 40 (2016), 4586–4611. https://doi.org/10.1016/j.apm.2015.11.041.

[11] J.V.L. Chew, J. Sulaiman, A. Sunarto, Numerical Approximation to Porous Medium Equation Using a Quarter-Sweep Based Finite Difference and Explicit Four-Points Group, Procedia Comput. Sci. 227 (2023), 849–857. https://doi.org/10.1016/j.procs.2023.10.591.

[12] A.A. Dahalan, A. Saudi, Computational Analysis of a Mobile Path-Planning via Quarter-Sweep Two-Parameter Over-Relaxation, in: X.-S. Yang, R.S. Sherratt, N. Dey, A. Joshi (Eds.), Proceedings of Eighth International Congress on Information and Communication Technology, Springer, Singapore, 2023: pp. 297–309. https://doi.org/10.1007/978-981-99-3243-6_23.

[13] A.A. Dahalan, A. Saudi, J. Sulaiman, Pathfinding for Mobile Robot Navigation by Exerting the Quarter-Sweep Modified Accelerated Overrelaxation (QSMAOR) Iterative Approach via the Laplacian Operator, Model. Simul. Eng. 2022 (2022), 9388146. https://doi.org/10.1155/2022/9388146.

[14] J.V.L. Chew, E. Aruchunan, A. Sunarto, J. Sulaiman, Quarter-Sweep Successive Over-relaxation Approximation to the Solution of Porous Medium Equations, IAENG Int. J. Appl. Math. 52 (2022), 466–472.

[15] L.H. Ali, J. Sulaiman, A. Saudi, Iterative Method for Solving Nonlinear Fredholm Integral Equations Using Quarter-Sweep Newton-PKSOR Method, in: D.-K. Kang, R. Alfred, Z.I.B.A. Ismail, A. Baharum, V. Thiruchelvam (Eds.), Proceedings of the 9th International Conference on Computational Science and Technology, Springer, Singapore, 2023: pp. 33–46. https://doi.org/10.1007/978-981-19-8406-8_3.

[16] J. Zhang, F. Wang, S. Nadeem, M. Sun, Simulation of Linear and Nonlinear Advection-Diffusion Problems by the Direct Radial Basis Function Collocation Method, Int. Commun. Heat Mass Transfer 130 (2022), 105775. https://doi.org/10.1016/j.icheatmasstransfer.2021.105775.