

Research Article

An Unconditionally Stable Positivity-Preserving Scheme for the One-Dimensional Fisher–Kolmogorov–Petrovsky–Piskunov Equation

Sangkwon Kim,¹ Chaeyoung Lee,¹ Hyun Geun Lee,² Hyundong Kim,¹ Soobin Kwak,¹ Youngjin Hwang,¹ Seungyoon Kang,¹ Seokjun Ham,¹ and Junseok Kim¹ 

¹Department of Mathematics, Korea University, Seoul 02841, Republic of Korea

²Department of Mathematics, Kwangwoon University, Seoul 01897, Republic of Korea

Correspondence should be addressed to Junseok Kim; cfdkim@korea.ac.kr

Received 2 September 2021; Accepted 1 October 2021; Published 19 October 2021

Academic Editor: Youssef N. Raffoul

Copyright © 2021 Sangkwon Kim et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this study, we present an unconditionally stable positivity-preserving numerical method for the Fisher–Kolmogorov–Petrovsky–Piskunov (Fisher–KPP) equation in the one-dimensional space. The Fisher–KPP equation is a reaction-diffusion system that can be used to model population growth and wave propagation. The proposed method is based on the operator splitting method and an interpolation method. We perform several characteristic numerical experiments. The computational results demonstrate the unconditional stability, boundedness, and positivity-preserving properties of the proposed scheme.

1. Introduction

In an infinitely long domain, the behavior of virile mutant propagation can be modeled by Fisher's equation [1]. It was also independently studied by Kolmogorov, Petrovsky, and Piskunov [2]. There are some numerical methods to solve the Fisher–Kolmogorov–Petrovsky–Piskunov (Fisher–KPP) or fractional Fisher–KPP equations such as the spectral collocation-type method [3, 4], Adams–Bashforth–Moulton method [5], and Mittag-Leffler kernel [6]. In addition, various finite difference methods [7–10] and the radial basis functions [11] have been developed.

In this paper, we present an unconditionally stable positivity-preserving numerical method for the Fisher–KPP equation:

$$\frac{\partial u}{\partial t}(x, t) = D \frac{\partial^2 u}{\partial x^2}(x, t) + Ku(x, t)(1 - u^p(x, t)), \quad (1)$$

$$x \in \Omega, t > 0$$

where $u(x, t)$ is the population density at position x and time t . Here, D is the diffusion coefficient, K is the

coefficient of nonlinearity, and p is a positive integer. There exist several other unique approaches to solve the Fisher–KPP equation. Saeed and Mustafa [12] used the reduced differential transformation method to solve the equation and showed that their result coincides with the exact solution. From a biological point of view, the population density should be nonnegative and bounded from above by one if the density is scaled by the carrying capacity. El-Hachem et al. [13] studied approaches to modelling biological invasion and recession using the Fisher–KPP model. In addition, they utilized the Fisher–Stefan model, another generalization of the Fisher–KPP-type model that simulates biological recession. Xu et al. [14] proposed a nonlinear finite volume scheme for two-dimensional diffusion equations which preserves the maximum principle. Fourth-order Fisher–KPP equation is one of the widely studied and extended variations, and various methods have been proposed to solve it. Kadri and Omrani [15] utilized the extrapolation technique, and Başhan et al. [16] applied the differential quadrature method. An explicit positivity-preserving numerical method for the Fisher–KPP equation has been

proposed [7]. Furthermore, the positivity-preserving schemes for the Fisher–KPP equation were developed using continuous finite element discretizations [17], and discontinuous Galerkin schemes were proved to be entropy-stable [18]. Ilati and Dehghan [19] proposed the local boundary integral equation method to solve the extended Fisher–KPP equation. The presented method is based on generalized moving least squares, which can be used to directly approximate local weak forms. Oruç [20] developed an efficient wavelet collocation method for the nonlinear 2D Fisher–KPP equation and extended Fisher–Kolmogorov equation. Castillo and Gómez [21] used the local discontinuous Galerkin method with the symmetric Strang operator splitting scheme to solve Fisher–KPP. Kabir [22] considered a model with travelling wave solutions with a qualitatively similar structure to that observed in the Fisher–KPP equation and performed numerical simulations. McCue et al. [23] studied exact sharp-fronted travelling wave solutions of the Fisher–KPP equation to the Fisher–Stefan-type moving boundary problem. They demonstrated large time evolutionary dynamics of the exact travelling wave solution and considered these travelling waves for cell migration and proliferation mathematical models. Recently, numerical research studies for the nonlocal Fisher–KPP equation have been actively conducted to validate the asymptotic behavior [24], positivity of the solution [25, 26], and stability of the travelling wave solution [27, 28].

However, if p in equation (1) is large, then the Fisher–KPP equation becomes a very stiff problem to be numerically solved. To resolve this stiff problem, we propose the operator splitting method (OSM) with a recently developed interpolation method [29]. The computational results demonstrate the unconditional stability and positivity-preserving properties of the proposed scheme.

The contents of this article are as follows. In Section 2, we present the proposed numerical scheme for the Fisher–KPP equation. In Section 3, we conduct computational experiments to verify the robustness of the proposed scheme. In Section 4, conclusions are drawn.

2. Numerical Solution Algorithm

To efficiently solve equation (1), we apply a recently developed interpolation method [29]. Now, we present the numerical solution scheme of the Fisher–KPP equation in one-dimensional space $\Omega = (L_x, R_x)$. Let N_x be an integer and $h = (R_x - L_x)/N_x$ be the uniform spatial step size. Let u_i^n be the numerical approximation of $u(x_i, t_n)$, where $x_i = L_x + (i - 0.5)h$ for $i = 1, 2, \dots, N_x$ and $t_n = n\Delta t$. Using the OSM, we divide equation (1) into two parts: diffusion and nonlinear equations. First, we solve the diffusion equation using the fully implicit Euler finite difference method (FDM):

$$\frac{\partial u}{\partial t}(x, t) = D \frac{\partial^2 u}{\partial x^2}(x, t). \quad (2)$$

Equation (2) can be discretized using the FDM as

$$\frac{u_i^{n+(1/2)} - u_i^n}{\Delta t} = D \frac{u_{i-1}^{n+(1/2)} - 2u_i^{n+(1/2)} + u_{i+1}^{n+(1/2)}}{h^2}, \quad \text{for } i = 1, \dots, N_x. \quad (3)$$

Equation (3) is reorganized as

$$-\frac{D}{h^2}u_{i-1}^{n+(1/2)} + \left(\frac{2D}{h^2} + \frac{1}{\Delta t}\right)u_i^{n+(1/2)} - \frac{D}{h^2}u_{i+1}^{n+(1/2)} = \frac{u_i^n}{\Delta t}. \quad (4)$$

For the boundary condition, we use the zero Neumann boundary condition: and $u_{N_x+1}^{n+(1/2)} = u_{N_x}^{n+(1/2)}$ for all values of $n + (1/2)$. The Dirichlet boundary condition can also be used. Then, we solve equation (4) using the Thomas algorithm [30]. Second, we solve the nonlinear equation using an interpolation method:

$$\frac{\partial u}{\partial t}(x, t) = Ku(x, t)(1 - u^p(x, t)). \quad (5)$$

For simplicity of exposition, let $K = 1$; then, equation (5) becomes

$$\frac{\partial u}{\partial t}(x, t) = u(x, t)(1 - u^p(x, t)). \quad (6)$$

To solve equation (6), we use a recently developed interpolation method [29]. Let us assume that the time step Δt is given. We uniformly discretize the interval $[0, 1]$, i.e., $I = \{I_s | I_s = (s - 1)/(M - 1), \text{ for } s = 1, \dots, M\}$. Note that M is a positive integer. We solve equation (6) on I with a smaller time step $\Delta\tau = \Delta t/N_\tau$ until $t = \Delta t$. Here, N_τ will be defined later. Figure 1 shows a simulation result with the initial condition $\Phi_s(0) = I_s$ for $s = 1, \dots, M$. That is, for $m = 0, \dots, N_\tau - 1$,

$$\Phi_s((m + 1)\Delta\tau) = \Phi_s(m\Delta\tau) + \Delta\tau\Phi_s(m\Delta\tau)(1 - \Phi_s^p(m\Delta\tau)). \quad (7)$$

Once we compute all $\Phi_s(\Delta t)$ for $s = 1, \dots, M$, then we will use these values when we numerically solve equation (6) by using an interpolation.

Next, we find a value of N_τ appropriate to stably integrate equation (7) for given Δt . If $I_s = 0$ or $I_s = 1$, then from equation (7), we have

$$\Phi_s((m + 1)\Delta\tau) = I_s, \quad \text{for } m = 0, \dots, N_\tau - 1. \quad (8)$$

Next, when $0 < I_s < 1$ and $\Phi_s(0) = I_s$, we want $0 \leq \Phi_s((m + 1)\Delta\tau) \leq 1$ for $m = 0, \dots, N_\tau - 1$. Here, to show $0 \leq \Phi_s(\Delta\tau) \leq 1$, by mathematical induction, it is sufficient to show $0 \leq \Phi_s((m + 1)\Delta\tau) \leq 1$. Then, let us suppose that $0 < I_s < 1$, $\Phi_s(0) = I_s$, and $m = 0$; from (7), we have

$$\Phi_s(\Delta\tau) = I_s + \Delta\tau I_s(1 - I_s^p). \quad (9)$$

As shown in Figure 2, $I_s(1 - I_s^p) > 0$ for $0 < I_s < 1$; therefore, we have $\Phi_s(\Delta\tau) > 0$ from equation (9).

Because $0 \leq I_s + \Delta\tau I_s(1 - I_s^p)$ is always satisfied, we only need to find $\Delta\tau$ satisfying the following condition:

$$\Delta\tau \leq \frac{1 - I_s}{I_s(1 - I_s^p)}. \quad (10)$$

The right-hand side of inequality (10) is decreasing with respect to I_s ; see Figure 3.

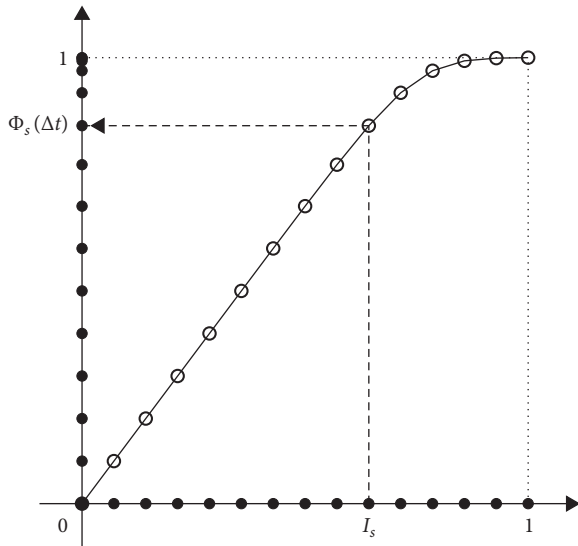


FIGURE 1: Numerical solution $\Phi(\Delta t)$ of equation (6) on the uniform grid I .

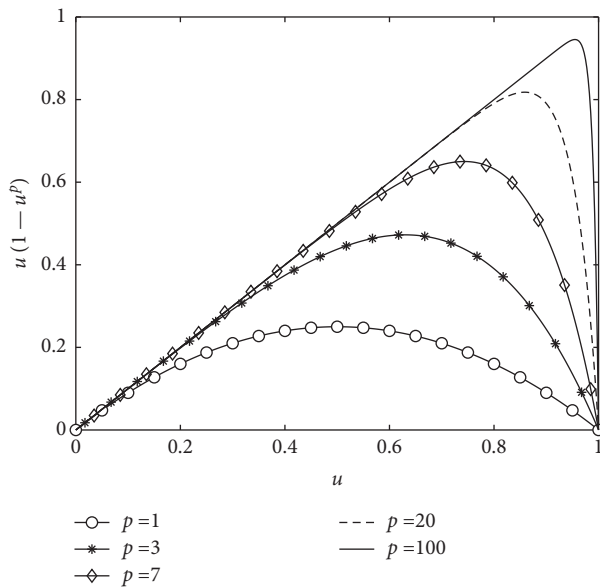


FIGURE 2: Nonlinear part $u(1-u^p)$ of equation (1) for $p = 1, 3, 7, 20$, and 100 .

Therefore, we consider the limit of the term as I_s approaches 1. By using L'Hopital's rule, we have

$$\Delta\tau \leq \lim_{I_s \rightarrow 1} \frac{1 - I_s}{I_s(1 - I_s^p)} = \lim_{I_s \rightarrow 1} \frac{-1}{1 - (p+1)I_s^p} = \frac{1}{p}. \quad (11)$$

From the definition of $\Delta\tau = \Delta t/N_\tau$ and stability condition (11), $N_\tau \geq p\Delta t$ should be satisfied. Therefore, unless otherwise specified, we set $N_\tau = [p\Delta t] + 1$, where $[x]$ is the greatest integer less than or equal to x . Given the intermediate solution from the first step, $u_i^{n+(1/2)}$, the second step is to solve equation (6) by using the precomputed values and interpolation. We can find an interval $[I_s, I_{s+1}]$ which

satisfies $I_s \leq u_i^{n+(1/2)} \leq I_{s+1}$ for some s and $1 \leq s \leq M-1$. Then, we define

$$u_i^{n+1} = \frac{I_{s+1} - u_i^{n+(1/2)}}{I_{s+1} - I_s} \Phi_s(\Delta t) + \frac{u_i^{n+(1/2)} - I_s}{I_{s+1} - I_s} \Phi_{s+1}(\Delta t), \quad (12)$$

which is schematically illustrated in Figure 4.

We should note that, in the solution algorithm, the first step is a diffusion equation solver, which is unconditionally stable and satisfies the discrete maximum principle. The second step is an interpolation, which is unconditionally stable, monotonically increasing, and bounded by one. Because two steps are both unconditionally stable and positive preserving, therefore, the full step is unconditionally stable, positive preserving, and bounded by one.

3. Numerical Experiments

In this section, we conduct various computational experiments such as the effect of the model and numerical parameters of the proposed algorithm. In order to demonstrate the performance of the proposed algorithm, we conduct comparative experiments with the previous work.

3.1. Convergence Tests. Let us consider an initial condition for equation (1) on the computational domain $\Omega = (-30, 120)$ with $D = 1$ and $K = 1$:

$$u(x, 0) = \left[0.5 - 0.5 \tanh\left(\frac{px}{2\sqrt{2p+4}}\right) \right]^{2/p}. \quad (13)$$

The exact solution is given in [31]:

$$u(x, t) = \left\{ 0.5 - 0.5 \tanh\left[\frac{p}{2\sqrt{2p+4}}\left(x - \frac{(p+4)t}{\sqrt{2p+4}}\right)\right] \right\}^{2/p}. \quad (14)$$

We run the simulation up to terminal time $t = 46$ with different time steps: $\Delta t = 0.4, 0.2, 0.1$, and 0.05 . The other numerical parameter values used are $M = 800$ and $N_x = 300$. Figure 5 shows the numerical results and the exact solution (14) with $p = 1$. We can observe that the numerical solutions converge to the exact solution as we refine the time step.

Table 1 shows the l_2 -norm error and convergence rate of numerical solutions with the same conditions above except for the final time $t = 4$ and the fixed $N_x = 1024$, where N_t is the number of time iterations of the simulation for each $\Delta t = t/N_t$. We can observe that the numerical scheme is first-order accurate in time.

Next, we investigate the accuracy of numerical results on the value of M . Figure 6 shows the convergence of numerical solutions with respect to M values on subdomain $(75, 105)$. Here, we used the same parameter values as before except M values and $\Delta t = 0.01$.

Last, we study the convergence of numerical results with respect to h . The applied conditions are the same as the time regarding simulation, except for $\Delta t = 0.0001$. With all the other parameters fixed, the spatial step size N_x is refined to calculate

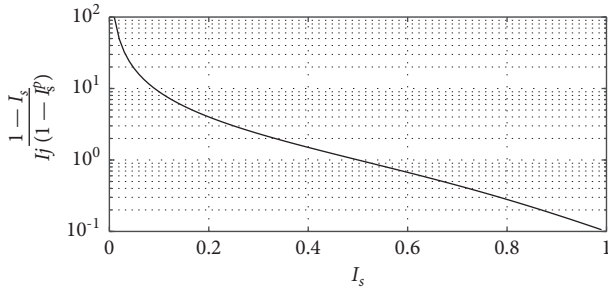


FIGURE 3: Semilog plot of $(1 - I_s) / [I_s (1 - I_s^p)]$ with respect to I_s . Here, $p = 10$ is used.

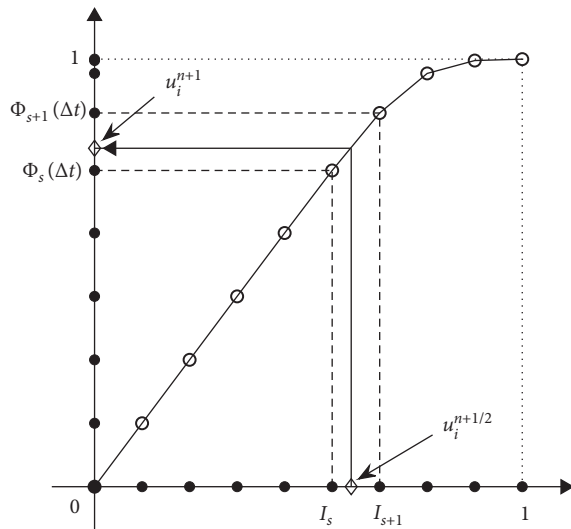


FIGURE 4: Schematic diagram of the interpolation.

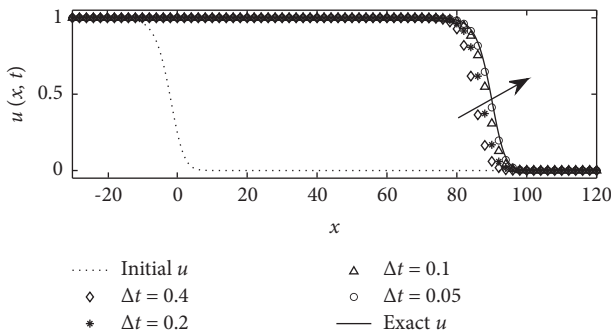


FIGURE 5: Convergence of the numerical solutions with respect to Δt values at $t = 46$.

the convergence rate. l_2 -norm error and convergence rate are shown in Table 2, and it can be observed that the numerical scheme is second-order accurate in space.

Time convergence rate and spatial convergence rate are clearly different: first order and second order. This is due to the fact that the interpolation method is second order in both time and space, yet the fully implicit Euler method is first order in time and second order in space. Therefore, after combining the two methods using the OSM, time convergence rate is one, while space convergence rate is two.

3.2. Stability Test. We demonstrate the stability of the proposed scheme with a numerical simulation with initial condition (13) for (1) on the computational domain $\Omega = (-30, 120)$ with $D = 1$, $K = 1$, $M = 800$, and $N_x = 600$. We run the simulation up to terminal time $t = 30$ with different time steps: $\Delta t = 0.03, 0.3$, and 3 . Figure 7 shows the numerical results and the exact solution (14) with $p = 1$. It can be observed that the numerical solutions do not blow up with large time step. Therefore, the proposed scheme is stable. However, when a large time system is used, the accuracy of the numerical solution is low.

3.3. Effect of p . Now, let us consider the effect of p on the evolution dynamics. Let us consider equation (6). By using the explicit Euler's method, we present the evolution of u with the initial condition $u^0 = 0.1$ on various values of p . Here, we use the time step $\Delta t = 0.0001$ and the terminal time $t = 7$:

$$u^{n+1} = u^n + \Delta t u^n [1 - (u^n)^p]. \quad (15)$$

Figure 8 shows the evolution of u with $p = 1, 3, 7, 20$, and 100 . As the parameter p value increases, the solution u evolves faster.

However, as shown in Figure 8, there is little difference between $p = 20$ and $p = 100$. Next, we investigate the numerical solution of equation (9) on I as $uM = 400$, $\Delta t = 0.0001$, and $t = 6$; see Figure 9.

To clearly identify this effect, numerical experiments are conducted using initial condition $u(x, 0) = 0.3 \cos(\pi x/5) + 0.3$. We set the computational domain $\Omega = 10$, $D = 1.0$, $K = 0.1$, terminal time $t = 0.5$, $\Delta t = 0.001$, $M = 800$, and $N_x = 400$. Figure 10 shows the numerical solutions with various values of $p = 1, 3, 7, 20$, and 100 .

The evolution with large p is faster than that with small p , especially near $u = 0.5$. However, evolution with p in the vicinity of $u = 0$ and $u = 1$ has a small difference.

Next, let us set an initial condition equation (13) on the computational domain $\Omega = (-30, 120)$ with $D = 1$ and $K = 1$. We run the simulation up to time $t = 10$ with $\Delta t = 0.005$, $M = 800$, and $N_x = 300$. Figure 11 shows the numerical results with different p values on subdomain $(-10, 40)$.

Figure 12 shows the numerical solution with an initial condition $u(x, 0) = 1$ if $x \leq 0$; otherwise, $u(x, 0) = 0$ for different p values. Here, we used the same parameter values as before. We can observe that the numerical solutions become stiffer as the p value increases.

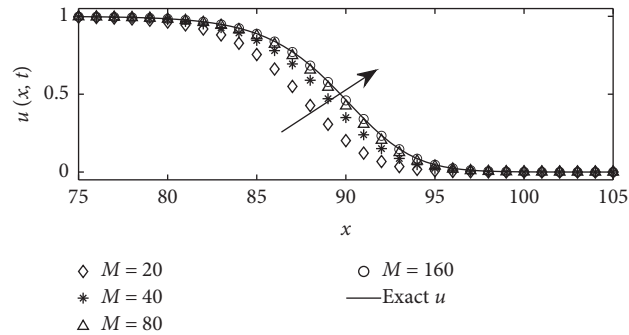
3.4. Positive Preserving. We conduct numerical experiment using the initial condition

$$u(x, 0) = 0.5 + 0.4 \cos\left(\frac{\pi x}{10}\right) + 0.09 \text{rand}(x). \quad (16)$$

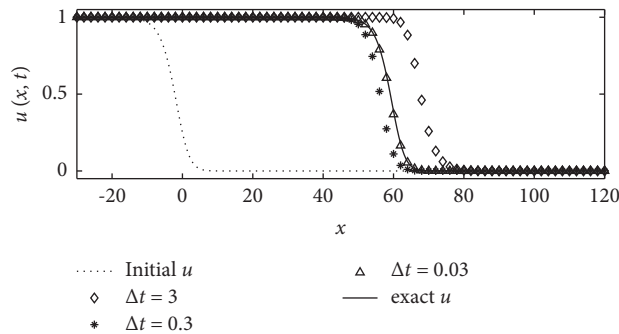
where $-1 < \text{rand}(x) < 1$ to confirm that the algorithm satisfies positive preserving and boundedness by one. The parameters used in the numerical experiment are as

TABLE 1: l_2 -norm error and temporal convergence rate of numerical solutions.

	$N_t = 16$	Rate	$N_t = 32$	Rate	$N_t = 64$	Rate	$N_t = 128$
Error	$8.8861e-3$	0.9328	$4.6551e-3$	0.9636	$2.3870e-3$	0.9812	$1.2091e-3$

FIGURE 6: Convergence of the numerical solutions with respect to M values.TABLE 2: l_2 -norm error and temporal convergence rate of numerical solutions.

	$N_x = 51$	Rate	$N_x = 101$	Rate	$N_x = 201$	Rate	$N_x = 401$
Error	$2.1493e-3$	1.9265	$5.6542e-4$	1.9881	$1.4253e-4$	2.0423	$3.4576e-5$

FIGURE 7: Numerical solutions with respect to three different temporal step sizes $\Delta t = 0.03, 0.3$, and 3 at $t = 30$.

follows: $\Omega = (0, 10)$, $D = 1$, $K = 1$, $\Delta t = 0.0005$, $M = 800$, and $N_x = 101$. Figures 13(a) and 13(b) show the evolution of numerical solutions u for $p = 1$ and $p = 20$ at $t = 1.2$. Figures 13(c) and 13(d) show the temporal evolution of the maximum and minimum of u over time when $p = 1$ and $p = 20$, respectively. We can confirm that $0 < u < 1$ is satisfied regardless of the p value. The proposed algorithm is a positive-preserving scheme.

3.5. Comparison with the Previous Work. To confirm the performance of our proposed method, we compare it with the recently developed method. Izadi [10] proposed a second-order accurate and unconditionally stable FDM for the classical Fisher-KPP equation:

$$\begin{aligned}
 & -\alpha R(u_{i-1}^{n+1} + u_{i+1}^{n+1}) + (1 + 2\alpha R - \beta \Delta t(1 - (u_i^n)^p) + (1 - \beta)\Delta t u_i^n)u_i^{n+1} \\
 & = (1 - 2R(1 - \alpha) + \Delta t(1 - \beta))u_i^n + R(1 - \alpha)(u_{i-1}^n + u_{i+1}^n),
 \end{aligned} \tag{17}$$

where $\alpha = \beta = 1/2$ and $R = D\Delta t/h^2$ with $D = 1$ and $K = 2$. Let us consider an initial condition on $\Omega = (-30, 40)$: $u(x, 0) = 1$ if $x \leq 0$; otherwise, $u(x, 0) = 0$. We run the simulation up to time $t = 5$ with $\Delta t = 0.05$, $M = 800$, and $N_x = 60$. Figure 14 shows the numerical solutions with various values of $p = 1, 80$, and 800 . When $p = 1$, the numerical solutions for both methods converge to the exact solution. For the large value p , the proposed method obtained a stable numerical solution, but the Izadi method [10] obtained an unstable numerical solution; see Figures 14(b) and 14(c).

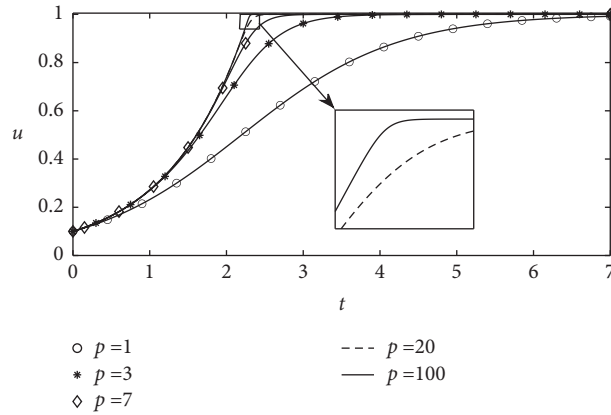


FIGURE 8: Evolution of u with $p = 1, 3, 7, 20$, and 100 , $\Delta t = 0.0001$, and $t = 7$.

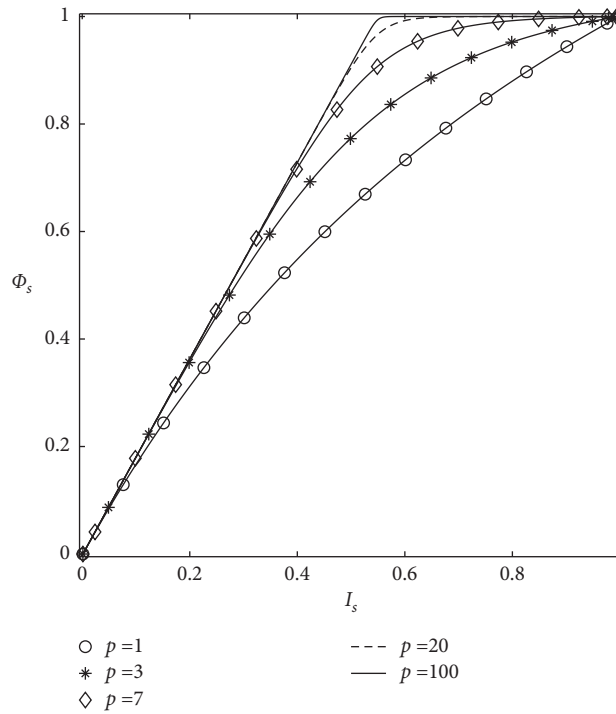


FIGURE 9: Numerical solutions of Φ with $p = 1, 3, 7, 20$, and 100 , $M = 400$, $\Delta t = 0.0001$, and $t = 6$.

3.6. Second-Order Accurate Scheme. In this section, we present a second-order accurate scheme in time. Previously, the proposed scheme, which is a first-order accurate scheme in time, separates the governing equation into two parts over a time step: the diffusion equation \mathcal{L} and nonlinear equation \mathcal{N} . That is, the numerical solution is $u(x, t + \Delta t) = (\mathcal{L}^{\Delta t} \circ \mathcal{N}^{\Delta t})u(x, t) + \mathcal{O}(\Delta t^2)$ with time step size Δt . For a second-order accurate scheme, meanwhile, $u(x, t + \Delta t) = (\mathcal{L}^{\Delta t/2} \circ \mathcal{N}^{\Delta t} \circ \mathcal{L}^{\Delta t/2})u(x, t) + \mathcal{O}(\Delta t^3)$. Note that all numerical schemes used for \mathcal{L} and \mathcal{N} should be at least second-order accurate in time. Please refer to [29] for more details. Here, we solve the diffusion equation using the Crank–Nicolson scheme and use a sufficiently large N_τ for the nonlinear equation to guarantee the second-order

accuracy in time. To verify the convergence of the numerical scheme, we set the same initial condition (13) and parameters as in Section 3.1. Figure 15 illustrates that the numerical solutions converge to the exact solution as the time step is refined. In addition, Table 3 indicates that the entire scheme is second-order accurate in time.

We tested the spatial convergence for the aforementioned scheme. It can be assumed that it is second-order accurate in space since both Crank–Nicolson method and interpolation method are second order. To confirm the convergence, the same condition as the spatial convergence test in Section 3.1 is used. The derived results are given in Table 4, and it shows that the method is second-order accurate in space.

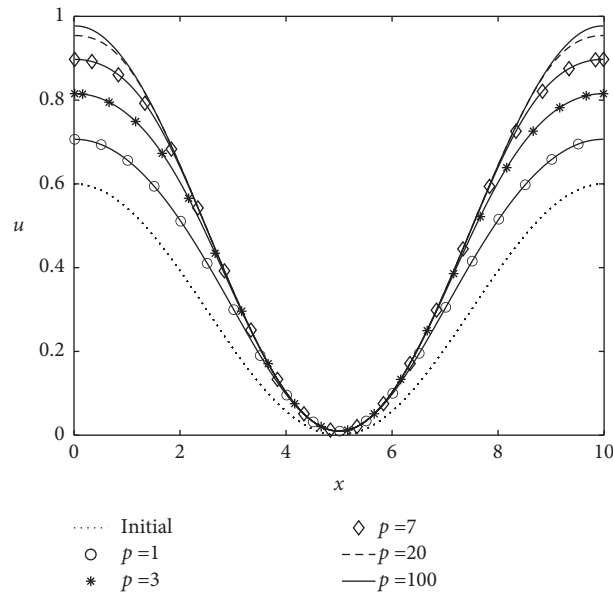


FIGURE 10: Snapshots of the numerical solutions of the Fisher-KPP equation with the initial condition $u(x, 0) = 0.3 \cos(\pi x/5) + 0.3$. Here, $t = 0.5$ with $\Delta t = 0.001$ and $p = 1, 3, 7, 20$, and 100 .

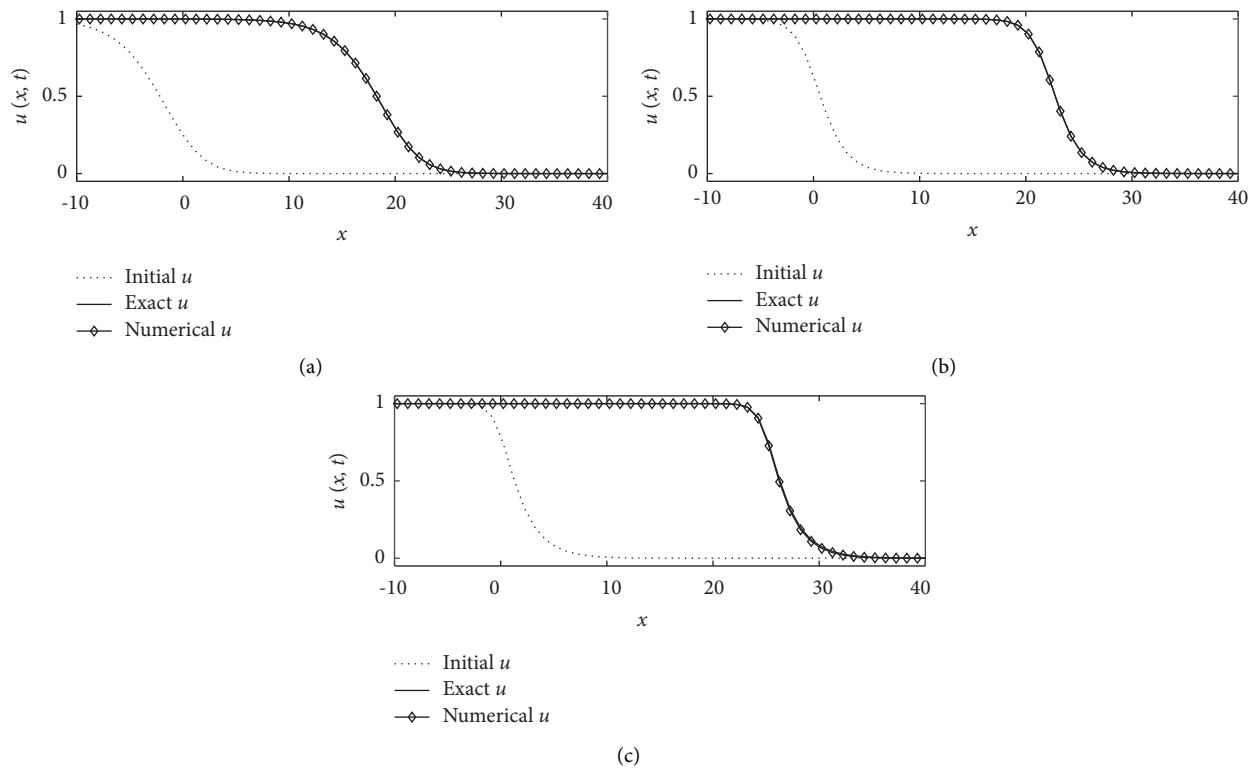
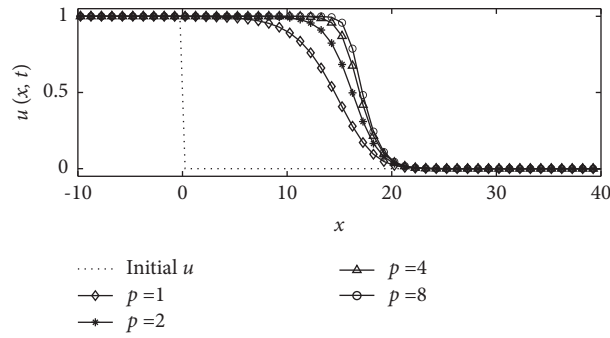
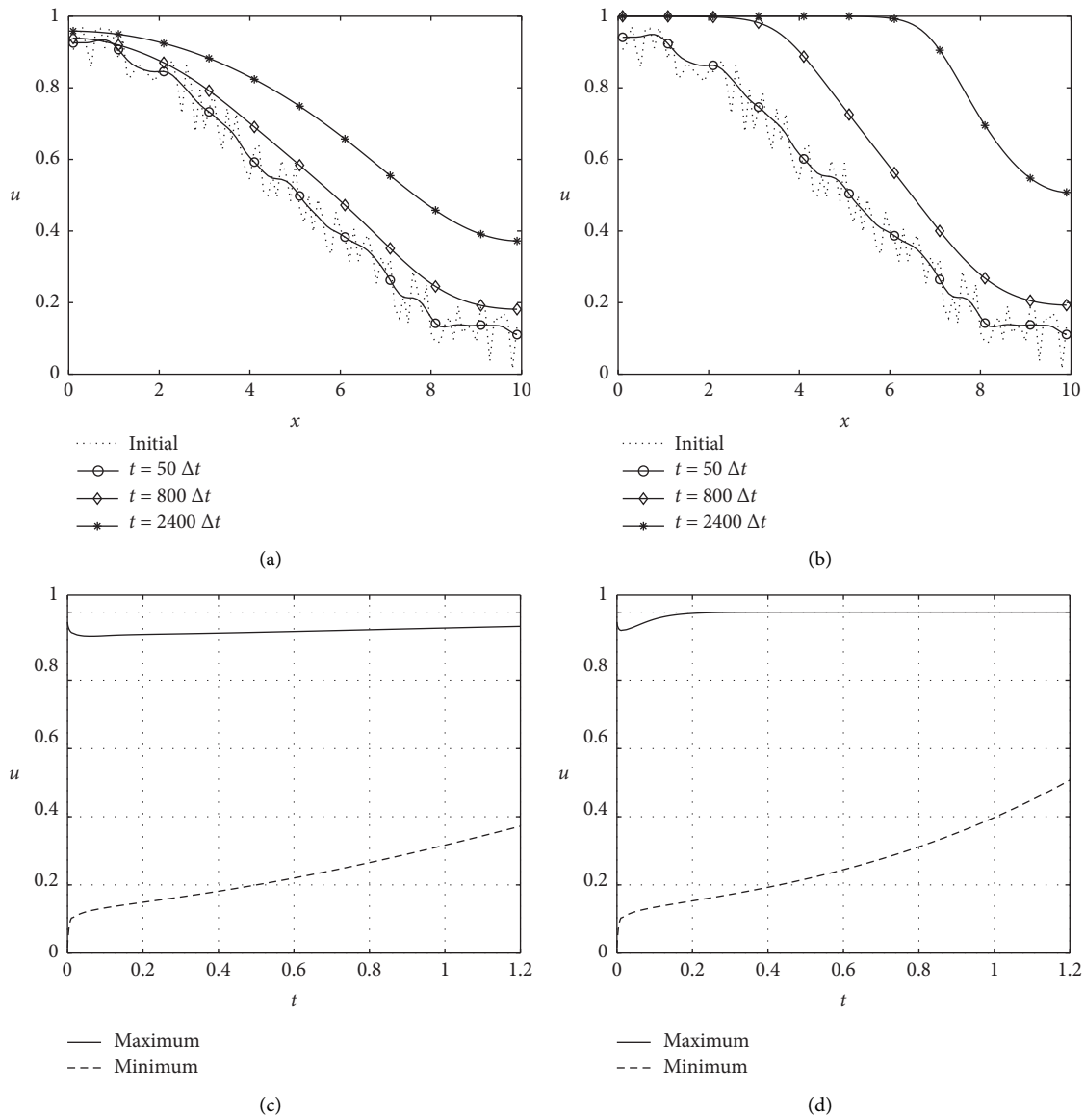


FIGURE 11: Numerical solutions with initial condition equation (13) for different p values. (a-c) Solutions with $p = 1, 3$, and 6 , respectively.

FIGURE 12: Numerical solutions with different p values.FIGURE 13: (a, b) Numerical solution of u with (a) $p = 1$ and (b) $p = 20$. (c, d) Temporal evolution of the minimum and maximum of u when (c) $p = 1$ and (d) $p = 20$.

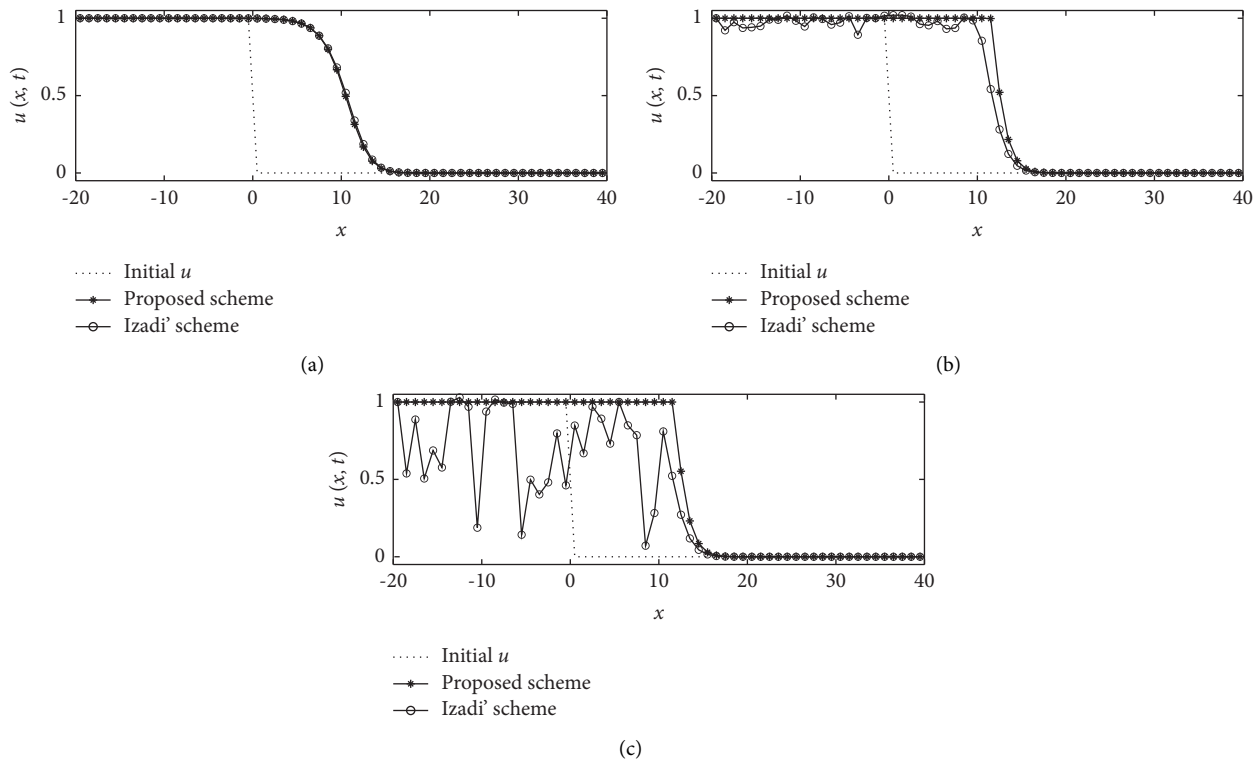


FIGURE 14: Numerical solution comparison of the proposed method and recently developed method [10] for various values of $p = 1, 80$, and 800 . (a–c) Numerical solutions with $p = 1, 80$, and 800 , respectively.

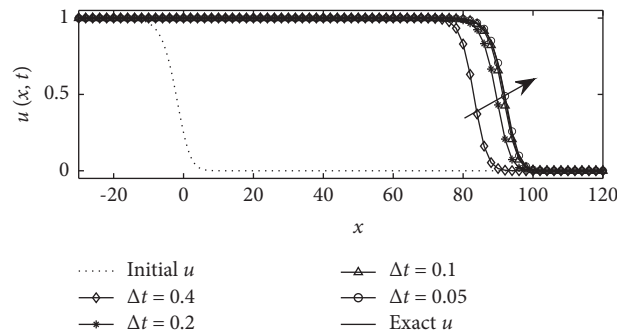


FIGURE 15: Convergence of the numerical solutions with respect to Δt values.

TABLE 3: l_2 -norm error and temporal convergence rates of the second-order accurate scheme.

	$N_t = 16$	Rate	$N_t = 32$	Rate	$N_t = 64$	Rate	$N_t = 128$
Error	$2.7444e-3$	1.9696	$7.0073e-4$	1.9936	$1.7595e-4$	2.0006	$4.3970e-5$

TABLE 4: l_2 -norm error and temporal convergence rate of numerical solutions.

	$N_x = 51$	Rate	$N_x = 101$	Rate	$N_x = 201$	Rate	$N_x = 401$
Error	$2.1491e-3$	1.9268	$5.6525e-4$	1.9892	$1.4238e-4$	2.0479	$3.4432e-5$

4. Conclusions

In this study, we presented an unconditionally stable positivity-preserving numerical method for the Fisher–KPP equation. The proposed method is based on the

OSM and an interpolation method. The computational results demonstrated the unconditional stability and positivity-preserving properties of the proposed scheme. In the future work, we will extend the proposed scheme to multidimensional spaces [20].

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

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

Acknowledgments

The corresponding author (J. S. Kim) was supported by the Brain Korea 21 FOUR from the Ministry of Education of Korea.

References

- [1] R. A. Fisher, "The wave of advance of advantageous genes," *Annals of eugenics*, vol. 7, pp. 355–369, 1936.
- [2] A. Kolmogorov, N. Petrovsky, and S. Piscounov, "Étude de l'équations de la diffusion avec croissance de la quantité de matière et son application a un problème biologique," *Bull Univ Moskow, Ser Internat, Sec A*, vol. 1, pp. 1–25, 1937.
- [3] M. M. Khader and K. M. Saad, "A numerical approach for solving the fractional Fisher equation using Chebyshev spectral collocation method," *Chaos, Solitons & Fractals*, vol. 110, pp. 169–177, 2018.
- [4] M. M. Khader and M. Adel, "Numerical and theoretical treatment based on the compact finite difference and spectral collocation algorithms of the space fractional-order Fisher's equation," *International Journal of Modern Physics C*, vol. 31, no. 9, pp. 1–13, 2020.
- [5] S. Kumar, A. Kumar, B. Samet, and H. Dutta, "A study on fractional host-parasitoid population dynamical model to describe insect species," *Numerical Methods for Partial Differential Equations*, vol. 37, no. 2, pp. 1673–1692, 2021.
- [6] P. Veerasha, D. G. Prakasha, J. Singh, I. Khan, and D. Kumar, "Analytical approach for fractional extended Fisher–Kolmogorov equation with Mittag–Leffler kernel," *Advances in Difference Equations*, vol. 174, no. 1, pp. 1–17, 2020.
- [7] J. E. Macías-Díaz and A. Puri, "An explicit positivity-preserving finite-difference scheme for the classical Fisher–Kolmogorov–Petrovsky–Piscounov equation," *Applied Mathematics and Computation*, vol. 218, no. 9, pp. 5829–5837, 2012.
- [8] V. Chandraker, A. Awasthi, and S. Jayaraj, "Implicit numerical techniques for Fisher equation," *Journal of Information and Optimization Sciences*, vol. 39, no. 1, pp. 1–13, 2018.
- [9] M. Izadi, "Split-step finite difference schemes for solving the nonlinear Fisher equation," *Journal of Mahani Mathematical Research Center*, vol. 7, no. 1, pp. 37–55, 2018.
- [10] M. Izadi, "A second-order accurate finite-difference scheme for the classical Fisher–Kolmogorov–Petrovsky–Piscounov equation," *Journal of Information and Optimization Sciences*, vol. 42, no. 2, pp. 431–448, 2021.
- [11] X. Zhang, L. Yao, and J. Liu, "Numerical study of Fisher's equation by the RBF-FD method," *Applied Mathematics Letters*, vol. 120, Article ID 107195, 2021.
- [12] R. K. Saeed and A. A. Mustafa, "Numerical solution of Fisher–KPP equation by using reduced differential transform method," *In AIP Conference Proceedings*, vol. 1888, Article ID 020045, 2017.
- [13] M. El-Hachem, S. W. McCue, and M. J. Simpson, "Invading and receding sharp-fronted travelling waves," *Bulletin of Mathematical Biology*, vol. 83, no. 4, pp. 1–25, 2021.
- [14] J. Xu, F. Zhao, Z. Sheng, and G. Yuan, "A nonlinear finite volume scheme preserving maximum principle for diffusion equations," *Communications in Computational Physics*, vol. 29, no. 3, pp. 747–766, 2021.
- [15] T. Kadri and K. Omrani, "A fourth-order accurate finite difference scheme for the extended-Fisher–Kolmogorov equation," *Bulletin of the Korean Mathematical Society*, vol. 55, no. 1, pp. 297–310, 2020.
- [16] A. Başhan, Y. Ucar, N. M. Yamurlu, and A. Esen, "Numerical solutions for the fourth order extended Fisher–Kolmogorov equation with high accuracy by differential quadrature method," *Sigma Journal of Engineering and Natural Sciences*, vol. 9, no. 3, pp. 273–284, 2018.
- [17] O. P. Yadav and R. Jiwar, "Finite element analysis and approximation of Burgers'–Fisher equation," *Numerical Methods for Partial Differential Equations*, vol. 33, no. 5, pp. 1652–1677, 2017.
- [18] F. Bonizzoni, M. Braukhoff, A. Jüngel, and I. Perugia, "A structure-preserving discontinuous Galerkin scheme for the Fisher–KPP equation," *Numerische Mathematik*, vol. 146, no. 1, pp. 119–157, 2020.
- [19] M. Ilati and M. Dehghan, "Direct local boundary integral equation method for numerical solution of extended Fisher–Kolmogorov equation," *Engineering with Computers*, vol. 34, no. 1, pp. 203–213, 2018.
- [20] Ö. Oruç, "An efficient wavelet collocation method for non-linear two-space dimensional Fisher–Kolmogorov–Petrovsky–Piscounov equation and two-space dimensional extended Fisher–Kolmogorov equation," *Engineering with Computers*, vol. 36, no. 3, pp. 839–856, 2020.
- [21] P. Castillo and S. Gómez, "An interpolatory directional splitting-local discontinuous Galerkin method with application to pattern formation in 2D/3D," *Applied Mathematics and Computation*, vol. 397, Article ID 125984, 2021.
- [22] M. H. Kabir, "Reaction-diffusion modeling of the spread of spruce budworm in boreal ecosystem," *Journal of Applied Mathematics and Computing*, vol. 66, no. 1, pp. 203–219, 2021.
- [23] S. W. McCue, M. El-Hachem, and M. J. Simpson, "Exact sharp-fronted travelling wave solutions of the Fisher–KPP equation," *Applied Mathematics Letters*, vol. 114, Article ID 106918, 2021.
- [24] J. Cai and H. Gu, "Asymptotic behavior of solutions of free boundary problems for Fisher–kpp equation," *Journal of Dynamics and Differential Equations*, vol. 33, no. 2, pp. 913–940, 2021.
- [25] J. Brasseur, "The role of the range of dispersal in a nonlocal Fisher–KPP equation: an asymptotic analysis," *Communications in Contemporary Mathematics*, vol. 23, no. 3, Article ID 2050032, 2021.
- [26] R. B. Salako and W. Shen, "Long time behavior of random and nonautonomous Fisher–kpp equations: Part I-stability of equilibria and spreading speeds," *Journal of Dynamics and Differential Equations*, vol. 33, no. 2, pp. 1035–1070, 2021.
- [27] J. F. Leyva and R. G. Plaza, "Spectral stability of traveling fronts for reaction diffusion-degenerate Fisher–kpp equations," *Journal of Dynamics and Differential Equations*, vol. 32, no. 3, pp. 1311–1342, 2020.

- [28] G. Tian, Z.-C. Wang, and G.-B. Zhang, "Stability of traveling waves of the nonlocal Fisher-KPP equation," *Nonlinear Analysis*, vol. 211, Article ID 112399, 2021.
- [29] C. Lee, H. Kim, S. Yoon et al., "An unconditionally stable scheme for the Allen-Cahn equation with high-order polynomial free energy," *Communications in Nonlinear Science and Numerical Simulation*, vol. 95, Article ID 105658, 2021.
- [30] S. Kim, D. Jeong, C. Lee, and J. Kim, "Finite difference method for the multi-asset Black-Scholes equations," *Mathematics*, vol. 8, no. 3, p. 391, 2020.
- [31] X. Y. Wang, "Exact and explicit solitary wave solutions for the generalised Fisher equation," *Physics Letters A*, vol. 131, no. 4-5, pp. 277-279, 1988.