SPLIT-STEP FINITE DIFFERENCE SCHEMES FOR SOLVING THE NONLINEAR FISHER EQUATION

MOHAMMAD IZADI
DEPARTMENT OF APPLIED MATHEMATICS
FACULTY OF MATHEMATICS AND COMPUTER
MAHANI MATHEMATICAL RESEARCH CENTER
SHAHID BAHONAR UNIVERSITY OF KERMAN
KERMAN, IRAN
E-MAIL: IZADI@UK.AC.IR

(Received:20 August 2018, Accepted: 31 October 2018)

Abstract. In this work, we propose several simple but accurate finite difference schemes to approximate the solutions of the nonlinear Fisher equation, which describes an interaction between logistic growth and diffusion process occurring in many biological and chemical phenomena. All schemes are based upon the time-splitting finite difference approximations. The operator splitting transforms the original problem into two subproblems: nonlinear logistic and linear diffusion, each with its own boundary conditions. The diffusion equation is solved by three well-known stable and consistent methods while the logistic equation by a combination of method of lagging and a two-step approximation that is not only preserve positivity but also boundedness. The new proposed schemes and the previous standard schemes are tested on a range of problems with analytical solutions. A comparison shows that the new schemes are simple, effective and very successful in solving the Fisher equation.

AMS Classification: 65F05, 65K05.
Keywords: Fisher equation, Finite difference schemes, Method of lagging, Operator splitting, Reaction-diffusion equation.

1. Introduction

The Fisher equation is a nonlinear model equation which arises in a wide variety of problems involving diffusion and reaction. The Fisher equation has been introduced in 1937 by Fisher [6] and, simultaneously, by Kolmogorov et al. [12]

JOURNAL OF MAHANI MATHEMATICAL RESEARCH CENTER
VOL. 7, NUMBERS 1-2 (2018) 37-55.
DOI: 10.22103/JMMRC.2018.12605.1069

for studying wave propagation phenomena of a gene in a population and logistic growth-diffusion phenomena. This equation as a prototype for reaction-diffusion equations is also appeared in a number of areas of science and engineering, among others, for the evolution of a neutron population in a nuclear reactor, the theory of combustion, diffusion and mass transfer, nonlinear diffusion, chemical kinetics, ecology, chemical wave propagation, see [5],[16] and reference therein.

To be more specific, we will study the mathematical model that involving reaction and diffusion in the second-order nonlinear partial differential equation of the form

$$\frac{\partial u}{\partial t} = \mathcal{L}(u) := D\frac{\partial^2 u}{\partial x^2} + F(u), \quad (x,t) \in \Omega := \mathbb{R} \times (0,\infty), \tag{1a}$$

with initial condition

$$u(x,0) = u_0(x), \quad x \in \mathbb{R}, \tag{1b}$$

where $u_0(x)$ is a given real valued function and must be chosen as

$$0 \leq u_0(x) \leq 1, \quad x \in \mathbb{R}.$$

The latter restriction motivates by the fact that the Fisher equation describes the nonlinear evolution of a population in a one-dimensional habitat and the habitat can support only a certain maximum population per unit length, where the unity is used for convenience and physically the population density or concentration cannot be negative. The nonnegative constant D is called the diffusion coefficient and the nonlinear function F describes the reaction of the system. Particularly, choosing the nonlinear term $F(u)$ equal to $Ku(1-u)$ in (1a) yields the Fisher equation, where $K(>0)$ is a constant parameter. In this work, two kinds of boundary conditions are supplemented with the initial-value problem (1a)- (1b):

$$\lim_{|x|\to+\infty} u(x,t) = 0, \tag{1c}$$

$$\lim_{x\to-\infty} u(x,t) = g_l(t), \quad \lim_{x\to+\infty} u(x,t) = g_r(t). \tag{1d}$$

Note that it is natural that in practice and computational setting the physical domain $(-\infty,+\infty)$ will be replaced by a finite interval $[x_l, x_r]$. We emphasize that from the property of boundedness of $u_0(x) \in [0,1]$ one can conclude the boundedness of $0 \leq u(x,t) \leq 1$, see cf. [17].

The properties of Fisher equation have been studied from both theoretical and numerical point of views by many researchers. It is shown in [5] that the Fisher equation describes a balance between linear diffusion and nonlinear local multiplication, and it admits shock-type solutions. It has been shown that by Kolmogorov et al. [12] that with appropriate initial and boundary conditions, the Fisher equation will support travelling waves of the form $u = u(z), z = x - ct$ moving in the positive x-direction, provided that the speed $c \geq 2\sqrt{KD}$. Many reaction-diffusion equations admit travelling wave solutions, which have important applications in chemistry,

biology and medicine. The study of travelling wave solution of Fisher's equation has been studied by many authors [2, 11, 13]. Moreover, in [23] an analytical study of Fisher's equation is considered by employing Adomian decomposition method.

From numerical perspective, various computational techniques such as finite difference, spectral, and finite element methods have been developed in the past to solve the nonlinear equation (1a) numerically. The works [7],[18] used the pseudospectral method accurately to discretize spatial derivatives for numerically solving the Fisher equation. The paper [24] presented an interesting and precise numerical study of the Fisher equation by a Petrov-Galerkin finite element method. In [8], the authors investigated the solution of the Fisher's equation by the exponential B-spline Galerkin method. In the context of finite difference schemes, various explicit and implicit algorithms have been reported in [7, 19, 3, 14, 4]. In this work, we are concerned with developing a number of simple finite difference methods that are based on operator splitting strategy.

The rest of this paper is divided into three sections: In Section 2, we first introduce some notations that will be used later on. Then, we briefly describe the main ideas of operator splitting techniques for solving the time-dependent reaction-diffusion problems by decomposing them into a linear and nonlinear subproblems. This consists of introducing two popular splitting schemes i.e., sequential and Strang splitting strategies, which are shortly illustrated from the viewpoint of the local splitting errors. Hence, we introduce various numerical techniques to approximate the solutions of linear and nonlinear subproblems individually such that by combining them the solution of the Fisher equation (1) is constructed. In addition, for the methods proposed to deal with the nonlinear subproblem, we establish the fact that each method is consistent and capable of preserving the positivity and boundedness under some reasonable parameter constraints. In computational Section 3, the performance and accuracy of proposed numerical schemes are verified by performing several simulations on two standard test problems with analytical exact solutions. Finally, the manuscript ends with a summary of the main conclusions in Section 4.

2. Numerical Schemes

In this part, we shall propose several finite difference methods for the Fisher equation in (1). To do this we first introduce some basic notation. We begin by first partitioning the spatial domain (x_l, x_r) into M subintervals with grid points $x_j = x_l + j\Delta x$, for $j = 0, \ldots, M$ so that $x_0 = x_l$ and $x_M = x_r$. Here, by $\Delta x = x_{j+1} - x_j$ we denote the uniform spatial mesh width of size $\Delta x = \frac{x_r - x_l}{M}$. Similarly, we subdivide the time interval $(0, T]$ into N subintervals with the uniform time step $\Delta t = t_{n+1} - t_n$, where $t_n = n\Delta t$ for $n = 0, 1, \ldots, N$. On the computational grid (x_j, t_n) we use the quantity U_j^n represents the computed finite difference approximation to the exact solution $u(x_j, t_n)$ of (1). The following notations shall be used throughout

the paper

$$r = \frac{D\Delta t}{\Delta x^2}, \quad s = K\Delta t.$$

The main idea of operator splitting technique for the nonlinear equations is to decompose a problem into linear and nonlinear subproblems [9, 22]. Hence, one needs to solve each subproblem separately for small time steps Δt, and then concatenate the solutions at the end of each time step. To solve the Fisher equation numerically, we use this idea to decompose the differential operator $\mathcal{L}$ in (1a) into a sum of two simpler differential operators as

$$\mathcal{L}(u) = (\mathcal{L}_1 + \mathcal{L}_2)(u), \tag{2}$$

where $\mathcal{L}_1(u) = D\frac{\partial^2 u}{\partial x^2}$ is the linear operator while $\mathcal{L}_2(u) = F(u)$ is the nonlinear operator. To proceed, we require to approximate the exact solution of (2) by solving the linear and nonlinear subproblems

$$u_t = \mathcal{L}_1(u), \tag{3a}$$

$$u_t = \mathcal{L}_2(u), \tag{3b}$$

in a given sequential order. Note that this order can also be reversed and there is no clear indications with respect to the proper choice of this ordering exist in the literature. Now, each subproblems can be discretized independently using different methods.

In this work we are mainly aimed at the first order operator splitting method which is also known as sequential splitting. However, to obtain more accurate results the second order splitting methods such as Strang method can be easily applied to our model problem. The evolution of time in the first-order operator splitting methods consists of two steps that mathematically can be expressed as

$$u(x, t + \Delta t) = \left(\mathcal{L}_2^{\Delta t} \circ \mathcal{L}_1^{\Delta t}\right) u(x, t) + \mathcal{O}(\Delta t^2), \tag{4}$$

where $\mathcal{L}_1^{\Delta t}$ and $\mathcal{L}_2^{\Delta t}$ are solution operators corresponding to (3a) and (3b) respectively. This implies that in the first step, an intermediate solution is computed by proceeding the solution according to the linear diffusion equation (3a) using the initial condition of the original problem. Then, in the second step, applying the obtained solution at time Δt as an initial condition, the solution is advanced according to the nonlinear equation (3b).

In the second-order operator splitting methods, the advancement in time is carried out in three steps. This particular method is also called the standard *Strang* splitting. In this version the solution takes the form

$$u(x, t + \Delta t) = \left(\mathcal{L}_1^{\Delta t/2} \circ \mathcal{L}_2^{\Delta t} \circ \mathcal{L}_1^{\Delta t/2}\right) u(x, t) + \mathcal{O}(\Delta t^3). \tag{5}$$

So for the second-order version of the method, as a first step, we solve (3a) using the initial condition of the original problem, and then, utilize the obtained solution

as an initial condition to solve (3b), at last use the solution as the initial condition, solve (3a) and obtain the approximate solution.

It is theoretically proven (cf. [10]) that second-order accuracy of the method (5) is guaranteed when the two parts are solved with at least second-order accuracy. In this work, we solve the nonlinear subequation (3a) by using a combined method of lagging and the MacCormack scheme, which are second and first-order schemes in time respectively. Thus, to achieve a second-order accuracy with (5) we require that the linear subequation (3b) is solved by a second-order method. For this purpose, we will use for instance the alternating-direction methods, Crank-Nicolson for solving the linear counterpart. However, we will also utilize simple first-order methods like FTCS (forward in time and central in space) for the diffusion equation, which may only be first-order in time. Clearly, in this case one achieves only the first-order accuracy in time and our splitting will be based upon the first-order version (4).

To solve the linear diffusion equation (3a) several numerical procedures may be applied. Among other, the following schemes are considered in this work:

A) Laasonen method or forward in time and implicit central in space (FTICS):.

$$-r\,U_{j-1}^{n+1} + \big(1+2r\big)U_j^{n+1} - r\,U_{j+1}^{n+1} = U_j^n.$$

This is an implicit, first-order accurate with truncation error $\mathcal{O}(\Delta t + \Delta x^2)$, and unconditionally stable scheme [25].

B) Crank-Nicolson (CN) method:

$$-\frac{r}{2}\,U_{j-1}^{n+1} + \big(1+r\big)U_j^{n+1} - \frac{r}{2}\,U_{j+1}^{n+1} = -\frac{r}{2}\,U_{j-1}^{n} + \big(1-r\big)U_j^{n} - \frac{r}{2}\,U_{j+1}^{n}.$$

This is an implicit, second-order accurate with truncation error $\mathcal{O}(\Delta t^2 + \Delta x^2)$, and unconditionally stable scheme [25]. It should be note that when using the two above schemes for solving the diffusion equation, in each time step the solution procedure involves only solving tridiagonal matrices. Next, we introduce a seemingly implicit scheme that has not such property and indeed is an explicit method.

C) Alternating-directional explicit (ADE) methods: The left to right (L→R) and right to left (R→L) numerical schemes for the diffuion equation (3a) due to Saulyev [20],[21] are respectively defined as

$$\big(1+r\big)\,U_j^{n+1} = r\,U_{j-1}^{n+1} + \big(1-r\big)\,U_j^{n} + r\,U_{j+1}^{n}, \tag{6a}$$

$$\big(1+r\big)\,U_j^{n+1} = r\,U_{j+1}^{n+1} + \big(1-r\big)\,U_j^{n} + r\,U_{j-1}^{n}. \tag{6b}$$

We emphasize that both the L→R and R→L formulae are seemingly implicit in nature but can be solved in an explicit manner from left (right) to right (left) using the imposed boundary condition on the left (right) to get started. Based

upon the L→R and R→L approximations, the first ADE scheme was proposed by Saulyev as follows [21]:

$$(1+r)\,U_j^{n+1} = r\,U_{j-1}^{n+1} + (1-r)\,U_j^n + r\,U_{j+1}^n, \tag{7a}$$

$$(1+r)\,U_j^{n+2} = r\,U_{j+1}^{n+2} + (1-r)\,U_j^{n+1} + r\,U_{j-1}^{n+1}. \tag{7b}$$

This is second-order accurate with truncation error $\mathcal{O}(\Delta t^2+\Delta x^2+\Delta t^2/\Delta x^2)$, and is an unconditionally stable scheme. Note that if r is constant, the scheme is formally first-order due to the presence of the inconsistent term $\mathcal{O}(\Delta t/\Delta x)^2$. To get ride of this inconsistence term, the second ADE version is proposed in [1], in which the computations are performed simultaneously in both L→R and R→L directions and the resulting solutions are averaged to obtain the ultimate $U_j^{n+1} = \frac{1}{2}\left(p_j^{n+1} + q_j^{n+1}\right)$:

$$(1+r)\,p_j^{n+1} = r\,p_{j-1}^{n+1} + (1-r)\,p_j^n + r\,p_{j+1}^n, \tag{8a}$$

$$(1+r)\,q_j^{n+1} = r\,q_{j+1}^{n+1} + (1-r)\,q_j^n + r\,p_{j-1}^{n+1}. \tag{8b}$$

This scheme is also unconditionally stable and its truncation error is $\mathcal{O}(\Delta t^2+\Delta x^2)$. In fact, when averaging is used the term $\mathcal{O}(\Delta t/\Delta x)^2$ cancel out an the resulting method becomes second-order.

On the other hand, to solve the nonlinear part (3b) numerically, the following schemes are employed:

D) Forward in time and combination of method of lagging (FTLag(θ)): For devising the first numerical method for the nonlinear equation (3a), the time derivative is approximated by the usual forward-difference expression

$$\left(\frac{\partial u}{\partial t}\right)_j^n \approx \frac{U_j^{n+1}-U_j^n}{\Delta t}.$$

To approximate the nonlinear term $F(u)$, we exploit the method of lagging to convert this nonlinear term into a linear term. In this approach, one term in the multiplication term $Ku(1-u)$ is calculated at the time level t_n while the other one is calculated at the time level t_{n+1}. For this purpose, we propose the following two approximations

$$\Big(u(1-u)\Big)_j^n \approx U_j^{n+1}(1-U_j^n), \tag{9a}$$

$$\Big(u(1-u)\Big)_j^n \approx U_j^n(1-U_j^{n+1}). \tag{9b}$$

Now, by introducing the parameter $\theta \in [0,1]$ and plugging the above finite difference approximations into the Fisher equation, one gets

$$\frac{U_j^{n+1}-U_j^n}{\Delta t} = \theta K U_j^{n+1}(1-U_j^n) + (1-\theta)K U_j^n(1-U_j^{n+1}). \tag{10}$$

After an easy calculation one can derive U_j^{n+1} from (10) explicitly in term of U_j^n as

$$U_j^{n+1} = \frac{1+s(1-\theta)}{1-s(\theta-U_j^n)}U_j^n, \quad \theta \in [0,1].$$

The next proposition establish conditions on the parameter s under which the property of positivity is satisfied:

Proposition 2.1. *Assume that U_j^n be non-negative real numbers such that $0 \leq U_j^n \leq 1$.*

i) If $\theta = 0$, then (10) provides a non-negative approximation U_j^{n+1} to solutions of (3b) so that $0 \leq U_j^{n+1} \leq 1$.

ii) If $0 < \theta \leq 1$, then (10) provides a non-negative approximation U_j^{n+1} to solutions of (3b) so that $0 \leq U_j^{n+1} \leq \frac{1+s(1-\theta)}{1-s\theta}U_j^n$ under restriction $0 < s < \frac{1}{\theta}$.

Proof: The proof for $\theta = 0$ is straightforward; just by taking $\theta = 0$ in (10) and using the fact $0 \leq U_j^n \leq 1$ we get

$$0 \leq U_j^{n+1} = \frac{1+s}{1+sU_j^n}U_j^n = \frac{U_j^n + sU_j^n}{1+sU_j^n} \leq \frac{1+sU_j^n}{1+sU_j^n} = 1.$$

To prove ii), starting from $0 \leq U_j^n \leq 1$ after some calculations one concludes that

$$\frac{1}{1+s(1-\theta)} \leq \frac{1}{1-s(\theta-U_j^n)} \leq \frac{1}{1-s\theta}, \quad 0 < \theta \leq 1.$$

From the last relation we immediately obtain that

$$0 \leq U_j^n \leq U_j^{n+1} \leq \frac{1+s(1-\theta)}{1-s\theta}U_j^n, \quad 0 < \theta \leq 1.$$

Therefore to have positivity of U_j^{n+1} one needs to have $1 - s\theta > 0$ which completes the proof. ■

Let us investigate the accuracy and consistency of the propose scheme (10). By definition, cf. [25], a finite difference approximation of a PDE is consistent if the finite difference equation approaches the PDE as the mesh size tends to zero. To this end, we consider the local truncation error (L.T.E.) associated with the finite difference equation (10). Assuming that the exact solution u is sufficiently smooth and using the notation $u_j^n = u(x_j, t_n)$ the L.T.E. can be written as

$$\text{L.T.E.} = \frac{u_j^{n+1} - u_j^n}{\Delta t} - \theta K u_j^{n+1}(1-u_j^n) - (1-\theta)K u_j^n(1-u_j^{n+1}). \tag{11}$$

Now, by expanding each u_j^{n+1} in a Taylor series expansion about u_j^n and substituting in (11) followed by an rearrangement we get

$$\begin{aligned}\text{L.T.E.} = &\Big(\frac{\partial u}{\partial t}(x_j,t_n) - Ku_j^n(1-u_j^n)\Big) - K\Delta t\,\frac{\partial u}{\partial t}(x_j,t_n)\Big(\theta - u_j^n\Big) \\ &+ \frac{\Delta t}{2}\frac{\partial^2 u}{\partial t^2}(x_j,t_n) + K\frac{\Delta t^2}{2}\frac{\partial^2 u}{\partial t^2}(x_j,t_n)\Big(u_j^n - \theta\Big).\end{aligned} \tag{12}$$

Since u is the exact solution of $\frac{\partial u}{\partial t} = Ku(1-u)$, the first term in (12) is vanished. Differentiation with respect to t from the exact solution we obtain $\frac{\partial^2 u}{\partial t^2} = K\frac{\partial u}{\partial t}(1-2u)$. Replacing $\frac{\partial^2 u}{\partial t^2}$ in the third term of (12) and combining with the second term we finally arrive at

$$\text{L.T.E.} = K\,\Delta t\left(\frac{1}{2}-\theta\right)\frac{\partial u}{\partial t}(x_j,t_n) + K\frac{\Delta t^2}{2}\frac{\partial^2 u}{\partial t^2}(x_j,t_n)\Big(u_j^n-\theta\Big). \tag{13}$$

Therefore, we have proved the following proposition:

Proposition 2.2. *Suppose that U_j^n generated by numerical scheme* (10)*. Then, the truncation error of this scheme satisfies*

$$L.T.E. = \begin{cases} \mathcal{O}(\Delta t^2), & \theta = \dfrac{1}{2}, \\ \mathcal{O}(\Delta t), & \theta \neq \dfrac{1}{2}. \end{cases}$$

E) Modified MacCormack method (MMac): The traditional MacCormack scheme[15] is known to be explicit and second-order accurate in both space and time, and has been proved its efficiency and accuracy to hyperbolic and parabolic equations. The popularity of MacCormack's explicit method is due in part to its simplicity and ease of implementation.

The standard algorithm based on MacCormack original scheme consists of a two-stage procedure known as the predictor-corrector method. This scheme for the nonlinear subequation (3b) takes the form

$$U_j^{n+1} = \frac{1}{2}(U_j^\star + U_j^{\star\star}), \tag{14}$$

where the predicted and corrected values $U_j^\star$ and $U_j^{\star\star}$ are obtained via the following difference schemes

$$U_j^\star = U_j^n + \Delta t\, F_j^n = U_j^n + s\, U_j^n\left(1-U_j^n\right), \tag{15a}$$

$$U_j^{\star\star} = U_j^n + \Delta t\, F_j^\star = U_j^n + s\, U_j^\star\left(1-U_j^\star\right), \tag{15b}$$

with $F_j^n = F(U_j^n)$ and $F_j^\star = F(U_j^\star)$. We refer to this scheme as explicit MacCormack method (EMac). A similar result as Proposition 2.1 can be proved for the two-stage scheme (15).

Proposition 2.3. *Assume that U_j^n be non-negative real numbers such that $0 \leq U_j^n \leq 1$. If $0 < s \leq 1$, then* (15) *provides a non-negative approximation U_j^{n+1} to solutions of* (3b)*.*

Proof: Since $0 \leq U_j^n \leq 1$, then clearly $0 \leq 1-U_j^n \leq 1$ and $U_j^\star \geq 0$. Thus, one can immediately prove that

$$1-s \leq 1-sU_j^n \leq 1.$$

It is an easy task to show that

$$1-U_j^\star = (1-U_j^n)(1-sU_j^n).$$

This implies that under condition $s \leq 1$ we have that $1 - U_j^\star \geq 0$ and consequently $U_j^{\star\star} \geq 0$. ■

To show that (15) is an consistent scheme, we first combine the two stages (15a)-(15b) into one and rewrite (15) as

$$U_j^{n+1} = U_j^n + sU_j^n(1-U_j^n) + \frac{s^2}{2}U_j^n(1-U_j^n)\left(1 - sU_j^n(1+s(1-U_j^n))\right).$$

It is not a difficult task to show that this scheme is a consistent first-order method. Indeed, we have

$$\begin{aligned}\text{L.T.E.} &= \Big(\frac{\partial u}{\partial t}(x_j,t_n) - Ku_j^n(1-u_j^n)\Big)\\ &+ \frac{\Delta t}{2}\Big(\frac{\partial^2 u}{\partial t^2}(x_j,t_n) - K^2u_j^n(1-u_j^n)\Big) + \mathcal{O}(\Delta t^2) = \mathcal{O}(\Delta t).\end{aligned}$$

Although applying the explicit MacCormack scheme (15a)-(15b) to (3b) is a positivity preserving method, but not necessarily guaranteed the boundedness, i.e., $0 \leq U_j^{n+1} \leq 1$. Next, we combine the idea of the method of lagging with the MacCormack procedure to propose a scheme that is not only a positivity preserving method but also guarantees that $0 \leq U_j^{n+1} \leq 1$ if we started with $0 \leq U_j^n \leq 1$. Therefore, we modify (15a)-(15b) appropriately to obtain (MMac)

$$U_j^\star = U_j^n + s\,U_j^n\left(1 - U_j^\star\right), \tag{16a}$$

$$U_j^{\star\star} = U_j^n + s\,U_j^\star\left(1 - U_j^{\star\star}\right). \tag{16b}$$

Solving equations (16a)-(16b) in terms of $U_j^\star$ and $U_j^{\star\star}$ respectively we find that

$$U_j^\star = \frac{(1+s)U_j^n}{1+sU_j^n}, \quad U_j^{\star\star} = \frac{U_j^n + sU_j^\star}{1+sU_j^\star}. \tag{17}$$

Starting with the assumption that $0 \leq U_j^n \leq 1$, we immediately conclude that from (17) that

$$0 \leq U_j^\star \leq 1 \Longrightarrow 0 \leq U_j^{\star\star} \leq 1 \Longrightarrow 0 \leq U_j^{n+1} = \frac{U_j^\star + U_j^{\star\star}}{2} \leq 1.$$

It is remained to show that (15) is consistent. For this purpose, we add (16a) and (16b) together and apply (14) to obtain

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} = \frac{K}{2}\Big(U_j^n(1-U_j^\star) + U_j^\star(1-U_j^{\star\star})\Big). \tag{18}$$

Using the facts that

$$1 - U_j^\star = \frac{1-U_j^n}{1+sU_j^n}, \quad 1 - U_j^{\star\star} = \frac{1-U_j^n}{1+sU_j^\star}, \quad (1+sU_j^n)(1+sU_j^\star) = 1 + s(s+2)U_j^n,$$

we can rewrite (18) as follows

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} = \frac{K}{2}U_j^n(1-U_j^n)\Big(\frac{1}{1+sU_j^n} + \frac{s+1}{1+s(s+2)U_j^n}\Big).$$

Therefore, by expanding the terms in the last equation the L.T.E. for the scheme (15) is simplified into

$$\begin{aligned}\text{L.T.E.} &= \frac{\partial u}{\partial t}(x_j,t_n)+\frac{\Delta t}{2}\frac{\partial^2 u}{\partial t^2}(x_j,t_n)+\dots\\ &\quad-\frac{K}{2}u_j^n(1-u_j^n)\Big(1-su_j^n+\dots+(s+1)\Big(1-s(s+2)u_j^n+\dots\Big)\Big)\\ &=\Big(\frac{\partial u}{\partial t}(x_j,t_n)-Ku_j^n(1-u_j^n)\Big)+\frac{\Delta t}{2}\Big(\frac{\partial^2 u}{\partial t^2}(x_j,t_n)-K^2u_j^n(1-u_j^n)\Big)=\mathcal{O}(\Delta t).\end{aligned}$$

Thus, the second version of MacCormack is also a first-order method.

To summarize, we consider the performance of the following schemes as combinations of previously proposed numerical methods applied to the linear and nonlinear subproblems (3a) and (3b) respectively:

1. FTICS and FTLag(θ): FTICS-FTLag(θ),
2. FTICS and EMac/MMac: FTICS-EMac/MMac,
3. CN and FTLag(θ): CN-FTLag(θ),
4. CN and EMac/MMac: CN-EMac/MMac,
5. ADE and FTLag(θ): ADE-FTLag(θ),
6. ADE and EMac/MMac: ADE-EMac/MMac.

3. Numerical experiments

In this section, we present some results of computations using the proposed combined finite difference schemes described in the preceding sections to test their accuracy and efficiency when applied to the Fisher equation. To measure the accuracy of the numerical algorithms, we compute the difference between the analytic and numerical solutions. For this purpose, we calculate the discrete L_*-norm error, i.e.,

$$E_* := E_*(\Delta x,\Delta t)=\|\boldsymbol{U}_h-\boldsymbol{u}_{exact}\|_*,$$

where $*$ stands for L_1, L_2 or L_∞ norm. We also compute the relative differences in these classical norms as

$$\rho_*=\frac{E_*}{\|\boldsymbol{u}_{exact}\|_*}.$$

In order to asses the numerical scheme more qualitatively, the convergence orders in temporal and spatial directions are defined respectively as

$$\log_2\left(\frac{E_*(\Delta x,2\Delta t)}{E_*(\Delta x,\Delta t)}\right),\quad \log_2\left(\frac{E_*(2\Delta x,\Delta t)}{E_*(\Delta x,\Delta t)}\right).$$

To test the validity and accuracy of proposed methods, two kinds of initial values according to (1c) and (1d) are considered. These two examples have non-zero boundary conditions with exact solutions. To compare our results with methods available in the standard literature, we also develop three different well-known or existing computational procedures to compare our simulations against the numerical results obtained by these methods. More precisely, we employ the following numerical methods for comparison purposes

- FTCS: Forward in time and central in space [19]

$$\frac{U_j^{n+1}-U_j^n}{\Delta t}=D\frac{U_{j-1}^n-2U_j^n+U_{j+1}^n}{\Delta x^2}+F_j^n. \tag{19}$$

- FTICS: Forward in time and implicit central in space [3]

$$\frac{U_j^{n+1}-U_j^n}{\Delta t}=D\frac{U_{j-1}^{n+1}-2U_j^{n+1}+U_{j+1}^{n+1}}{\Delta x^2}+KU_j^{n+1}(1-U_j^n). \tag{20}$$

- EPPFD(θ): Explicit positivity-preserving finite difference scheme [14]

$$(1-\theta)\frac{U_j^{n+1}-U_j^n}{\Delta t}+\theta\frac{U_j^{n+1}-U_j^{n-1}}{2\Delta t}=D\frac{U_{j-1}^n-(U_j^{n+1}+U_j^{n-1})+U_{j+1}^n}{\Delta x^2}+\frac{F_{j+1}^n+F_{j-1}^n}{2}. \tag{21}$$

Example 3.1. *We first consider the Fisher equations* (1a) *on the computational domain* $[0,1]$ *with the continuous initial profile and boundary conditions given by*

$$u_0(x)=\frac{1}{(1+e^x)^2},\quad g_l(t)=\frac{1}{(1+e^{-5t})^2},\quad g_r(t)=\frac{1}{(1+e^{1-5t})^2},$$

as considered in [3, 4]. *The diffusion coefficient is* $D=1.0$, $K=6.0$. *The analytical solution of the Fisher equation in this case takes the form*

$$u(x,t)=\frac{1}{(1+e^{x-5t})^2}.$$

The simulation results for this example can be found in Table 1-2 and in Fig. 2-3 for different values of Δx and Δt and final time T. In Table 1, we summarize the numerical results at $T=0.01$ obtained by the sequential splitting procedure (4) using $\Delta t=0.001$ for a different choices of the number of spatial grid points $M=25,50,100$, and 200. In this Table, we compare the performance of different schemes ADE/CN/FTICS-FTLag(θ) for special values of $\theta=0,\frac{1}{2},1$, and ADE/CN/FTICS-EMac/MMac in the L_2 norm. The corresponding results based on the Strang splitting procedure (5) are reported in Table 2. Note that in the ADE we utilize the second version (8a)-(8b) which is based on the averaging.

It can be seen from Table 1 that among numerical methods based on first-order splitting strategy, the performance of the second class CN-FTLag(θ)/EMac/MMac is superior compared to two other classes i.e., ADE-FTLag(θ)/EMac/MMac and FTICS-FTLag(θ)/EMac/MMac. In each class, almost the same accuracy is achieved for all schemes in the class and a more closely result is obtained when using FTLag($\frac{1}{2}$) and EMac for approximating the nonlinear part (3b). In the scheme FTLag(θ), we only used three particular cases $\theta=0,\frac{1}{2},1$. For these values of θ, the numerical experiments in Table 1 show that choosing various Δx has a direct influence on the performance. For instance, for $\theta=1$ the smaller errors are obtained correspond to $M=25,50$ while for $\theta=0$ the same behaviour observed with $M=100,200$. To see whether there exits a special value of θ that yields the best performance of the FTLag(θ), we examine the behaviour of this scheme for different values of $\theta\in[0,1]$. Dividing the interval $[0,1]$ into 100 subintervals with $\theta_i=i/100$

Method	$M = 25$	$M = 50$	$M = 100$	$M = 200$
ADE-FTLag(0)	7.6871E-4	7.1868E-4	7.1625E-4	8.1713E-4
ADE-FTLag($\frac{1}{2}$)	7.7665E-4	7.2748E-4	7.2632E-4	8.2883E-4
ADE-FTLag(1)	7.8513E-4	7.3682E-4	7.3685E-4	8.4078E-4
ADE-EMac	7.7663E-4	7.2746E-4	7.2630E-4	8.2881E-4
ADE-MMac	7.7497E-4	7.2564E-4	7.2431E-4	8.2671E-4
CN-FTLag(0)	4.8658E-4	5.4367E-4	3.8962E-4	3.3022E-4
CN-FTLag($\frac{1}{2}$)	4.8064E-4	5.3969E-4	3.9014E-4	3.3692E-4
CN-FTLag(1)	4.7555E-4	5.3652E-4	3.9185E-4	3.4485E-4
CN-EMac	4.8065E-4	5.3969E-4	3.9014E-4	3.3690E-4
CN-MMac	4.8220E-4	5.4089E-4	3.9043E-4	3.3578E-4
FTICS-FTLag(0)	6.9492E-4	6.4291E-4	4.6394E-4	4.6394E-4
FTICS-FTLag($\frac{1}{2}$)	6.9159E-4	6.4141E-4	4.6773E-4	4.7135E-4
FTICS-FTLag(1)	6.8889E-4	6.4061E-4	4.7249E-4	4.8033E-4
FTICS-EMac	6.9159E-4	6.4141E-4	4.6772E-4	4.7133E-4
FTICS-MMac	6.9268E-4	6.4215E-4	4.6732E-4	4.6986E-4

TABLE 1. Comparison of L_2 error norms for different finite difference schemes based on sequential splitting for Example 3.1 with $D = 1.0, K = 6.0, \Delta t = 0.001$ when $\Delta x = 0.04, 0.02, 0.01, 0.005$ evaluated at time $T = 0.01$.

for $i = 0, \ldots, 100$, the behaviour of different methods ADE/CN/FTICS-FTLag(θ_i) measured in the L_2 norm. The results correspond to CN-FTLag(θ_i) are visualized in Fig. 1 for various $M = 25, 50, 100, 200$ with the same values of $\Delta t, T$ as above. The same behaviour for the scheme FTICS-FTLag(θ_i) is observed, which also seen from the Table 1. However, for the method ADE-FTLag(θ_i) the story is different so that for all values of M the error is always an increasing function of θ.

As a consequence, among all proposed schemes based on the sequential splitting, the performance of CN-FTLag(0) is superior for a moderate problem size and hence used for the next computations. Fig. 2 contains two graphs. The right one illustrates the numerical solutions (dashed, dotted, and dashdotted lines) along with the exact solutions (solid lines) for example 3.1 generated using CN-FTLag(0) scheme from $t = 0$ to $T = 1$ with $\Delta t = 0.001$. These solutions are depicted at time steps $\Delta t, 50\Delta t, \ldots, T$. While on the left plot, we investigate the behaviour of errors with respect to time in three different norms L_1, L_2, and L_∞. It can be seen from Fig. 2 that the numerical solutions are very close to the exact ones.

Let us now consider the performance of numerical schemes discussed in Table 1 but with a different based splitting procedure i.e. Strang splitting (5). With the

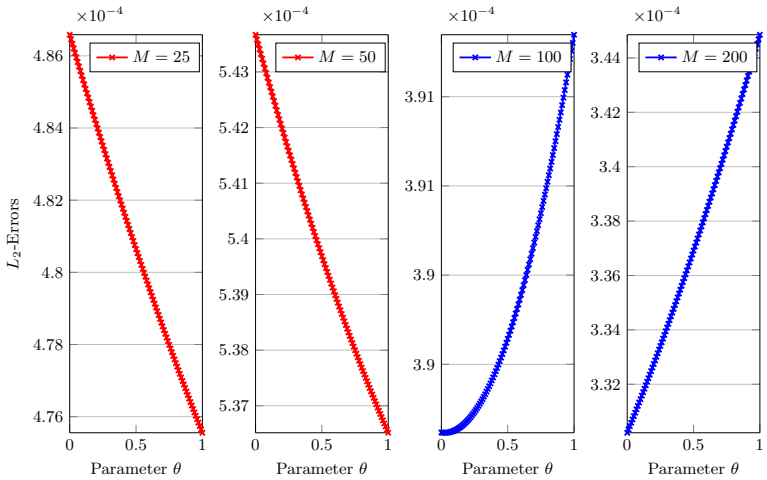


FIGURE 1. L_2 error norm of CN-FTLag(θ) for Example 3.1 with different $M = 25, 50, 100, 200$ from left to right versus the parameter θ.

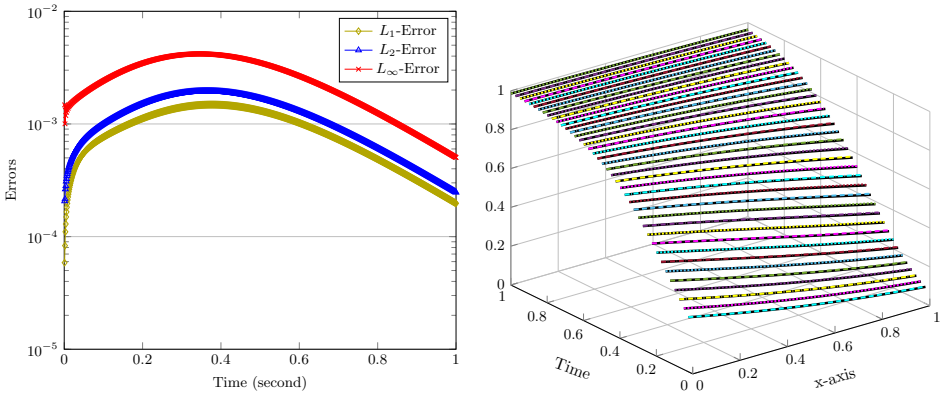


FIGURE 2. L_1, L_2, L_∞ error norms of CN-FTLag(0) at time $t = 1$ (left), numerical and exact (solid lines) solutions at different times $t = s\Delta t$, for $s = 1, 50, \ldots, 1000$ (right).

same problem configurations as for the sequential splitting in Table 1, we report the L_2 error norms for the Strang splitting at time $T = 0.01$ in Table 2. The experimental results in Table 2 reflect that contrary to the sequential splitting, the best accurate numerical schemes in terms of smallest magnitude of L_2 errors based on the Strang splitting is the ADE-FTLag(θ)/EMac/MMac type methods. Again, the results obtained show that in each class the ADE/CN/FTICS-FTLag($\frac{1}{2}$) schemes and the ADE/CN/FTICS-EMac techniques produced errors of similar magnitude.

In the scheme ADE-FTLag(θ), we found that there exists a particular value of $\theta \in [0, 1]$ that yields the best performance. We take $M = 25, 50, 100, 200$ and investigate the performance of this scheme as a function of the parameter θ. The results are depicted in Fig. 3. In all plots, we take $\Delta t = 0.001$, $T = 0.01$. Looking

Method	$M = 25$	$M = 50$	$M = 100$	$M = 200$
ADE-FTLag(0)	3.1773E-4	3.2354E-4	3.2444E-4	3.3735E-4
ADE-FTLag($\frac{1}{2}$)	3.2607E-4	3.3189E-4	3.3290E-4	3.4628E-4
ADE-FTLag(1)	3.3558E-4	3.4139E-4	3.4244E-4	3.5609E-4
ADE-EMac	3.2605E-4	3.3188E-4	3.3288E-4	3.4626E-4
ADE-MMac	3.2429E-4	3.3011E-4	3.3112E-4	3.4450E-4
CN-FTLag(0)	5.8408E-4	7.4740E-4	5.2231E-4	3.1648E-4
CN-FTLag($\frac{1}{2}$)	5.7784E-4	7.4243E-4	5.1923E-4	3.1707E-4
CN-FTLag(1)	5.7229E-4	7.3801E-4	5.1699E-4	3.1908E-4
CN-EMac	5.7785E-4	7.4244E-4	5.1923E-4	3.1707E-4
CN-MMac	5.7944E-4	7.4379E-4	5.2019E-4	3.1718E-4
FTICS-FTLag(0)	6.5695E-4	6.9476E-4	4.4671E-4	2.5475E-4
FTICS-FTLag($\frac{1}{2}$)	6.5171E-4	6.9037E-4	4.4462E-4	2.5778E-4
FTICS-FTLag(1)	6.4710E-4	6.8659E-4	4.4354E-4	2.6252E-4
FTICS-EMac	6.5171E-4	6.9037E-4	4.4462E-4	2.5777E-4
FTICS-MMac	6.5315E-4	6.9164E-4	4.4541E-4	2.5745E-4

TABLE 2. Comparison of L_2 error norms for different finite difference schemes based on Strang splitting for Example 3.1 with $D = 1.0, K = 6.0, \Delta t = 0.001$ when $\Delta x = 0.04, 0.02, 0.01, 0.005$ evaluated at time $T = 0.01$.

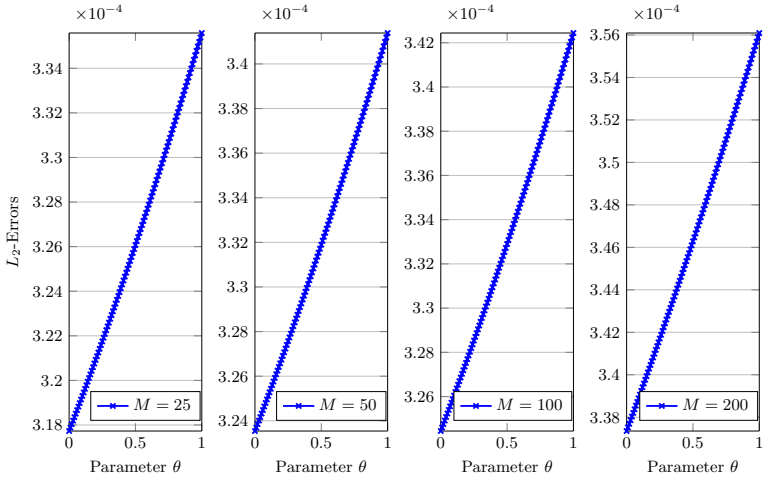


FIGURE 3. L_2 error norm of ADE-FTLag(θ) for Example 3.1 with different $M = 25, 50, 100, 200$ from left to right versus the parameter θ.

at Fig. 3 reveals that the special $\theta = 0$ gives the most accurate results in the ADE-FTLag(θ) based on the Strang splitting. This conclusion can also be deduced from Table 2. On the other hand, our experiments for CN/FTICS-FTLag(θ) (and also are visible in Table 2) show that these two methods are decreasing functions of θ for $M = 25, 50, 100$ while by selecting $M = 200$ they are increased in terms of θ.

Example 3.2. *In the second and last test problem, we consider the Fisher equation with $D = 1.0$, $K = 2.0$. It is known that this equation admits the following travelling wave solution*

$$u(x,t) = \frac{1}{\left(1 + 4\exp(\sqrt{\frac{K}{6}}x - \frac{5K}{6}t)\right)^2}, \quad x \in \mathbb{R},$$

which is taken from [14]. *The initial condition is obtained by setting $t = 0$ and satisfies $u_0(x) \in [0, 1]$. In this case, we take the boundary conditions as*

$$u(-20, t) = 1, \quad u(40, t) = 0, \ t > 0,$$

to ensure that there enough space for wave to propagate. For the simulations, we set $\Delta x = 1$ or $M = 60$, $\Delta t = 0.1$, and the final time is $T = 8$ for computations.

To justify our numerical results, we compare them with the results of numerical procedures described in (19)- (21). Compared with other numerical methods in the class of finite difference schemes, numerical study given in [14] is more satisfactory with regard to its accuracy and stability. The performance of all three classes of finite difference schemes based on both the sequential and Strang splitting procedures (4) and (5) are carried out. Due to the large number of schemes obtained by specifying θ in the FTLag(θ) and to save space, we only report the results correspond to the values of this parameter that have smallest errors.

Table 3 presents the relative differences $\rho_1, \rho_2, \rho_\infty$ using the three norms L_1, L_2, and L_∞ introduced in the beginning of this subsection. These results are obtained at time $T = 8$ and all are based on the sequential as well as the Strang splitting procedures (4) and (5). The results in Table 3 show that, indeed, a slightly accurate result is obtained if one uses the Starng splitting, even though it is more time-consuming instead. Among others, the schemes ADE/CN/FTICS-FTLag(θ) for the special values of parameter θ give the best outcomes. For these schemes the smallest errors are corresponded to the values of θ equal to 0.25/0.31, 0.33/0.35, and 0.31/0.34 respectively for the sequential and Strang splitting procedures. Moreover, the computational results achieved by our proposed method are more accurate approximations to the numerical results obtained through schemes (19)-(20).

Next, for fixed values $\Delta t = 0.1$ and $\Delta x = 1$ we plot the numerical solutions obtained by the schemes ADE/CN/FTICS-FTLag(θ) as the most accurate methods in Table 3 and compared them with the existing methods (19)-(21) at two different times $t = 4$ and $t = 8$ in Fig. 4. The initial profile at time $t = 0$ as well as the

Method	Sequential splitting			Strang splitting		
	ρ_1	ρ_2	ρ_∞	ρ_1	ρ_2	ρ_∞
ADE-FTLag(0.31/0.34)	9.2744E-2	2.5481E-2	9.9985E-3	9.6112E-2	2.7106E-2	1.1313E-2
ADE-EMac	4.7657E-1	1.6170E-1	8.6090E-2	3.6783E-1	1.2556E-1	6.2756E-2
ADE-MMac	3.3216E-1	1.1778E-1	6.1013E-2	2.5401E-1	8.6673E-2	4.6264E-2
CN-FTLag(0.33/0.35)	7.8178E-2	2.1438E-2	8.5299E-2	8.9857E-2	2.4865E-2	1.0584E-2
CN-EMac	4.1294E-1	1.4031E-1	7.0007E-2	3.5225E-1	1.2031E-1	6.0147E-2
CN-MMac	2.7727E-1	9.7687E-2	5.0871E-2	2.4191E-1	8.2061E-2	4.3951E-2
FTICS-FTLag(0.25/0.31)	3.3911E-3	5.9386E-3	1.4432E-2	2.9777E-3	5.2524E-3	1.3563E-2
FTICS-EMac	1.7160E-2	3.7092E-2	1.1368E-1	1.2262E-2	2.6680E-2	8.1989E-2
FTICS-MMac	1.2992E-2	2.9483E-2	9.2383E-2	8.7594E-3	1.9800E-2	6.3429E-2
	ρ_1		ρ_2		ρ_∞	
FTCS (19)	5.1371E-2		1.1532E-1		3.5534E-2	
FTICS (20)	1.1335E-1		2.2534E-1		6.1894E-1	
EPPFD(0.4) (21)	7.5457E-1		1.5046E-1		4.2492E-1	

TABLE 3. Comparison of relative $L_1/L_2/L_\infty$ error norms for different finite difference schemes based on sequential/Strang splitting for Example 3.2 with $D = 1.0, K = 2.0, \Delta t = 0.1$, and $\Delta x = 1$ evaluated at time $T = 8$.

corresponding exact solutions are also represented in Fig. 4. All plots are obtained using the sequential slitting procedure. In all plots, the exact solutions are indicated by solid lines while the numerical counterpart are depicted by coloured curves. In order to see the behaviour of numerical solutions more precisely, we magnify these solutions at time $t = 8$ in Fig. 4.

It can be seen from Fig. 4 and in particular at the magnification box that the numerical results are found by FTICS-FTLag(0.25), CN-FTLag(0.33), and ADE-FTLag(0.31) schemes are very close together and hardly distinguishable from the analytical solutions. On the other hand, the performance of FTCS, FTICS, and EPPFD(0.4) are very poor in particular at the final time $t = 8$. Therefore, in the following we will focus on the schemes ADE/CN/FTICS-FTLag(θ).

In the next simulation, we investigate the impact of refining the spatial mesh size and see the behaviour of L_∞ errors in the methods ADE/CN/FTICS-FTLag(θ) when applied to Example 3.2. We fix the time step size $\Delta t = 0.01$ and employ different mesh points $M = 2^s, s = 2, 3, \ldots, 7$. We first run the ADE/CN/FTICS-FTLag(θ) with the parameter θ respectively equal to $0.34, 0.31, 0.25$ as utilized in Table 3. Although using these values of θ in the corresponding methods yields an order of accuracy about $2.5-3$, but our experiments show that the best convergence rates or superconvergence rates are achieved when the parameter θ in these schemes are chosen as $\theta_A = 0.1625, \theta_C = 0.16875$, and $\theta_F = 0.0875$ respectively. These results evaluated at time $t = 8$ are reported in Fig. 5, left table. The snapshots

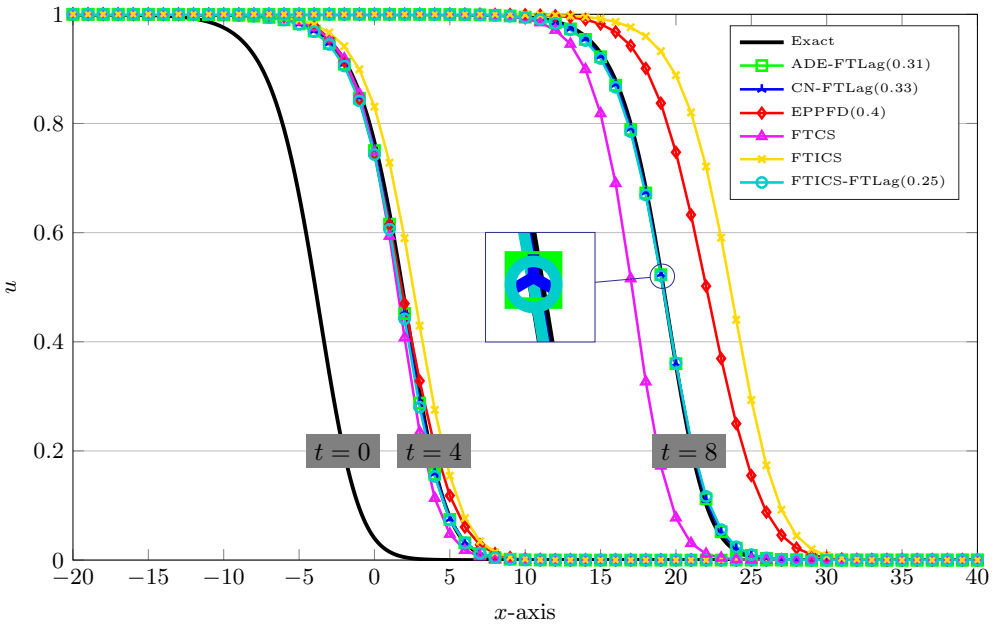


FIGURE 4. Numerical solutions of various finite difference schemes for Example 3.2 with $\Delta t = 0.1$ and $\Delta x = 1$ evaluated at times $t = 5, 10$. The magnification of solutions at time $t = 8$ is plotted in the box. The exact solutions are displayed by thick lines.

of the corresponding convergence rates are displayed in the same figure, right plot. Note that all computations are based on the sequential splitting procedure. The numerical experiments shown in Fig. 5 indicate that achieving an order 4.47 of accuracy is possible, if one uses the CN-FTLag(θ_C) method. More precisely, the corresponding spatial order of convergence for the ADE-FTLag(θ_A) and FTICS-FTLag(θ_F) are about 4.40 and 4.05 respectively.

M	ADE-FTLag (θ_A)	CN-FTLag (θ_C)	FTICS-FTLag (θ_F)
2^2	9.8603E-1	9.8604E-1	9.8592E-1
2^3	8.4306E-1	8.4326E-1	8.4132E-1
2^4	5.6286E-1	5.6313E-1	5.6132E-1
2^5	1.9447E-1	1.9477E-1	1.9431E-1
2^6	4.7765E-2	4.7976E-2	4.8143E-2
2^7	2.2617E-3	2.1509E-3	2.9048E-3

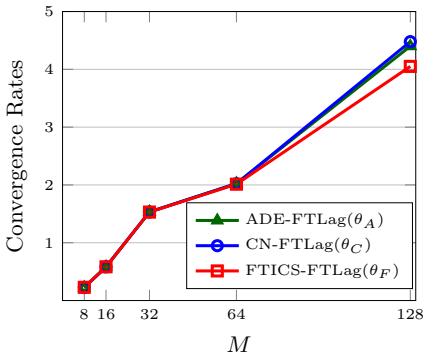
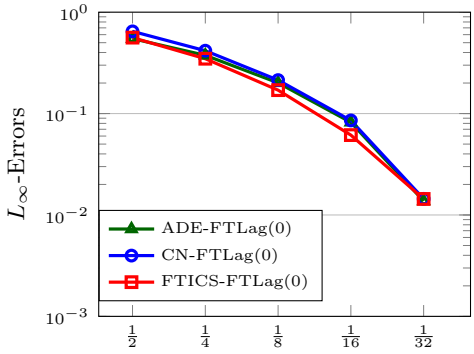


FIGURE 5. L_∞-errors in ADE/CN/FTICS-FTLag($\theta_A/\theta_C/\theta_F$) for Example 3.2 (left) and the corresponding convergence rates (right) at time $t = 8$ for $\Delta t = 0.01$ for different M.

Finally, for Example 3.2 we examine the influence of refining the time step size on the L_∞ errors while the number of spatial grid points M is fixed. To end that, we take $\Delta x = 1$ and consider various $\Delta t = 2^{-s}$ for $s = 1, 2, \ldots, 5$. Fig. 6, left plot, shows the measurement of L_∞ errors in the schemes ADE/CN/FTICS-FTLag(0) evaluated at the final time $t = 8$. The corresponding convergence rates are reported in the left table. It can be clearly from Fig. 6 seen that an achievement of order of accuracy between 2 and 3 is possible if one utilizes the finite difference schemes ADE/CN/FTICS-FTLag(0).



Δt	ADE-FTLag(0)	CN-FTLag(0)	FTICS-FTLag(0)
2^{-1}	-	-	-
2^{-2}	0.55	0.63	0.69
2^{-3}	0.90	0.96	1.02
2^{-4}	1.29	1.32	1.47
2^{-5}	2.58	2.57	2.09

FIGURE 6. L_∞-errors in ADE/CN/FTICS-FTLag(0) for Example 3.2 (left) and the corresponding convergence rates (right) with $\Delta x = 1$ evaluated at time $T = 8$ for different $\Delta t = 2^{-s}, s = 1, 2, 3, 4, 5$.

4. Conclusions

In this paper, various numerical methods are proposed for the one-dimensional reaction-diffusion equations and in particular the nonlinear Fisher equation arising from modelling many real-world phenomena. To be more precise, several simple but sufficiently accurate finite difference schemes based upon operator splitting strategy are applied to this model problem. All methods are consistent of order $\mathcal{O}(\Delta t^\alpha + \Delta x^2)$, for $\alpha = 1$ or 2. The discussed computational procedures solved our model quite satisfactorily compared to the existing and standard finite difference algorithms.

References

[1] H. Z. Barakat, J. A. Clark, On the solution of the diffusion equations by numerical methods, Transaction of the ASME, J. Heat. Transfer., Vol. 88, 421-427 (1966).

[2] P. Brazhnik, J. Tyson, On travelling wave solutions of Fisher's equation in two spatial dimensions, SIAM J. Appl. Math., Vol. 60, Number 2, 371-391 (1999).

[3] V. Chandraker, A. Awasthi, and S. Jayaraj, A numerical treatment of Fisher equation, Procedia Engineering, Vol.127, 1256-1262 (2015).

[4] V. Chandraker, A. Awasthi, and S. Jayaraj, Implicit numerical techniques for Fisher equation, Journal of Information & Optimization Sciences, DOI : 10.1080/02522667.2017.1374722.

[5] J. Canosa, On a nonlinear diffusion equation describing population growth, IBM J. Res. Develop., Vol. 17, 307-313 (1973).

[6] R. A. Fisher, The wave of advance of advantageous genes, Ann. Eugenics, Vol. 7, 353-369 (1937).

[7] J. Gazdag, J. Canosa, Numerical solution of Fisher's equation, J. Appl. Probab. Vol. 11, 445-457 (1974).

[8] M. Z. Gorgulu, I. Dag, Exponential B-splines Galerkin method for the mumerical Solution of the Fisher's equation, Iran. J. Sci. Technol: Trans. Sci., https://doi.org/10.1007/s40995-017-0403-x.

[9] H. Holden, K. H. Karlsen, K.-A. Lie, and N. H. Risebro. Splitting for Partial Differential Equations with Rough Solutions. European Math. Soc. Publishing House, Zürich, 2010.

[10] T. Jahnke, C. Lubich, Error bounds for exponential operator splittings, BIT, Vol. 40, 735-744 (2000).

[11] T. Kawahara, M. Tanaka, Interactions of travelling fronts: an exact solution of a nonlinear diffusion equation, Phys. Lett. A, Vol. 97, Number 8, 311-314 (1983).

[12] A. Kolmogorov, N. Petrovsky, and S. Piscounov, Étude de l'équations de la diffusion avec croissance de la quantitaté de matiére et son application a un probléme biologique, Bull. Univ. Moskou Ser. Int. 1A, 1-25, (1937).

[13] D.A. Larson, Transient bounds and time-asymptotic behaviour of solutions to nonlinear equations of Fisher's type, SIAM J. Appl. Math., Vol. 34, 93-103 (1978).

[14] J.E. Macías-Díaz, A. Puri, An explicit positivity-preserving finite-difference scheme for the classical Fisher-Kolmogorov-Petrovsky-Piscounov equation, Appl. Math. Comput., Vol. 218, 5829-5837 (2012).

[15] R.W. MacCormack, The effect of viscosity in hypervelocity impact cratering, AIAA Paper, 69-354 (1969).

[16] W. Malfliet, Solitary wave solutions of nonlinear wave equations, Am. J. Phys., Vol. 60, Number 7, 650-654 (1992).

[17] J. D. Murray, Mathematical Biology, Springer-Verlag, Berlin, 1989.

[18] D. Olmos, B.D. Shizgal, A pseudospectral method of solution of Fisher's equation, J. Comput. Appl. Maths, Vol. 193, 219-242 (2006).

[19] N. Parekh, S. Puri, A new numerical scheme for the Fisher equation, J. Phys. A: Math. Gen., Vol. 23, L1085-Ll091 (1990).

[20] V. K. Saul'yev. A method of numerical solution for the diffusion equations, Dokl. Akad. Nauk SSSR, Vol. 115, 1077-1099 (1957) (In Russian).

[21] V. K. Saul'yev, Integration of Equations of Parabolic Type by the Method of Nets,. Pergamon Press, Oxford,1964.

[22] H. Q. Wang, Numerical studies on the split-step finite difference method for nonlinear Schrödinger equations, Appl. Math. Comput. 170, 17-35 (2005).

[23] A-M Wazwaz, A. Gorguis, An analytic study of Fisher's equation by using Adomian decomposition method, Appl. Math. Comput. 154, 609-620 (2004).

[24] S. Tang, R.O. Weber. Numerical study of Fisher's equation by a Petrov-Galerkin finite element method, Jour. Austr. Math. Sot. Vol. 33, 27-38 (1991).

[25] Thomas, J.W., Numerical Partial Differential Equations: Finite Difference Methods, Springer-Verlag, 1995.